

# CCIE Enterprise Infrastructure – A Complete Guide

**Authored By:**

**Khawar Butt**

CCIE # 12353

Hepta CCIE#12353

CCDE # 20110020

**CCIE Enterprise Infrastructure – A Complete Guide**



## Table of Contents

<b>Module 1 – Layer 2 Technologies – Ethernet Switching</b>	
Lab 1	Configuring Trunking – Dot1q
Lab 2	Configuring Port-Channels - Manual
Lab 3	Configuring Port-Channels - LACP
Lab 4	Configuring VLAN Trunking Protocol (VTP)
Lab 5	Configuring VLANs
Lab 6	Configuring PVSTP - Root Switch Selection
Lab 7	Configuring MST
Lab 8	Configuring MST - Root Switch Selection
Lab 9	Configuring Physical-To-Logical Mapping
Lab 10	Configuring the L3 Logical Topology
Lab 11	Configuring the PortFast Feature
Lab 12	Configuring the BPDU Guard Feature
Lab 13	Configuring VLAN ACLs
Lab 14	Configuring the Root Guard Feature
Lab 15	Configuring Port-Security
<b>Module 2– Configuring EIGRP</b>	
Lab 1	Initializing EIGRP – Network Statement
Lab 2	Passive Interfaces
Lab 3	EIGRP – Unicast Neighbors
Lab 4	Metric Calculations
Lab 5	Equal & Unequal Cost Load Balancing
Lab 6	Route Summarization – Auto Summary
Lab 7	Route Summarization – Manual Summarization
Lab 8	Route Summarization – Leak Maps
Lab 9	Route Filtering using ACLS
Lab 10	Route Filtering using Prefix-Lists
Lab 11	Authenticating EIGRP Neighbors using MD5
Lab 12	Configuring a Basic Name-Mode Configuration
Lab 13	Configuring Authentication – SHA
Lab 14	Configuring Authentication – MD5
Lab 15	Configuring a Multi-Domain Network
Lab 16	Redistributing Connected & Static Routes
Lab 17	Redistributing between RIP & EIGRP
Lab 18	Redistributing between 2 EIGRP Autonomous Systems

Lab 19	Redistributing between OSPF & EIGRP
Lab 20	Redistributing with Route Filtering
Lab 21	Redistributing with Route Tagging
Lab 22	Multi-Point Redistribution with Route Tagging
Lab 23	Configuing BFD for EIGRP
<b>Module 3 – Configuring OSPF</b>	
Lab 1	Configure OSPF on Ethernet - Area 10
Lab 2	Configuring OSPF on Serial Links - Area 10
Lab 3	Configuring OSPF in Area 0
Lab 4	Configuring Unicast-based OSPF
Lab 5	Configuring an OSPF ASBR
Lab 6	Configuring a Multi-Area / Multi-Domain Topology
Lab 7	Configuring Inter-Area Route Summarization
Lab 8	Configuring External Route Summarization
Lab 9	LSA Type 3 Filtering
Lab 10	Configuring OSPF Authentication
Lab 11	Configuring OSPF Area Types
Lab 12	Configuring Virtual Link
Lab 13	Configuring BFD for OSPF
Lab 14	Configuring IP FRR - OSPF
<b>Module 4 – BGP</b>	
Lab 1	Configuring eBGP
Lab 2	Configuring eBGP Multi-Hop
Lab 3	Redistributing Networks into BGP
Lab 4	Configuring BGP Authentication
Lab 5	Configuring iBGP with Route Reflectors
Lab 6	Route Filtering using ACLs
Lab 7	Route Filtering using Prefix-Lists
Lab 8	Route Filtering using AS Path-Filter
Lab 9	Configuring Route Aggregation – Summary Only
Lab 10	Configuring Route Aggregation – Manual Filtering
Lab 11	Configuring Route Aggregation – Suppress Maps
Lab 12	Configuring Base BGP Topology – eBGP & iBGP
Lab 13	Configuring BGP Attributes – Local Preference
Lab 14	Configuring BGP Attributes – MED
Lab 15	Configuring BGP Attributes – Weight
Lab 16	Configuring BGP Attributes – AS-Path

Lab 17	Configuring BGP Attributes – No-Export Community
Lab 18	Configuring BGP Attributes – No-Advertise Community
Lab 19	Configuring BGP Conditional Advertisement
Lab 20	Configuring BGP Multi-Path – eBGP – iBGP
Lab 21	Configuring to Redistribute iBGP Routes into IGP
Lab 22	Configuring BGP Route Reflector with Next-Hop Changed
Lab 23	Configuring BGP Route Reflection based on Dynamic Neighbors
Lab 24	Working with Private AS Numbers
Lab 25	Configuring the Local-AS Command
Lab 26	Configuring BFD for BGP
Lab 27	Configuring BGP Confederations
<b>Module 5 – IPv6</b>	
Lab 1	Configuring IPv6 Addressing
Lab 2	Configuring OSPFv3
Lab 3	Configuring EIGRP for IPv6
Lab 4	Configuring IS-IS for IPv6
Lab 5	Configuring BGP for IPv6
Lab 6	Configuring IPv6IP Tunneling
Lab 7	Configuring NAT64
<b>Module 6 – Configuring Virtual Private Networks (VPNs)</b>	
Lab 1	Point-to-Point GRE
Lab 2	Encrypting GRE Tunnels Using IPsec
Lab 3	Configuring a Native IPsec Tunnel Interface
Lab 4	Configuring a mGRE VPN
Lab 5	Configuring DMVPN – Phase I
Lab 6	Configuring DMVPN – Phase II
Lab 7	Configuring DMVPN – Phase III
Lab 8	Configuring a Dual-Hub DMVPN
Lab 9	Encrypting the DMVPN Traffic Using IPsec
Lab 10	Configure Flex VPN – Point-To-Point
<b>Module 7 – Configuring MPLS Unicast Routing</b>	
Lab 1	Configuring MPLS Unicast Routing
Lab 2	Authenticating LDP Peers
Lab 3	Configuring MPLS VPN – PE-CE Using Static Routing
Lab 4	Configuring MPLS VPN – PE-CE Using EIGRP Routing
Lab 5	Configuring MPLS VPN – PE-CE Using BGP Routing – 1

Lab 6	Configuring MPLS VPN – PE-CE Using BGP Routing – 2
Lab 7	Configuring MPLS VPN – PE-CE Using OSPF
Lab 8	Configuring MPLS VPN – PE-CE Using OSPF – Domain-ID
Lab 9	Configuring MPLS VPN – PE-CE Using OSPF – Sham-link
Lab 10	Configuring MPLS VPN Extranets
<b>Module 8 – Implementing SD-WAN</b>	
Lab 1	Configuring the WAN Components
Lab 2	Installing the Enterprise Certificate Server
Lab 3	Initializing vManage – CLI
Lab 4	Initializing vManage – GUI
Lab 5	Initializing vBond – CLI
Lab 6	Initializing vBond – GUI
Lab 7	Initializing vSmart – CLI
Lab 8	Initializing vSmart – GUI
Lab 9	Initializing vEdges – CLI
Lab 10	Registering vEdges in vManage
Lab 11	Initializing cEdge – CLI
Lab 12	Registering cEdge in vManage
Lab 13	Configuring Feature Template - System
Lab 14	Configuring Feature Template - Banner
Lab 15	Configuring Feature Template – VPN & VPN Interfaces for VPN 0 & 512 – Branche Site (vEdges)
Lab 16	Configuring Feature Template – External Routing for VPN 0 – Branch Site (vEdges)
Lab 17	Configuring and Deploying Device Templates for vEdge – Branch Site (vEdge2)
Lab 18	Configuring Internal Routing Protocols on the Internal Routing Devices – HQ & All Branches
Lab 19	Configuring Feature Templates – Service VPN – VPN, VPN Interfaces & Internal Routing – Branch Site (vEdges)
Lab 20	Implementing a Service VPN using Templates – Branch Site (vEdge2)
Lab 21	Pushing Templates to configure other Branch Sites – Branch Site (vEdge3 & vEdge4)
Lab 22	Configuring Feature Templates for HQ-Site (vEdge1) – VPNs, VPN Interfaces, External & Internal Routing
Lab 23	Configuring Device Templates for HQ-Site (vEdge1) to deploy VPN 0, VPN 1 & VPN 512

Lab 24	Configuring Feature Templates for CSR – VPNs, VPN Interfaces, External & Internal Routing
Lab 25	Configure Device Templates for CSR to deploy VPN 0, 1 & 512
Lab 26	Configuring and Deploying Feature and Device Templates for vSmart Controllers
Lab 27	Configuring Application Aware Policies using Telnet & Web
Lab 28	Configuring Application Aware Policies using Chat Applications
Lab 29	Manipulating Traffic Flow using TLOCs
Lab 30	Configure Route Filtering
Lab 31	Configuring the WAN Components
Lab 32	Configuring Firewalls to support SD-WAN
Lab 33	Configure vEdges with NAT thru Firewall
Lab 34	Configuring TLOC Extensions
Lab 35	Load Balancing using Multiple vEdges
Lab 36	Configuring a Hub-n-Spoke Topology using a TLOC
Lab 37	Implementing QoS – Configuring Custom Options
Lab 38	Implementing QoS – Configuring the Scheduler
Lab 39	Implementing QoS – Configure & apply the Localized Policies
Lab 40	Implementing QoS – Configure the Interface Parameters using Templates
<b>Module 9 – Implementing SDA</b>	
Lab 1	Configuring DNAC & ISE Integration
Lab 2	Configuring Border Switch Initial Configuration
Lab 3	Configuring Fusion Router Initial Configuration
Lab 4	DNAC Design – Network Hierarchy – Site & Building
Lab 5	DNAC Design – Server Configuration – AAA, NTP & DHCP
Lab 6	DNAC Design – Device Credentials
Lab 7	DNAC Design – IP Address Pools
Lab 8	Manual Underlay Configuration – Fabric Skinny Configuration
Lab 9	Manual Underlay Configuration – Configuring IGP - OSPF
Lab 10	Manual Underlay Configuration – Device Discovery & Provisioning
Lab 11	LAN Automation – Seed Device Configuration & Discovery
Lab 12	LAN Automation – Seed Device Assignment

Lab 13	LAN Automation – Implementing LAN Automation
Lab 14	LAN Automation – Provisioning the devices to HQ Site
Lab 15	Reserve the IP Pools for HQ Site for Overlay & Underlay
Lab 16	Create the VNs for the Fabric
Lab 17	Create the Transit Network (L3 Handoff)
Lab 18	Configuring Host Onboarding
Lab 19	Configuring & Provisioning the Control / Border Devices
Lab 20	Configuring & Provisioning the Fabric Edge Devices
Lab 21	Configure the Fusion Router – VRF, SVI, BGP & Route Leaking
Lab 22	Configure User & Groups on ISE
Lab 23	Configure Authorization Profiles for the DNAC VNs
Lab 24	Configure Authorization Policies for the DNAC VNs
Lab 25	Configure the DHCP Server to provide IP Configuration to Clients
Lab 26	Verifying Macro Segmentation
Lab 27	Micro Segmentation – Creating SGTs
Lab 28	Micro Segmentation – Assigning SGTs via Authorization Policies on ISE
Lab 29	Micro Segmentation – Using Default Contract to Block all communications between SGTs
Lab 30	Micro Segmentation – Creating a SG ACL - Contract
Lab 31	Micro Segmentation – Applying & Verifying a Custom SG-ACL - Contract
Lab 32	Configuring L2 Handoff
Lab 33	Configuring Templates
<b>Module 11 – IP Services &amp; Security</b>	
Lab 1	Configuring Zone-Based Firewall
Lab 2	Configuring FHRP - HSRP
Lab 3	Configuring FHRP - VRRP
Lab 4	Configuring DHCP Server
Lab 5	Configuring DHCP Relay Agent
Lab 6	Configuring DHCP Snooping
Lab 7	Configuring NTP
Lab 8	Configuring AAA Services
Lab 9	Configuring IP SLA
Lab 10	Configuring Dynamic NAT
Lab 11	Configuring Dynamic PAT

Lab 12	Configuring Static NAT
Lab 13	Configuring Static PAT
<b>Module 12 - Configuring Quality of Service (QoS)</b>	
Lab 1	Configuring Policing
Lab 2	Configuring Congestion Management with Bandwidth Reservation
Lab 3	Configuring Congestion Management with Low-Latency Queuing (LLQ)
Lab 4	Classifying Traffic Using NBAR
Lab 5	Configuring Shaping
Lab 6	Configuring Advanced Class Maps
<b>Module 13 - Configuring Multicast Routing</b>	
Lab 1	Configuring PIM - Dense Mode
Lab 2	Configuring PIM - Sparse-Mode using Single Static RP
Lab 3	Configuring PIM - Sparse-Mode using Multiple Static RP
Lab 4	Configuring PIM - Sparse-Mode using Dense-Mode for Fallback
Lab 5	Configuring PIM - Sparse Mode - Auto RP
Lab 6	Configuring PIM - Sparse Mode - BSR
Lab 7	Configuring MSDP
<b>Module 14 - Automation &amp; Programmability</b>	
Lab 1	Configuring EEM - Controlling Interface Shutdown
Lab 2	Configuring EEM - E-Mailing Errors to Administrators
Lab 3	Retrieving Information from Routers Using Python - Interactive
Lab 4	Configuring Network Devices Using Python
Lab 5	Configuring Network Devices Using Python - Interactive
Lab 6	Configuring Network Devices Using Python - Interactive Login & Configuration
Lab 7	Initialize the Router using a Python Script - Using Netmiko
Lab 8	Initialize the Router using a Python Script - Using Netmiko (Interactive)
Lab 9	Retrieving Information from Multiple Routers - Using Netmiko
Lab 10	Backing up Configuration of a single Router - Using Netmiko
Lab 11	Backing up Configuration of Multiple Routers - Using Netmiko



<b>Lab 12</b>	Configuring Multiple Devices – Netmiko Library
<b>Lab 13</b>	Configuring Multiple Devices – Netmiko Library (Interactive)

# Configuring Layer 2 Technologies – Ethernet Switching

**Authored By:**

**Khawar Butt**

CCIE # 12353

Hepta CCIE#12353

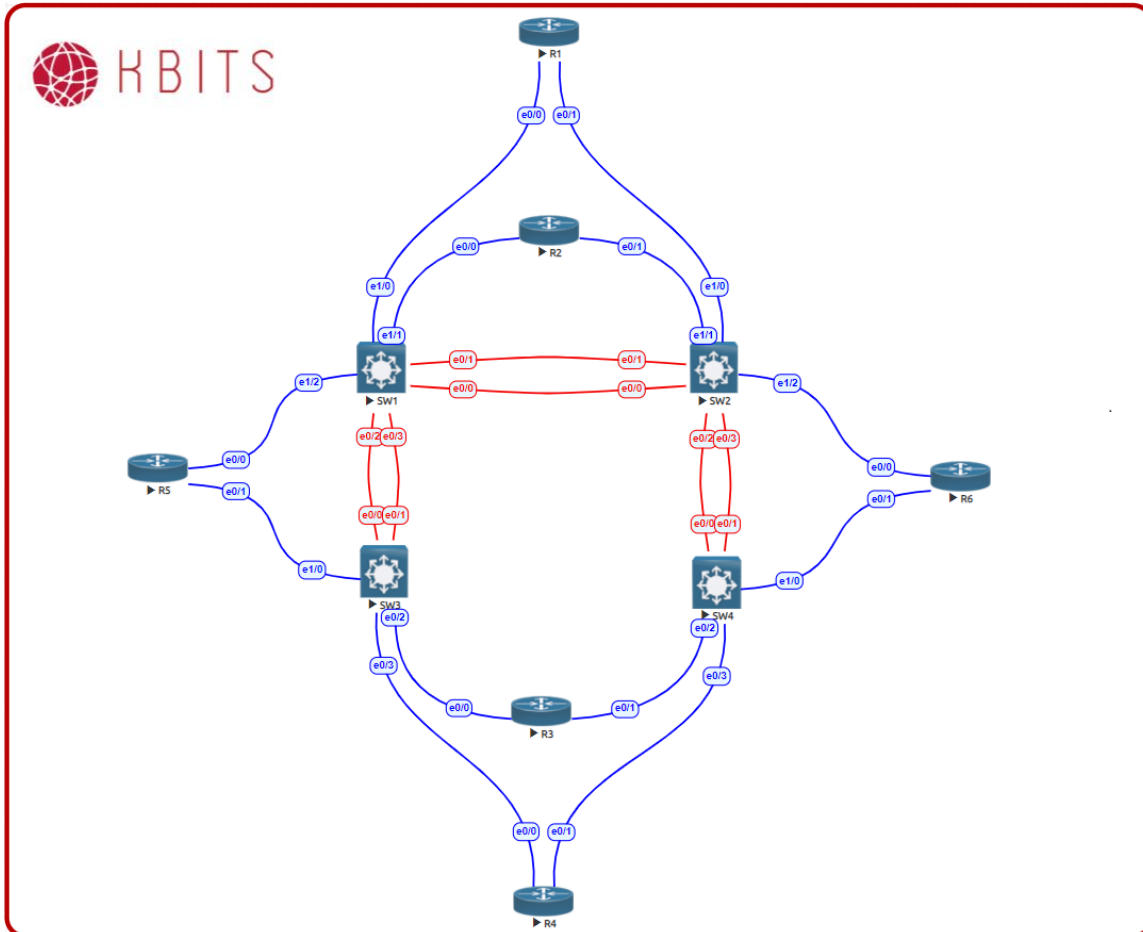
CCDE # 20110020

**Configuring Layer 2 Technologies –  
Ethernet Switching**



# Lab 1 – Configuring Trunking – Dot1q

## Physical Diagram



## Task 1 – Configure Trunking between SW1 & SW3

- Configure the Links between SW1 & SW3 as trunks.
- SW1 ports E 0/2 & E 0/3 are connected to SW3 ports E 0/0 & E 0/1.
- Use Dot1q as the Trunk encapsulation mechanism.

### SW1

```
Hostname SW1
!  
Interface range E 0/2-3  
  switchport trunk encapsulation dot1q  
  switchport mode trunk
```

### SW3

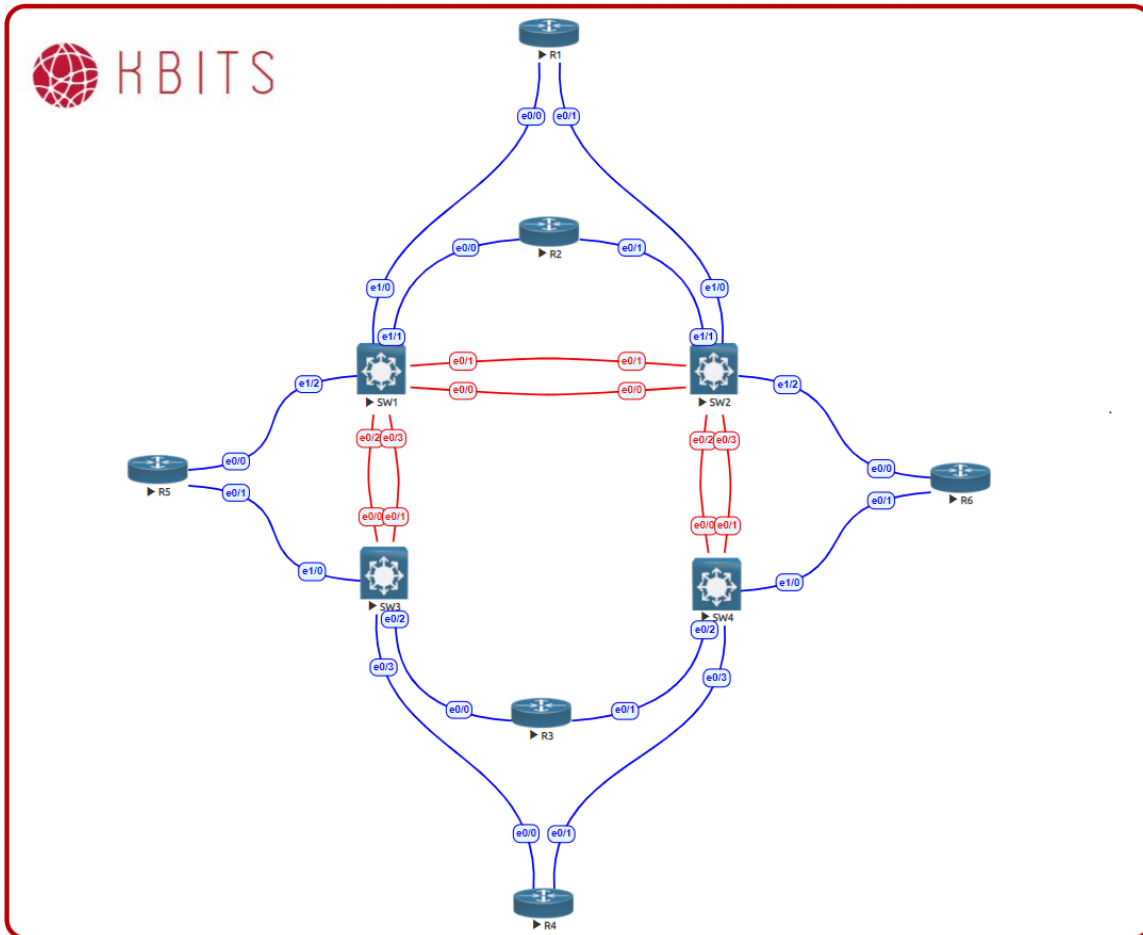
```
Hostname SW3
!  
Interface range E 0/0-1  
  switchport trunk encapsulation dot1q  
  switchport mode trunk
```

## Task 2 – Verification

- Verify the Spanning-tree status on SW1 & SW3 by using the “**Show Spanning-tree**” command.
- What is the status of the ports on the Root Bridge?
- What is the status of the ports on the non-root bridge?

# Lab 2 – Configuring Port Channels – Manual

## Physical Diagram



## Task 1 – Configure Port-Channels between SW1 & SW2

- Configure a Port-Channels between SW1 & SW2 using ports E 0/0 & E0/1 on both switches.
- Use the manual Port-channel setup mechanism.

### SW1

```
Interface range E 0/0-1
channel-group 12 mode on
no shut
!
Interface port-channel 12
switchport trunk encapsulation dot1q
switchport mode trunk
```

### SW2

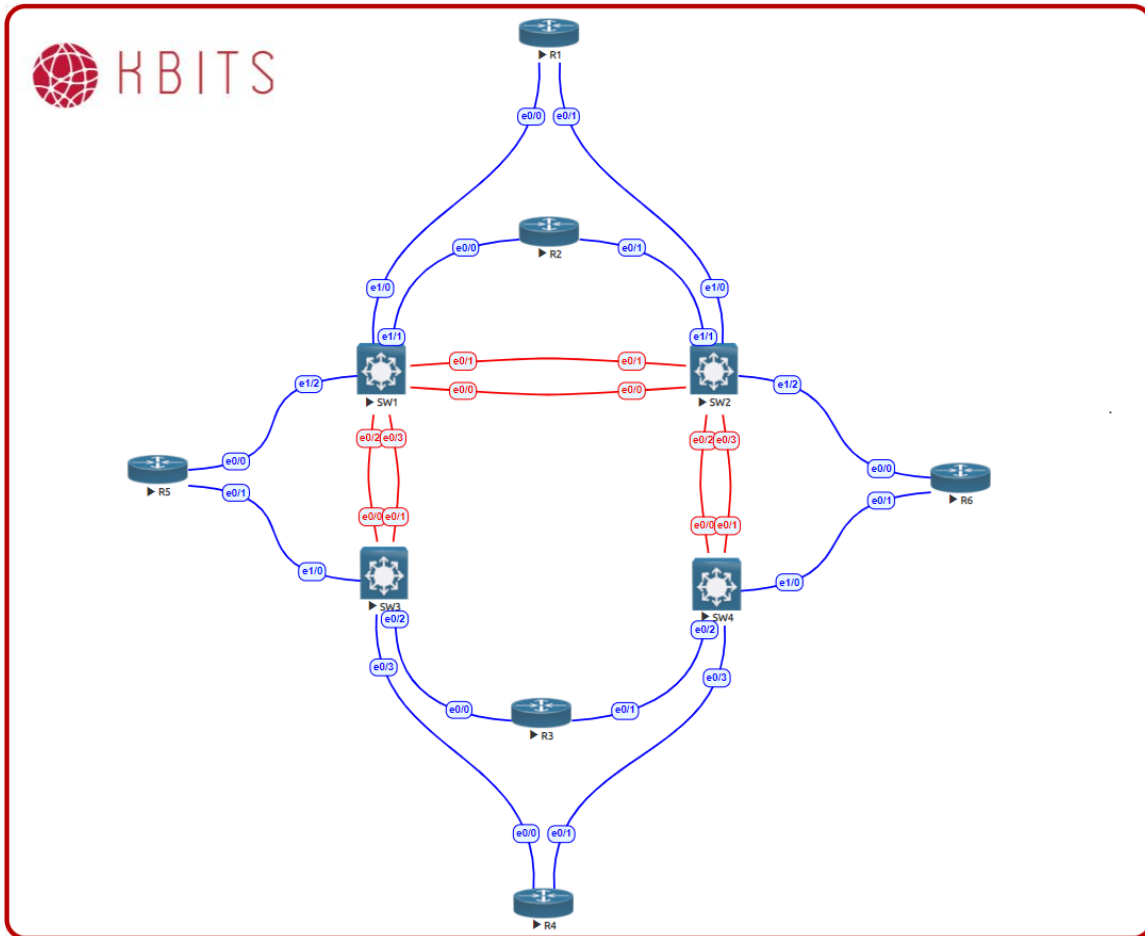
```
Interface range E 0/0-1
channel-group 12 mode on
no shut
!
Interface port-channel 12
switchport trunk encapsulation dot1q
switchport mode trunk
```

## Task 2 – Verification

- Verify that the port channel is operational by using the "**Show etherchannel summary**" command.

# Lab 3 – Configuring Port Channels – LACP

## Physical Diagram



### Task 1 – Configure Port-Channels between SW2 & SW4

- Configure a Port-Channels between SW2 & SW4 using ports E 0/0 & E0/1 on SW4 and ports E 0/2 & E 0/3 on SW2.
- Use the LACP as the Port-channel negotiation protocol.
- Both switches should be able to initiate the negotiation.

#### SW2

```
Interface range E 0/2-3
channel-group 24 mode active
no shut
!
Interface port-channel 24
switchport trunk encapsulation dot1q
switchport mode trunk
```

#### SW4

```
Interface range E 0/0-1
channel-group 24 mode active
no shut
!
Interface port-channel 24
switchport trunk encapsulation dot1q
switchport mode trunk
```

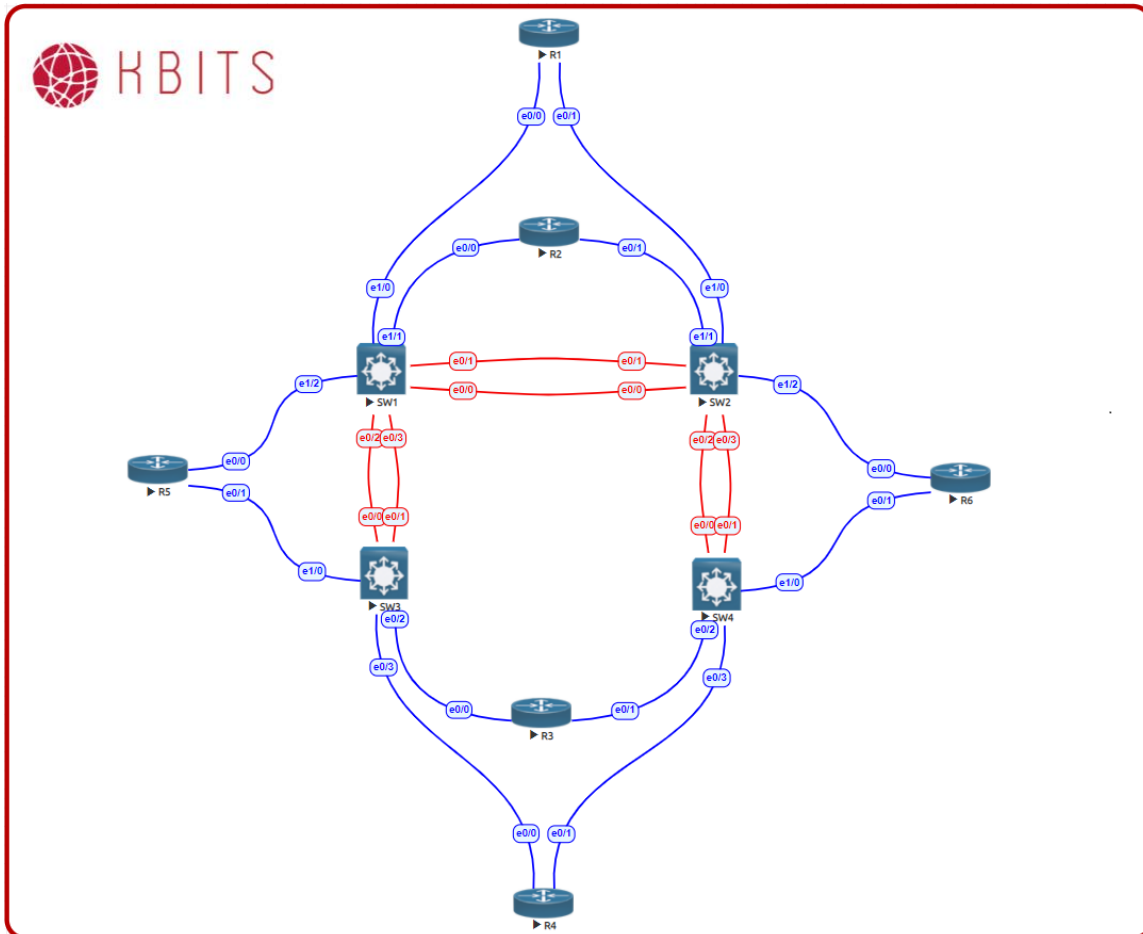
### Task 2 – Verification

- Verify that the port channel is operational by using the "**Show etherchannel summary**" command.



# Lab 4 – Configuring VLAN Trunking Protocol (VTP)

## Physical Diagram



## Task 1 – Configure SW1 as VTP Server

- Configure SW1 as the VTP Server in a Domain called KBITS.
- Configure it with VTP v2.
- Configure a password of kbits@123.

### SW1

```
vtp mode server  
vtp domain KBITS  
vtp version 2  
vtp password kbits@123
```

## Task 2 – Configure the other switches as VTP clients

- Configure SW2, SW3 & SW4 as the VTP Clients in a Domain called KBITS.
- Configure is with VTP v2.
- Configure a password of kbits@123.

### SW2

```
vtp domain KBITS  
vtp version 2  
vtp password kbits@123  
vtp mode client
```

### SW3

```
vtp domain KBITS  
vtp version 2  
vtp password kbits@123  
vtp mode client
```

### SW4

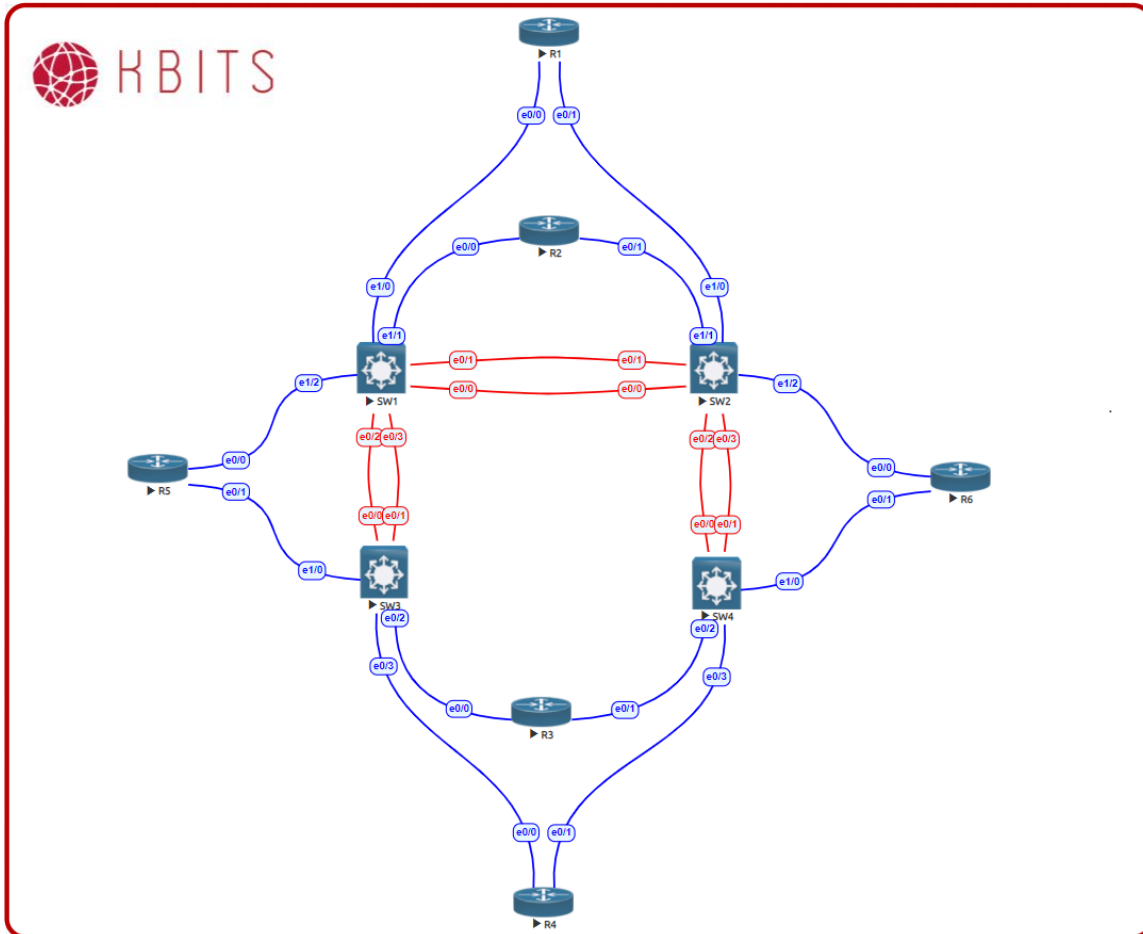
```
vtp domain KBITS  
vtp version 2  
vtp password kbits@123  
vtp mode client
```

## Task 3 – Verification

- Verify the VTP Status on the devices using the "**Show vtp status**" command.

# Lab 5 – Configuring VLAN

## Physical Diagram



### **Task 1 - Configure VLANs on the VTP Server**

- Configure the following VLANs: 10, 20, 30, 40, 50, 60, 70 & 80.

**SW1**

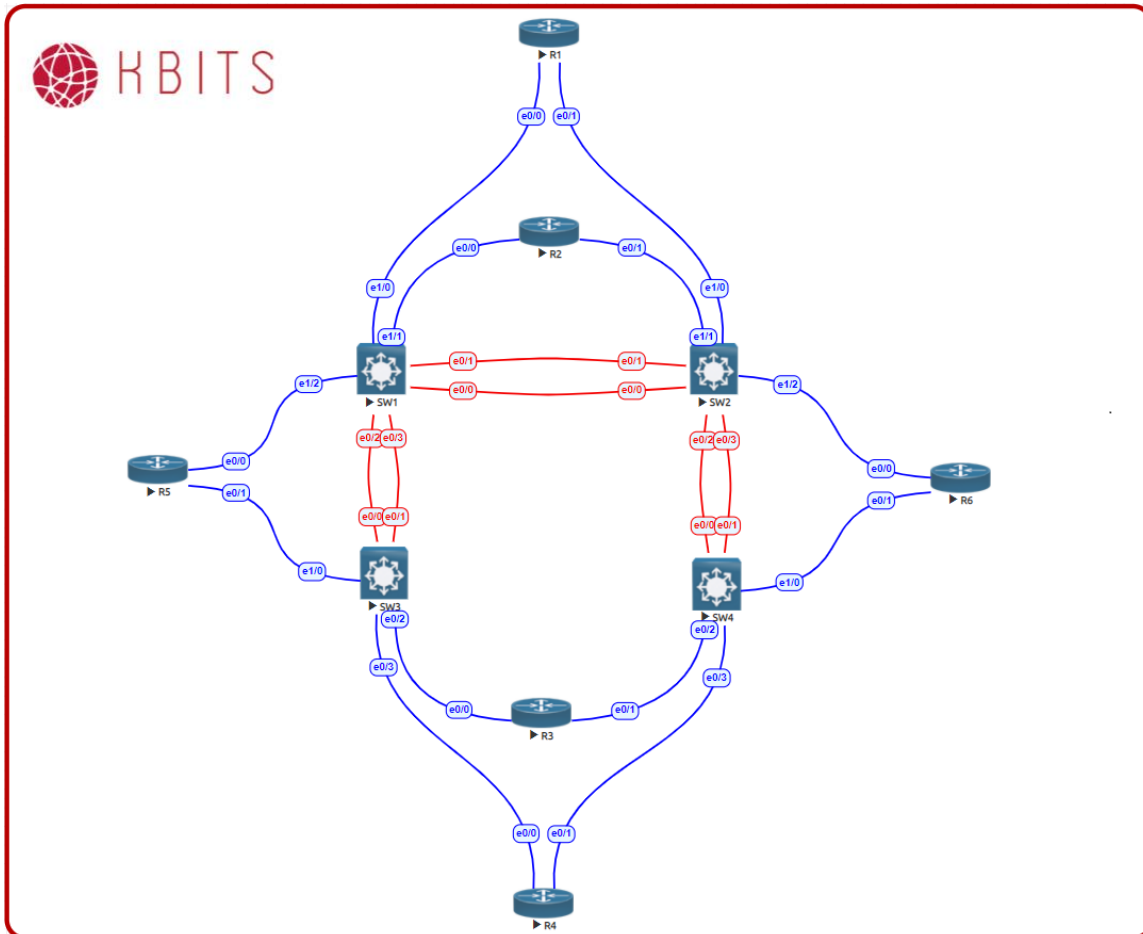
VLAN 10,20,30,40,50,60,70,80

### **Task 2 - Verification**

- Verify the creation of the VLANs on the VTP Clients.

# Lab 6 – Configuring PVSTP – Root Switch Selection

## Physical Diagram



### Task 1 – Configure Root Bridge selection for VLANs 1, 10, 20, 30 & 40

- Configure SW1 as the preferred Root Switch for VLANs 1,10,20,30,40 with SW2 as the backup Root Switch.
- Do not use the "Root Primary" or "Root Secondary" option to accomplish this step.

#### SW1

```
Spanning-tree vlan 1,10,20,30,40 priority 0
```

#### SW2

```
Spanning-tree vlan 1,10,20,30,40 priority 4096
```

### Task 2 – Configure Root Bridge selection for VLANs 50, 60, 70 & 80

- Configure SW2 as the preferred Root Switch for VLANs 50,60,70,80 with SW1 as the backup Root Switch.
- Do not use the Priority command to accomplish this task.

#### SW2

```
spanning-tree vlan 50,60,70,80 root primary
```

#### SW1

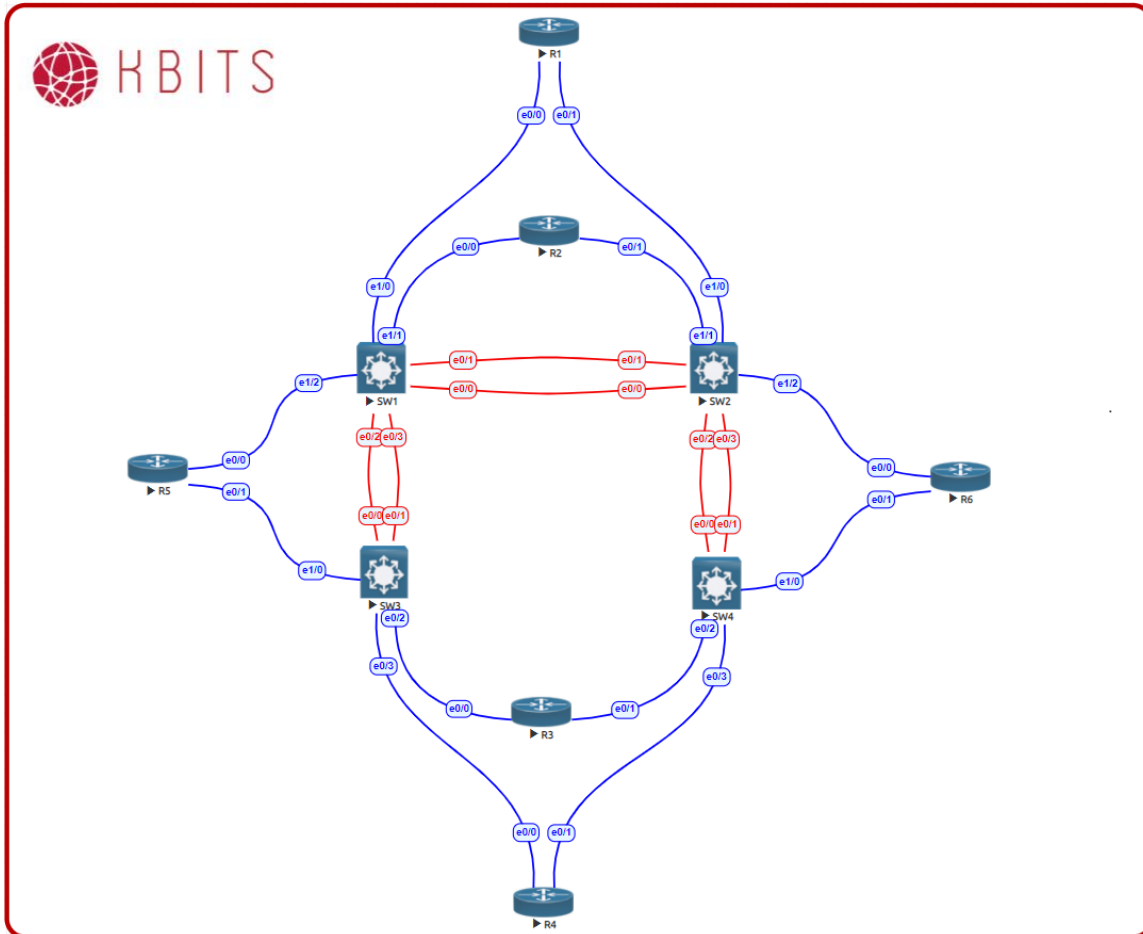
```
spanning-tree vlan 50,60,70,80 root secondary
```

### Task 3 – Verification

- Verify the Root Bridge selection by using the “**Show spanning-tree vlan 10**” on SW1. It should be the root bridge.
- Verify the Root Bridge selection by using the “**Show spanning-tree vlan 50**” on SW1. It should be using the Port-channel towards SW2 as the root port.
- Verify the Root Bridge selection by using the “**Show spanning-tree vlan 50**” on SW2. It should be the root bridge.
- Verify the Root Bridge selection by using the “**Show spanning-tree vlan 10**” on SW2. It should be using the Port-channel towards SW1 as the root port.

# Lab 7 – Configuring MST

## Physical Diagram





## Task 1 – Configure Root Bridge selection for VLANs 1, 10, 20, 30 & 40

- Configure the switches to run MSTP.
- MSTP name should be configured as "CCIE-EI" with a version of 1.
- VLANs 10,20,30,40 should be in instance 1.
- VLANs 50,60,70,80 should be in instance 2.

### SW1

```
spanning-tree mode mst
spanning-tree mst configuration
name CCIE-EI
revision 1
instance 1 vlan 10,20,30,40
instance 2 vlan 50,60,70,80
```

### SW2

```
spanning-tree mode mst
spanning-tree mst configuration
name CCIE-EI
revision 1
instance 1 vlan 10,20,30,40
instance 2 vlan 50,60,70,80
```

### SW3

```
spanning-tree mode mst
spanning-tree mst configuration
name CCIE-EI
revision 1
instance 1 vlan 10,20,30,40
instance 2 vlan 50,60,70,80
```

### SW4

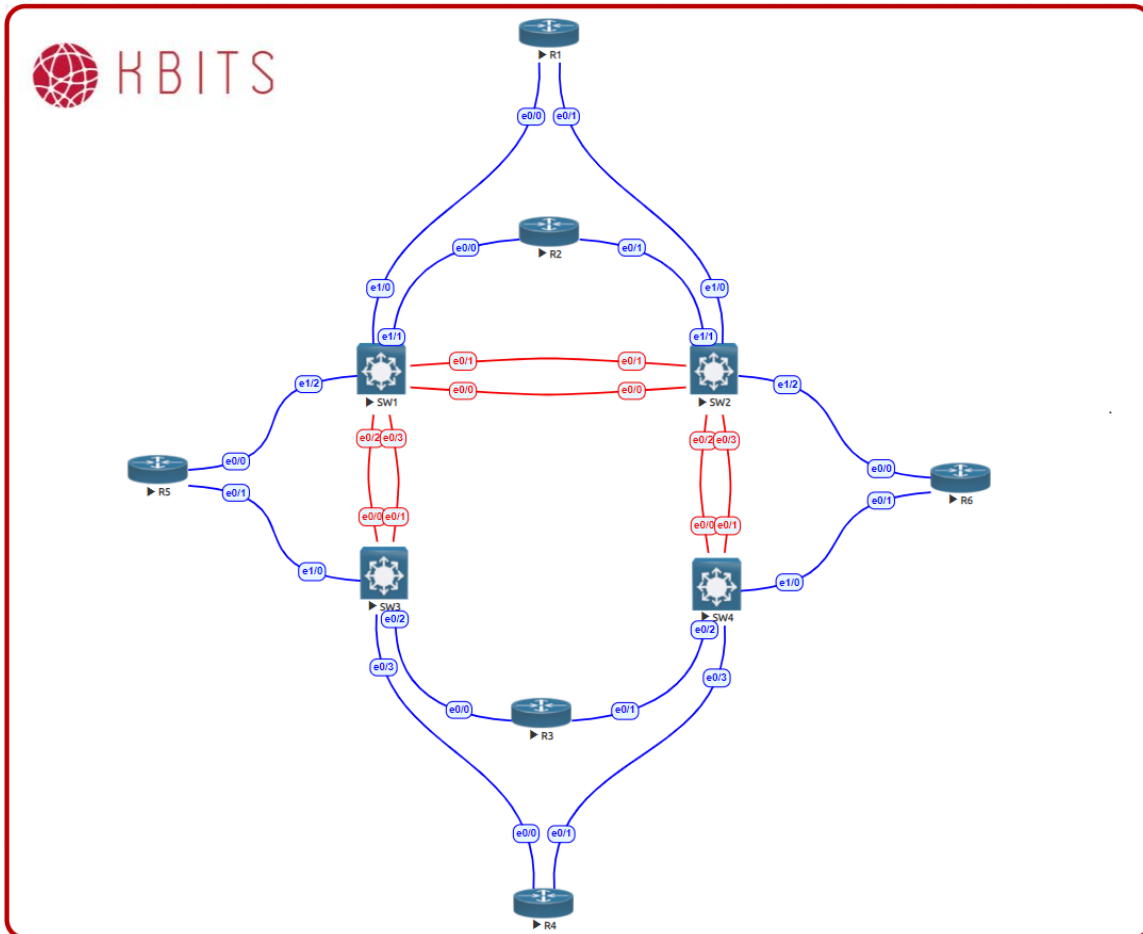
```
spanning-tree mode mst
spanning-tree mst configuration
name CCIE-EI
revision 1
instance 1 vlan 10,20,30,40
instance 2 vlan 50,60,70,80
```

## Task 2 – Verification

- Verify the Spanning-tree status by using the “**Show spanning-tree**” on the Switches. What is the Spanning-tree mode set to?

# Lab 8 – Configuring MST – Root Switch Selection

## Physical Diagram



### Task 1 – Configure Root Bridge selection for MST 1

- Configure SW1 as the preferred Root Switch for MST 1 with SW2 as the backup Root Switch.
- Do not use the "Root Primary" or "Root Secondary" option to accomplish this step.

#### SW1

Spanning-tree mst 1 priority 0

#### SW2

Spanning-tree mst 1 priority 4096

### Task 2 – Configure Root Bridge selection for MST 2

- Configure SW2 as the preferred Root Switch for MST 2 with SW1 as the backup Root Switch.
- Do not use the Priority command to accomplish this task.

#### SW2

spanning-tree mst 2 root primary

#### SW1

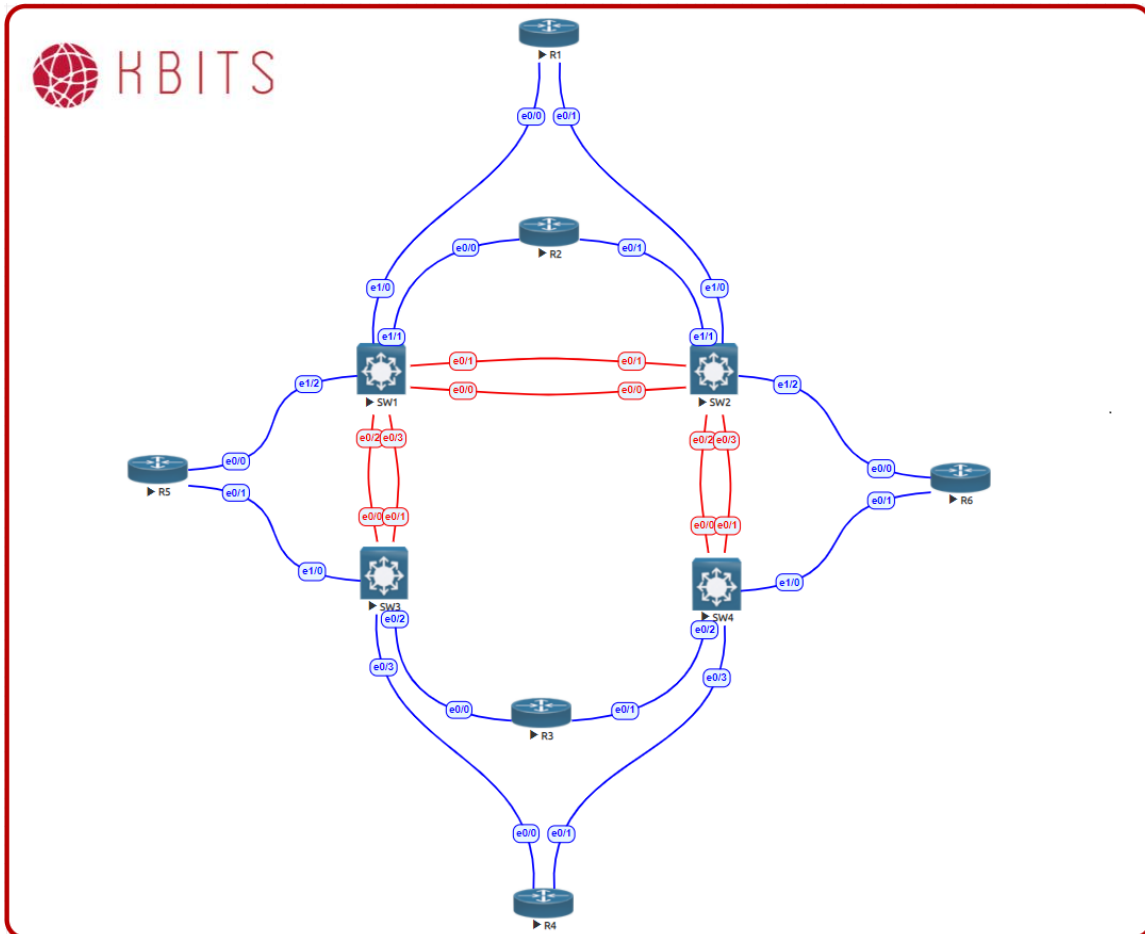
spanning-tree mst 2 root secondary

### Task 3 – Verification

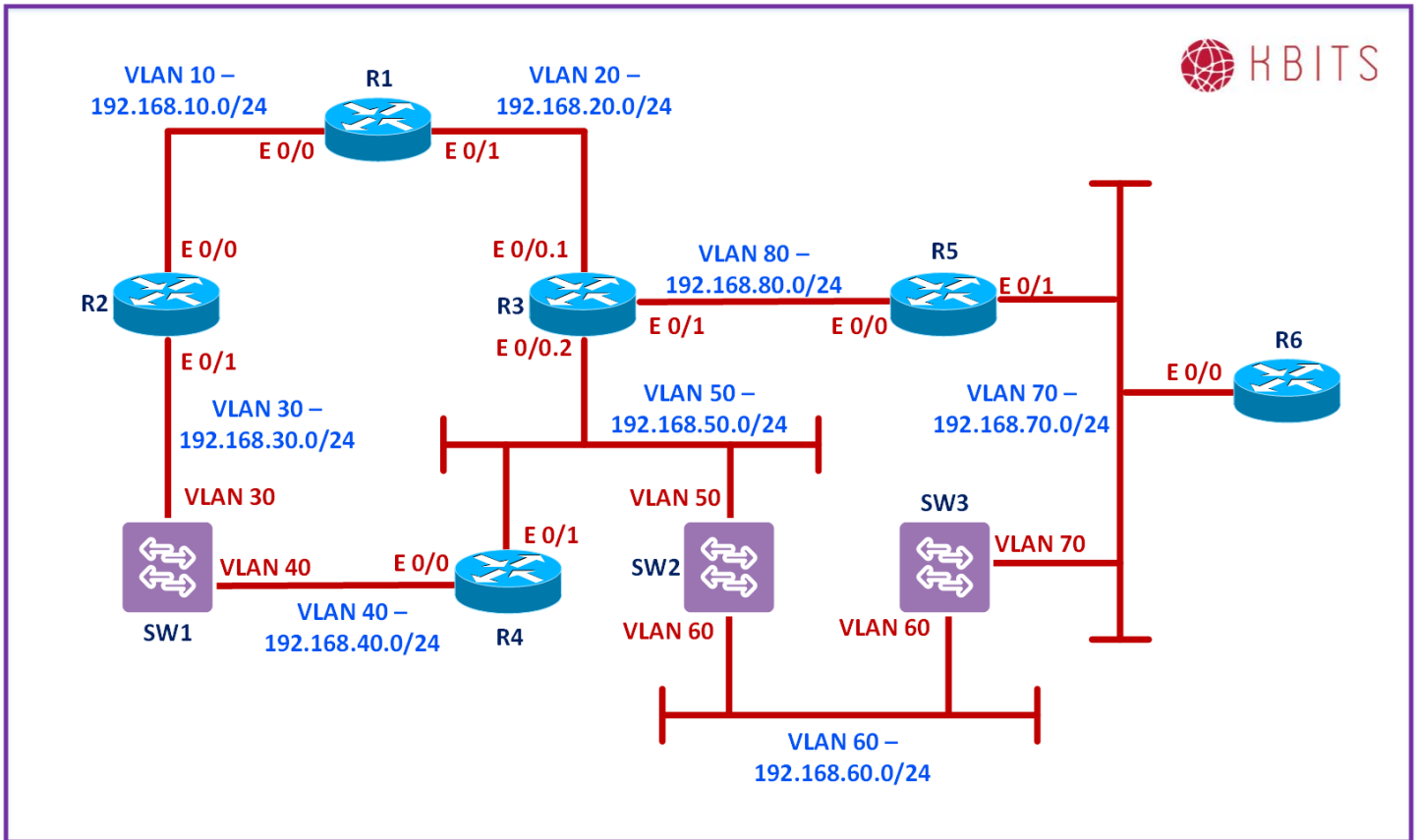
- Verify the Root Bridge selection by using the “**Show spanning-tree MST 1**” on SW1. It should be the root bridge.
- Verify the Root Bridge selection by using the “**Show spanning-tree MST 2**” on SW1. It should be using the Port-channel towards SW2 as the root port.
- Verify the Root Bridge selection by using the “**Show spanning-tree MST 2**” on SW2. It should be the root bridge.
- Verify the Root Bridge selection by using the “**Show spanning-tree MST 1**” on SW2. It should be using the Port-channel towards SW1 as the root port.

# Lab 9 – Configuring Physical-To-Logical Mapping

## Physical Diagram



# Logical Diagram



### **Task 1 – Assign ports to VLAN 10 based on the Logical and Physical Diagram**

- R1 & R2 physical ports are in VLAN 10.
- Assign the corresponding Switchports to VLAN 10 as Access Ports.

#### **SW1**

```
Interface range e 1/0-1
switchport mode access
switchport access vlan 10
```

### **Task 2 – Assign ports to VLAN 20 based on the Logical and Physical Diagram**

- R1 has a physical port are in VLAN 20.
- Assign the corresponding Switchport to VLAN 20 as an Access Port.
- R3 has a sub-interface in VLAN 20.
- Configure the switchport for R3 as a Trunk as the corresponding port is a Sub-interface.

#### **SW2**

```
Interface e 1/0
switchport mode access
switchport access vlan 20
```

#### **SW3**

```
Interface e 0/2
switchport trunk encapsulation dot1q
switchport mode trunk
```

### Task 3 – Assign ports to VLAN 30 based on the Logical and Physical Diagram

- R2 has a physical port are in VLAN 30.
- Assign the corresponding Switchport for R2 to VLAN 30 as an Access Port.
- SW1 has a L3 connection in VLAN 30. Use a SVI to connect to VLAN 30. Configure SW1 with an IP of .11.

#### SW2

```
Interface e 1/1
switchport mode access
switchport access vlan 30
```

#### SW1

```
ip routing
!
interface vlan 30
ip address 192.168.30.11 255.255.255.0
no shut
```

### Task 4 – Assign ports to VLAN 40 based on the Logical and Physical Diagram

- R4 has a physical port are in VLAN 40.
- Assign the corresponding Switchport for R4 to VLAN 40 as an Access Port.
- SW1 has a L3 connection in VLAN 40. Use a SVI to connect to VLAN 40. Configure SW1 with an IP of .11.

#### SW3

```
Interface e 0/3
switchport mode access
switchport access vlan 40
```

#### SW1

```
ip routing
!
interface vlan 40
ip address 192.168.40.11 255.255.255.0
no shut
```



### **Task 5 – Assign ports to VLAN 50 based on the Logical and Physical Diagram**

- R4 has a physical port are in VLAN 50.
- Assign the corresponding Switchport for R4 to VLAN 50 as an Access Port.
- SW2 has a L3 connection in VLAN 50. Use a SVI to connect to VLAN 50. Configure SW2 with an IP of .22.
- R3 has a sub-interface in VLAN 50. The switchport for R3 is already configured as a Trunk.

#### **SW4**

```
Interface e 0/3
switchport mode access
switchport access vlan 50
```

#### **SW2**

```
ip routing
!
interface vlan 50
ip address 192.168.50.22 255.255.255.0
no shut
```

### **Task 6 – Assign ports to VLAN 60 based on the Logical and Physical Diagram**

- SW2 & SW3 have a L3 connection in VLAN 60. Use a SVI to connect the switches to VLAN 60. Configure SW2 with an IP of .22 and SW3 of .33.

#### **SW2**

```
ip routing
!
interface vlan 60
ip address 192.168.60.22 255.255.255.0
no shut
```

#### **SW3**

```
ip routing
!
interface vlan 60
ip address 192.168.60.33 255.255.255.0
no shut
```

### Task 7 – Assign ports to VLAN 70 based on the Logical and Physical Diagram

- R5 & R6 have physical ports are in VLAN 70.
- Assign the corresponding Switchport for R5 & R6 to VLAN 70 as Access Ports.
- SW3 has a L3 connection in VLAN 70. Use a SVI to connect to VLAN 70. Configure SW3 with an IP of .33.

#### SW3

```
Interface e 1/0
switchport mode access
switchport access vlan 70
```

#### SW2

```
Interface e 1/2
switchport mode access
switchport access vlan 70
```

#### SW3

```
ip routing
!
interface vlan 70
ip address 192.168.70.33 255.255.255.0
no shut
```

### Task 8 – Assign ports to VLAN 80 based on the Logical and Physical Diagram

- R3 & R5 have physical ports are in VLAN 80.
- Assign the corresponding Switchports for R3 & R5 to VLAN 80 as Access Ports.

#### SW1

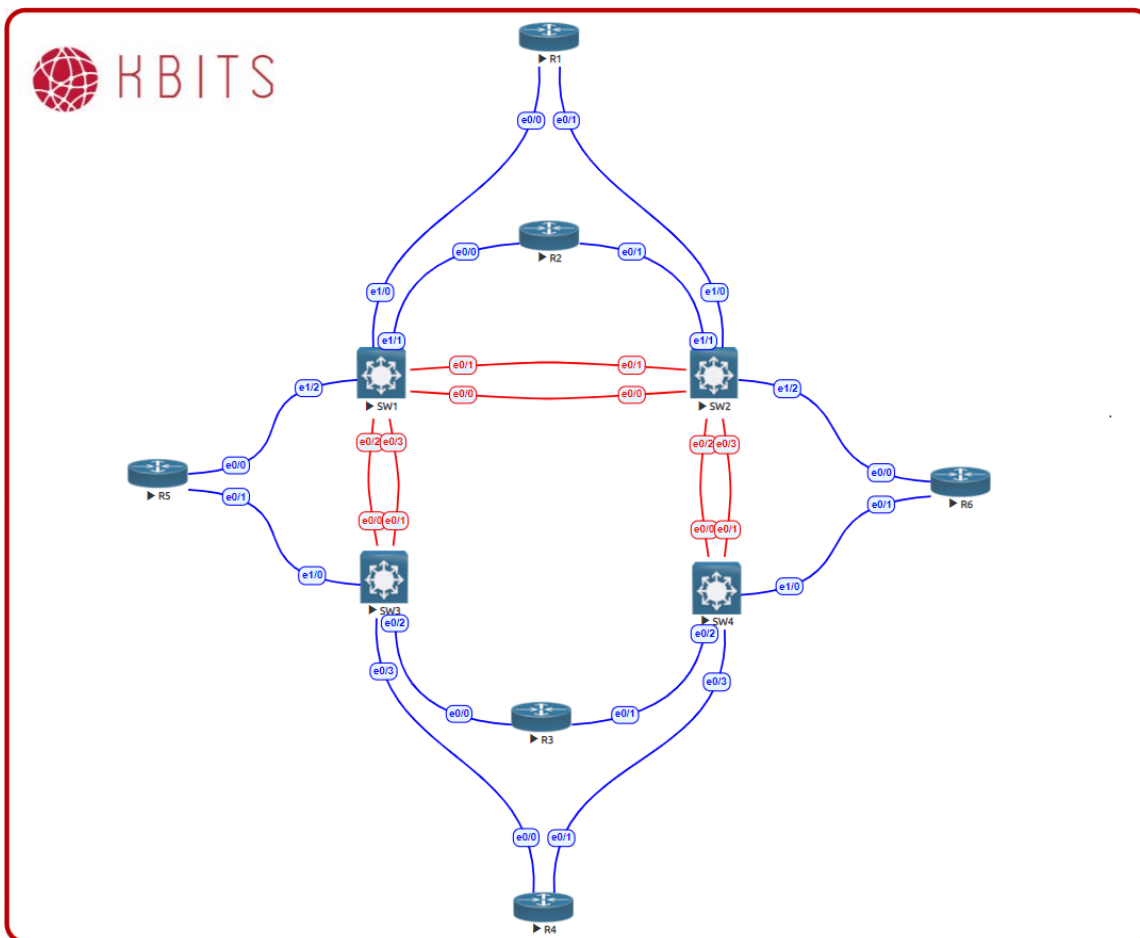
```
Interface e 1/2
switchport mode access
switchport access vlan 80
```

#### SW4

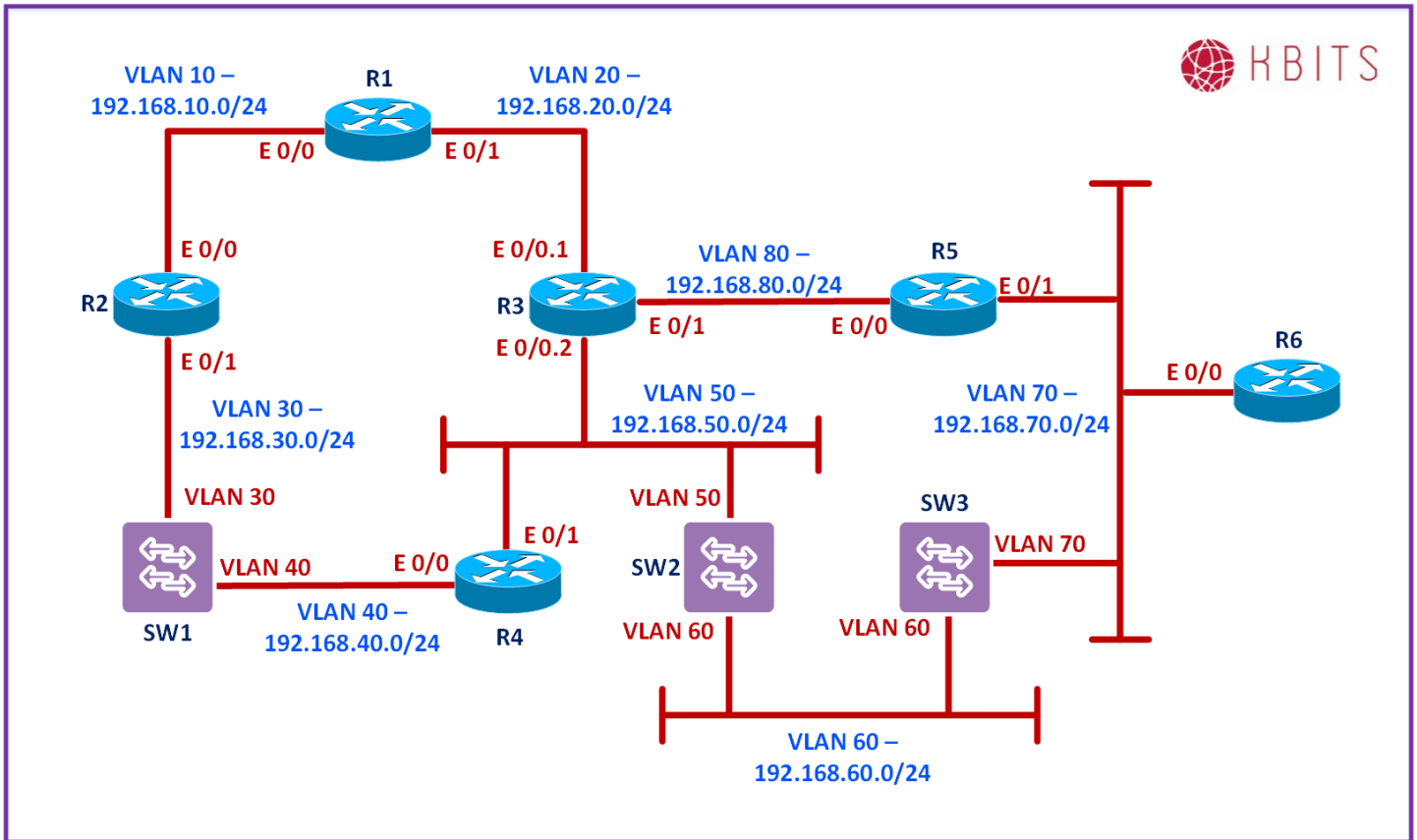
```
Interface e 0/2
switchport mode access
switchport access vlan 80
```

# Lab 10 – Configuring the L3 Logical Topology

## Physical Diagram



## Logical Diagram



### Interface Configuration

#### R1

Interface	IP Address	Subnet Mask
E 0/0	192.168.10.1	255.255.255.0
E 0/1	192.168.20.1	255.255.255.0
Loopback0	1.1.1.1	255.0.0.0

#### R2

Interface	IP Address	Subnet Mask
E 0/0	192.168.10.2	255.255.255.0
E 0/1	192.168.30.2	255.255.255.0
Loopback0	2.2.2.2	255.0.0.0

### R3

Interface	IP Address	Subnet Mask
E 0/0.1	192.168.20.3	255.255.255.0
E 0/0.2	192.168.50.3	255.255.255.0
E 0/1	192.168.80.3	255.255.255.0
Loopback0	3.3.3.3	255.0.0.0

### R4

Interface	IP Address	Subnet Mask
E 0/0	192.168.40.4	255.255.255.0
E 0/1	192.168.50.4	255.255.255.0
Loopback0	4.4.4.4	255.0.0.0

### R5

Interface	IP Address	Subnet Mask
E 0/0	192.168.80.2	255.255.255.0
E 0/1	192.168.70.2	255.255.255.0
Loopback0	5.5.5.5	255.0.0.0

### R6

Interface	IP Address	Subnet Mask
E 0/0	192.168.70.6	255.255.255.0
Loopback0	6.6.6.6	255.0.0.0

### SW1

Interface	IP Address	Subnet Mask
VLAN 30	192.168.30.11	255.255.255.0
VLAN 40	192.168.40.11	255.255.255.0
Loopback0	11.11.11.11	255.0.0.0

### SW2

Interface	IP Address	Subnet Mask
VLAN 50	192.168.50.22	255.255.255.0
VLAN 60	192.168.60.22	255.255.255.0
Loopback0	22.22.22.22	255.0.0.0

### SW3

Interface	IP Address	Subnet Mask
VLAN 60	192.168.60.33	255.255.255.0
VLAN 70	192.168.70.33	255.255.255.0
Loopback0	33.33.33.33	255.0.0.0

### Task 1 – Assign ports IP Addresses to L3 Devices

- Assign IP Addresses to the Devices based on the above table.
- SVIs on the Switches have been configured in the previous lab.

#### R1

```
Interface E 0/0
ip address 192.168.10.1 255.255.255.0
duplex full
no shut
Interface E 0/1
ip address 192.168.20.1 255.255.255.0
duplex full
no shut
Interface Loopback0
ip address 1.1.1.1 255.0.0.0
```

#### R2

```
Interface E 0/0
ip address 192.168.10.2 255.255.255.0
duplex full
no shut
Interface E 0/1
ip address 192.168.30.2 255.255.255.0
duplex full
no shut
Interface Loopback0
ip address 2.2.2.2 255.0.0.0
```

#### R3

```
Interface E 0/0
no shut
duplex full
Interface E0/0.1
encapsulation dot1q 20
ip address 192.168.20.3 255.255.255.0
```

```
Interface E0/0.2
encapsulation dot1q 50
ip address 192.168.50.3 255.255.255.0
!
Interface E 0/1
ip address 192.168.80.3 255.255.255.0
duplex full
no shut
Interface Loopback0
ip address 3.3.3.3 255.0.0.0
```

#### **R4**

```
Interface E 0/0
ip address 192.168.40.4 255.255.255.0
duplex full
no shut
Interface E 0/1
ip address 192.168.50.4 255.255.255.0
duplex full
no shut
Interface Loopback0
ip address 4.4.4.4 255.0.0.0
```

#### **R5**

```
Interface E 0/0
ip address 192.168.80.5 255.255.255.0
duplex full
no shut
Interface E 0/1
ip address 192.168.70.5 255.255.255.0
duplex full
no shut
Interface Loopback0
ip address 5.5.5.5 255.0.0.0
```

#### **R6**

```
Interface E 0/0
ip address 192.168.70.6 255.255.255.0
duplex full
no shut
Interface Loopback0
ip address 6.6.6.6 255.0.0.0
```

#### **SW1**

```
interface loop0
```

```
ip address 11.11.11.11 255.0.0.0
```

### **SW2**

```
interface loop0  
ip address 22.22.22.22 255.0.0.0
```

### **SW3**

```
interface loop0  
ip address 33.33.33.33 255.0.0.0
```

## **Task 2 – Configure EIGRP as the Routing Protocol**

- Configure EIGRP as the Routing Protocol to provide full connectivity.
- Use 100 as the Autonomous System.

### **R1**

```
router eigrp 100  
network 192.168.10.0  
network 192.168.20.0  
network 1.0.0.0
```

### **R2**

```
router eigrp 100  
network 192.168.10.0  
network 192.168.30.0  
network 2.0.0.0
```

### **R3**

```
router eigrp 100  
network 192.168.20.0  
network 192.168.50.0  
network 192.168.80.0  
network 3.0.0.0
```

### **R4**

```
router eigrp 100  
network 192.168.40.0  
network 192.168.50.0  
network 4.0.0.0
```

### **R5**

```
router eigrp 100
```



```
network 192.168.70.0
network 192.168.80.0
network 5.0.0.0
```

### **R6**

```
router eigrp 100
network 192.168.70.0
network 6.0.0.0
```

### **SW1**

```
router eigrp 100
network 192.168.30.0
network 192.168.40.0
network 11.0.0.0
```

### **SW2**

```
router eigrp 100
network 192.168.50.0
network 192.168.60.0
network 22.0.0.0
```

### **SW3**

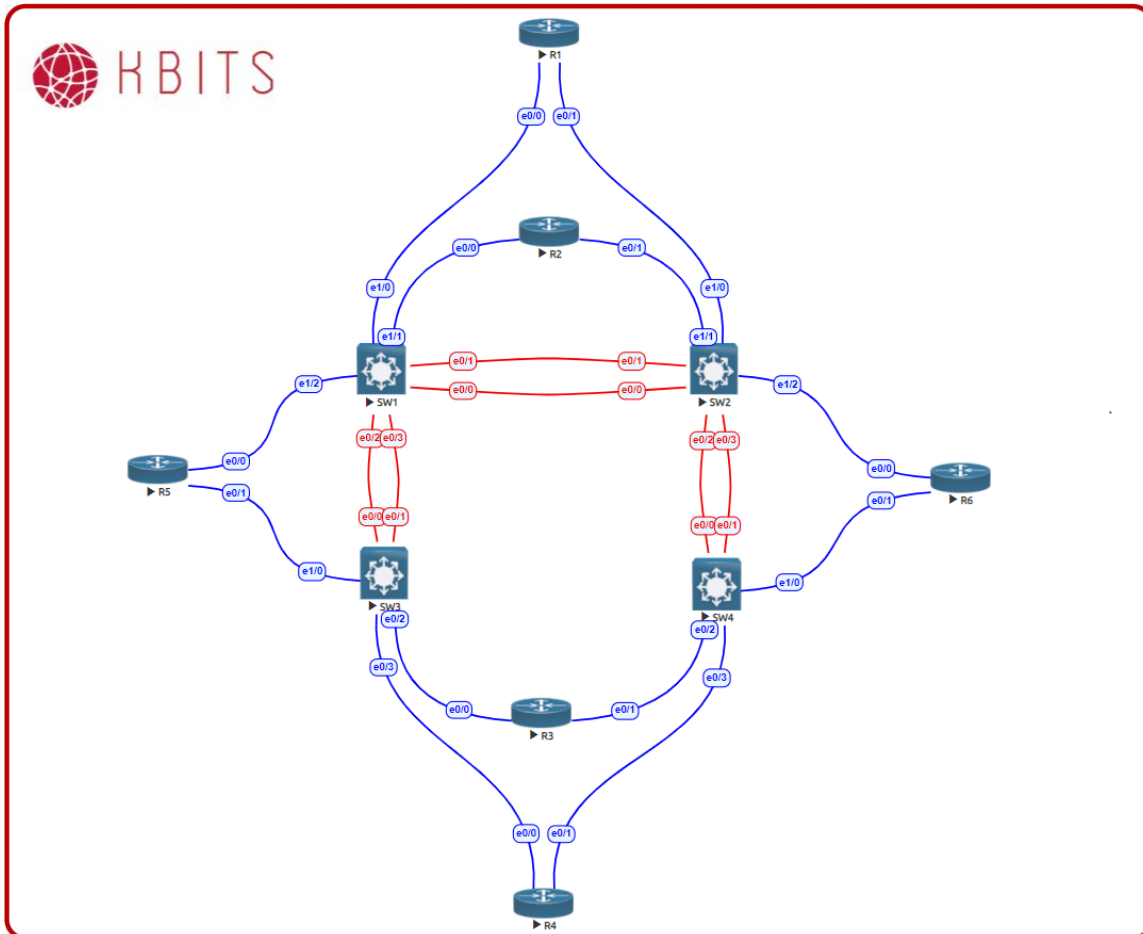
```
router eigrp 100
network 192.168.70.0
network 192.168.60.0
network 33.0.0.0
```

### **Task 3 – Verification**

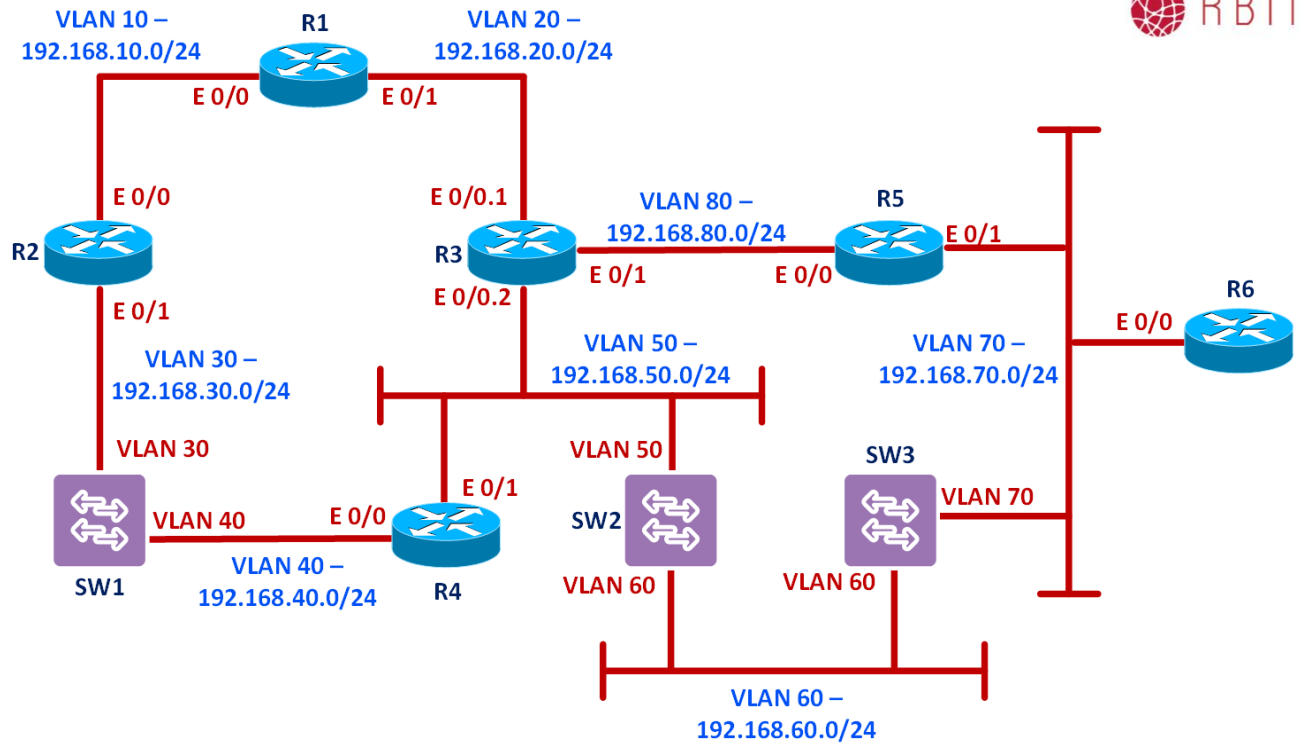
- Verify that all the Loopback routes are available in the Routing table and reachable.

# Lab 11 – Configuring the Port Fast Feature

## Physical Diagram



# Logical Diagram



## Task 1 – Configure PortFast

- Configuring all the ports that are connected towards the routers such that they bypass the STP Listening and Learning states.
- They should go into the STP Forwarding state immediately after being plugged in.

### SW1

Interface range E 1/0-2  
spanning-tree portfast

### SW2

Interface range E 1/0-2  
spanning-tree portfast

### SW3

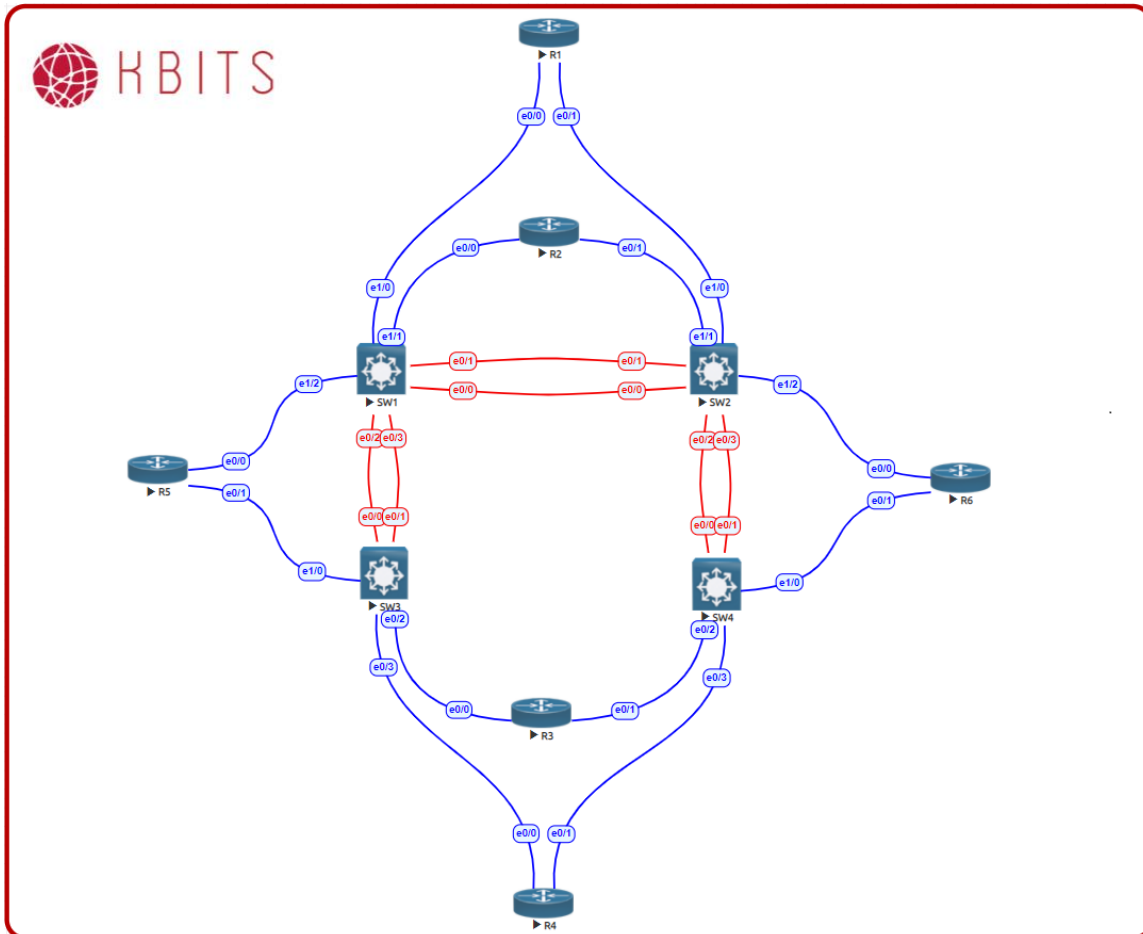
Interface range E 1/0, E 0/2-3  
spanning-tree portfast

### SW4

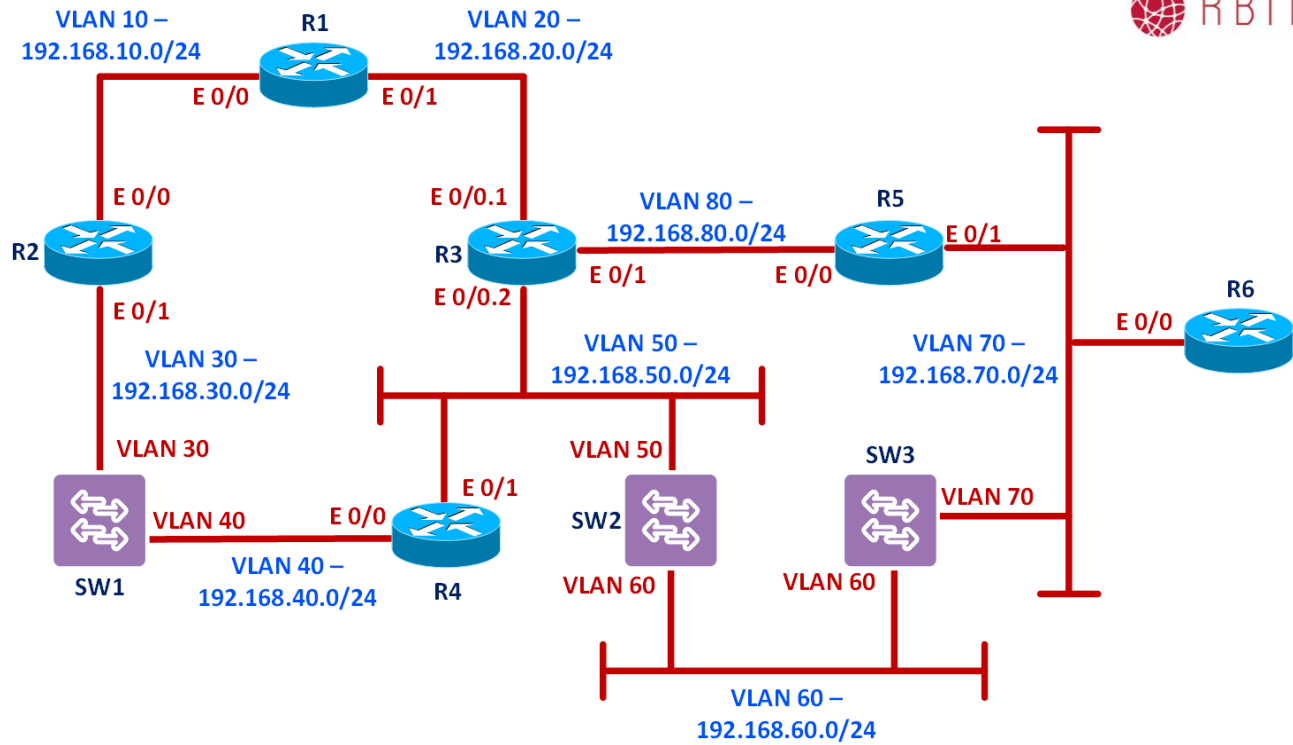
Interface range E 0/2-3  
spanning-tree portfast

# Lab 12 – Configuring the BPDU Guard Feature

## Physical Diagram



# Logical Diagram



### Task 1 – Configure BPDU Guard

- Configure the switches such that if it receives a BPDU on a port that is configured as portfast, the port should be disabled.

#### SW1

```
Interface range E 1/0-2
 spanning-tree bpduguard enable
```

#### SW2

```
Interface range E 1/0-2
 spanning-tree bpduguard enable
```

#### SW3

```
Interface range E 1/0, E 0/2-3
 spanning-tree bpduguard enable
```

#### SW4

```
Interface range E 0/2-3
 spanning-tree bpduguard enable
```

### Task 2 – Configure BPDU Guard Automatic Recovery

- The Switch should attempt to bring an error disabled port up automatically if it has been error disabled because of the BPDU Guard feature. It should try to recover the port after 180 seconds of it been error disabled.

#### SW1

```
errdisable recovery cause bpduguard
errdisable recovery interval 180
```

#### SW2

```
errdisable recovery cause bpduguard
errdisable recovery interval 180
```

#### SW3

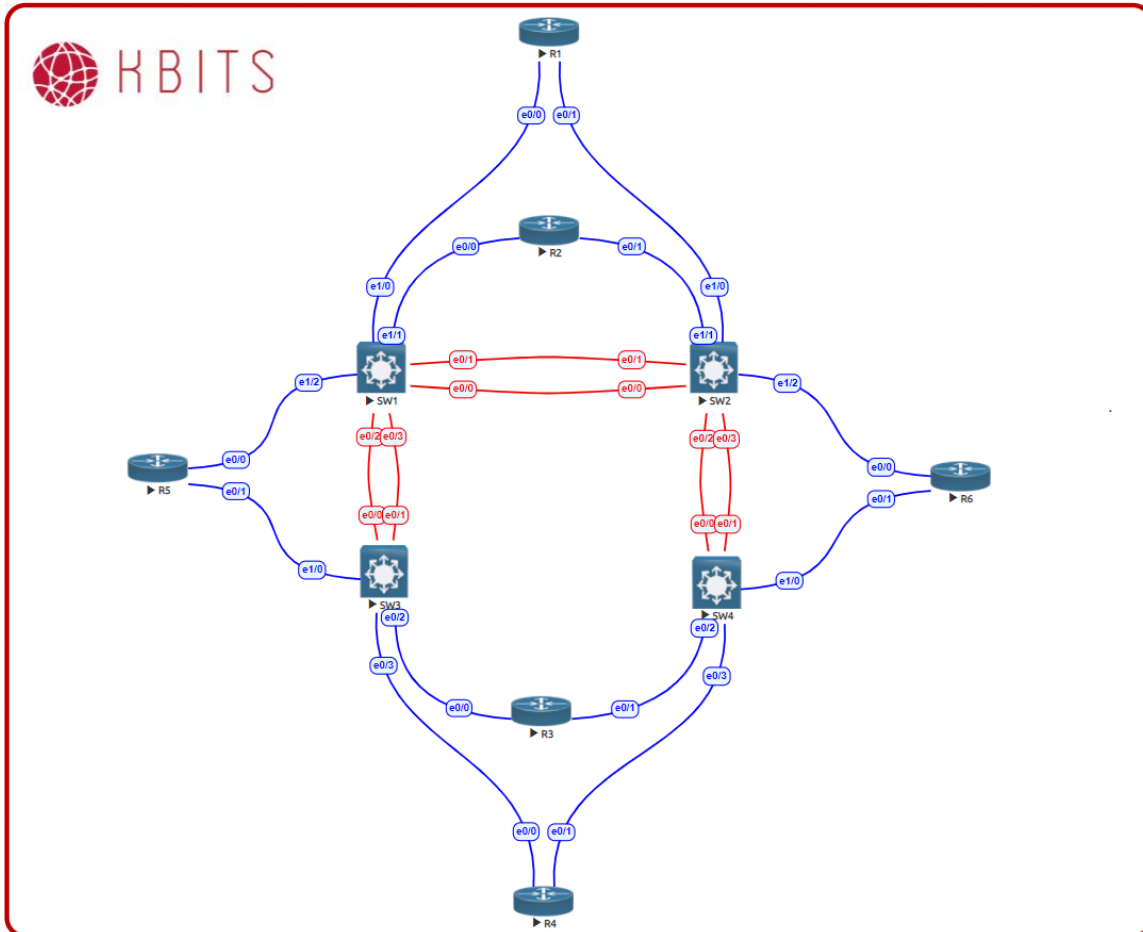
```
errdisable recovery cause bpduguard
errdisable recovery interval 180
```

#### SW4

```
errdisable recovery cause bpduguard
errdisable recovery interval 180
```

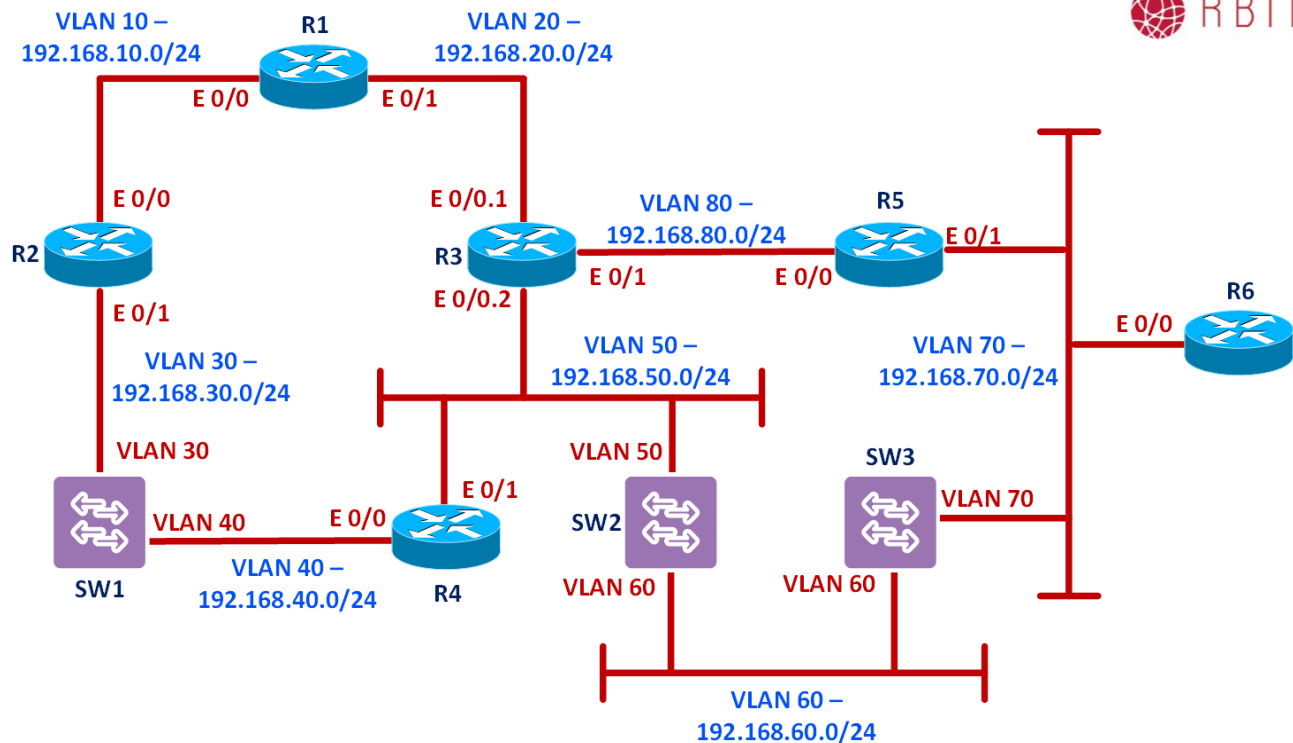
# Lab 13 – Configuring VLAN ACLs

## Physical Diagram





## Logical Diagram



### Task 1 – Configure a VLAN ACL

- You have been requested to implement the following Filtering policy on SW1:
  - Deny IGMP in VLAN 10
  - Deny TFTP in VLAN 20
  - Deny IGMP and TFTP in VLAN 30
  - There is a MAC address 0001.0012.2222 trying to attack VLAN 40. Block this MAC address from accessing any device in VLAN 40.

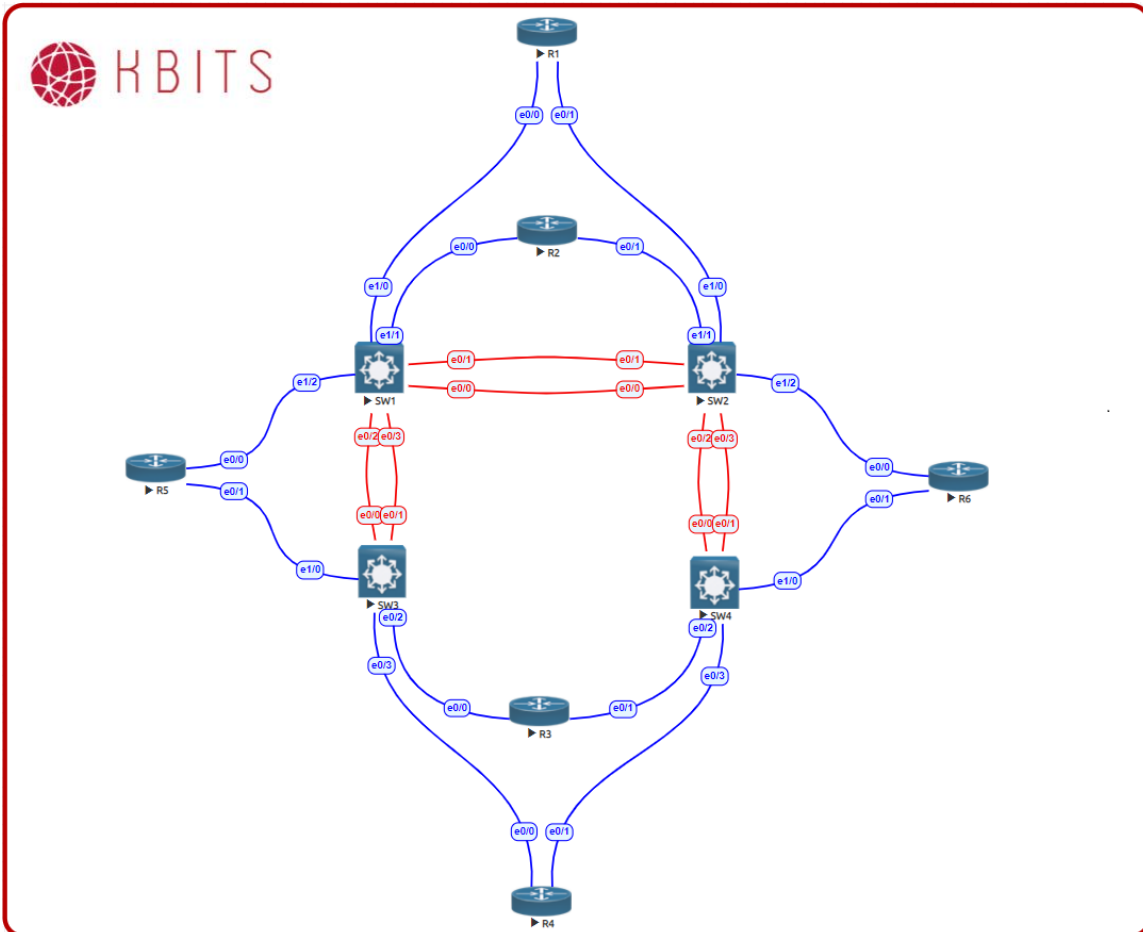
#### SW1

```
Access-list 101 permit igmp any any
!
Access-list 102 permit udp any any eq 69
!
Access-list 103 permit igmp any any
Access-list 103 permit udp any any eq 69
```

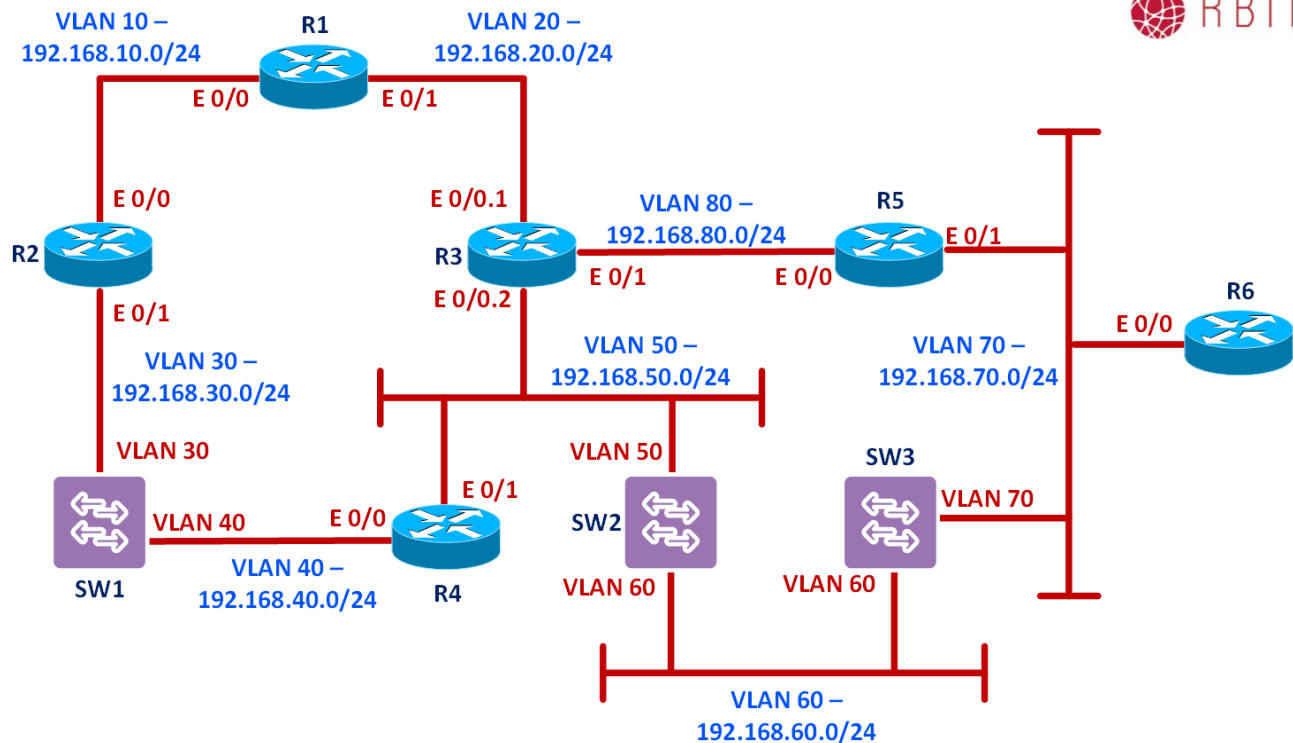
```
!  
Mac access-list extended MAC-ACL  
  Permit host 0001.0012.2222 any  
!  
Vlan access-map VLAN10 10  
  Match ip addr 101  
  Action drop  
Vlan access-map VLAN10 100  
!  
Vlan access-map VLAN20 10  
  Match ip addr 102  
  Action drop  
Vlan access-map VLAN20 100  
!  
Vlan access-map VLAN30 10  
  Match ip addr 103  
  Action drop  
Vlan access-map VLAN30 100  
!  
Vlan access-map VLAN40 10  
  Match mac address MAC-ACL  
  Action drop  
Vlan access-map VLAN40 100  
!  
Vlan filter VLAN10 vlan-list 10  
Vlan filter VLAN20 vlan-list 20  
Vlan filter VLAN30 vlan-list 30  
Vlan filter VLAN40 vlan-list 40
```

# Lab 14 – Configuring Root Guard

## Physical Diagram



## Logical Diagram



### Task 1 – Configure Root Guard

- Configure the ports that connect SW1 to SW2 in such a way that if for some reason the spanning-tree causes the links towards SW3 or SW4 to be selected as root ports, the ports should transition to a root-inconsistent (blocked) state.

#### SW1

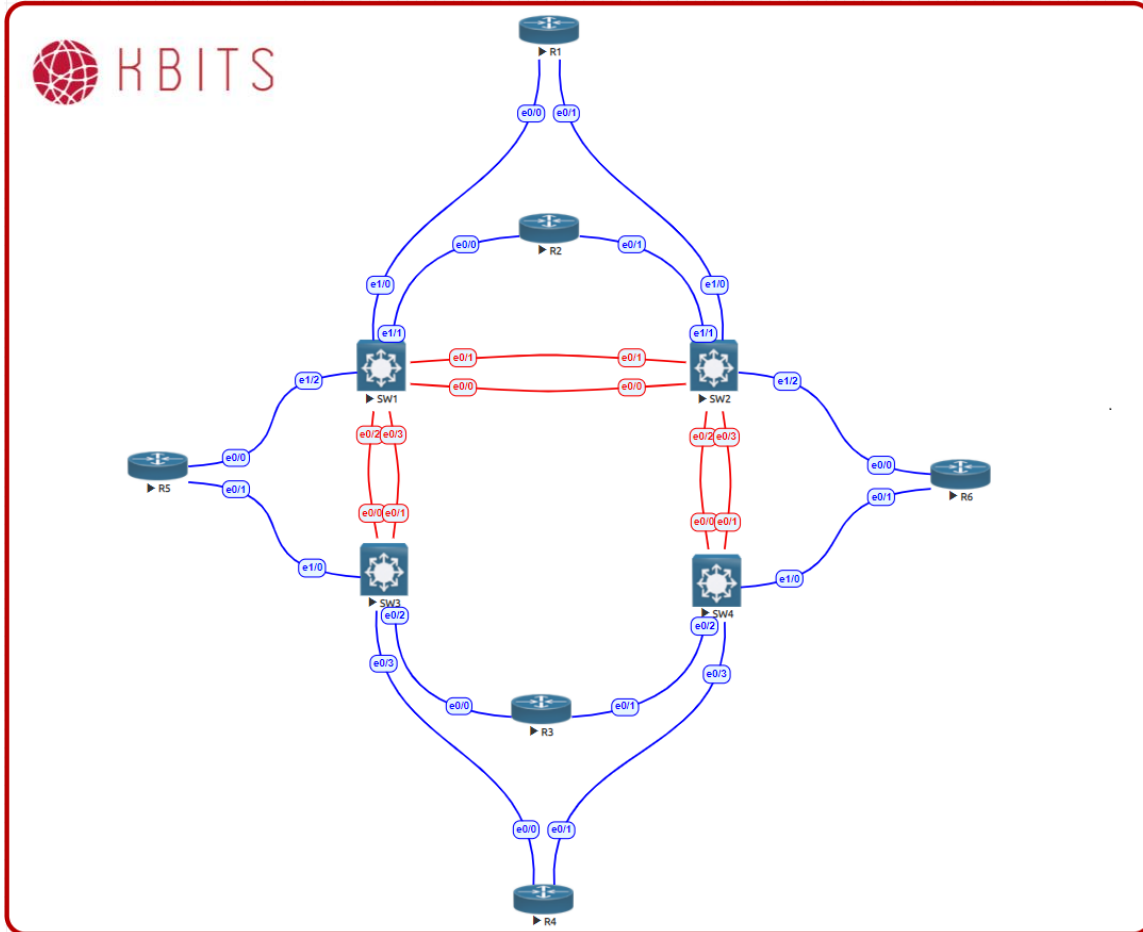
Interface range E 0/2-3  
Description Connection towards SW3  
Spanning-tree guard root

#### SW2

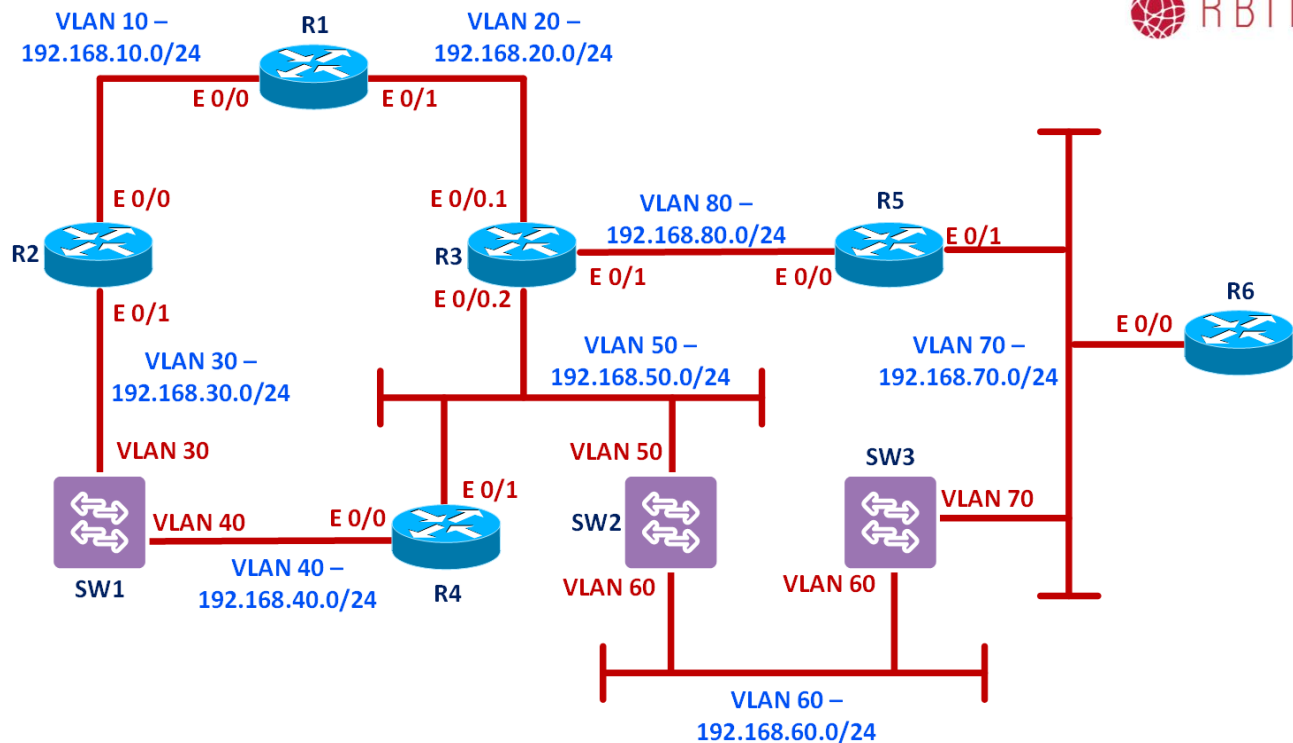
Interface port-channel24  
Description Connection towards SW4  
Spanning-tree guard root

# Lab 15 – Configuring Port-Security

## Physical Diagram



## Logical Diagram



### Task 1 – Configure Port Security – Static

- Configure the Router ports that connect SW1 in such a way that only the MAC to Port mappings is allowed on the specified ports:
  - R1 E0/0 MAC Address - Port E 1/0
  - R2 E0/0 MAC Address - Port E 1/1
  - R5 E0/0 MAC Address - Port E 1/2
- Find the MAC address of the Router Ports and statically enter them on SW1. (Use the **Show Interface** command on the individual router or use the **show mac address** command on SW1 to find the MAC address of the Router Ports.)

#### SW1

```

Interface range E 1/0
Description Connection towards R1
Switchport port-security
Switchport port-security mac xxxx.xxxx.xxxx (MAC address of R1)
!
Interface range E 1/1
    
```

```
Description Connection towards R2
Switchport port-security
Switchport port-security mac xxxx.xxxx.xxxx (MAC address of R2)
!
Interface range E 1/2
Description Connection towards R5
Switchport port-security
Switchport port-security mac xxxx.xxxx.xxxx (MAC address of R5)
```

### **Task 2 – Configure Port Security – Sticky**

- Configure the Router ports that connect SW2 in such a way that only the MAC to Port mappings is allowed on the specified ports:
  - R1 E0/1 MAC Address - Port E 1/0
  - R2 E0/1 MAC Address - Port E 1/1
  - R6 E0/0 MAC Address - Port E 1/2
- SW2 should learn the MAC address dynamically and store it in the running configuration file.

### **SW2**

```
Interface range E 1/0
Description Connection towards R1
Switchport port-security
Switchport port-security mac sticky
!
Interface range E 1/1
Description Connection towards R2
Switchport port-security
Switchport port-security mac sticky
!
Interface range E 1/2
Description Connection towards R5
Switchport port-security
Switchport port-security mac sticky
```

# Configuring EIGRP for IPv4 Networks

**Authored By:**

**Khawar Butt**

CCIE # 12353

Hepta CCIE#12353

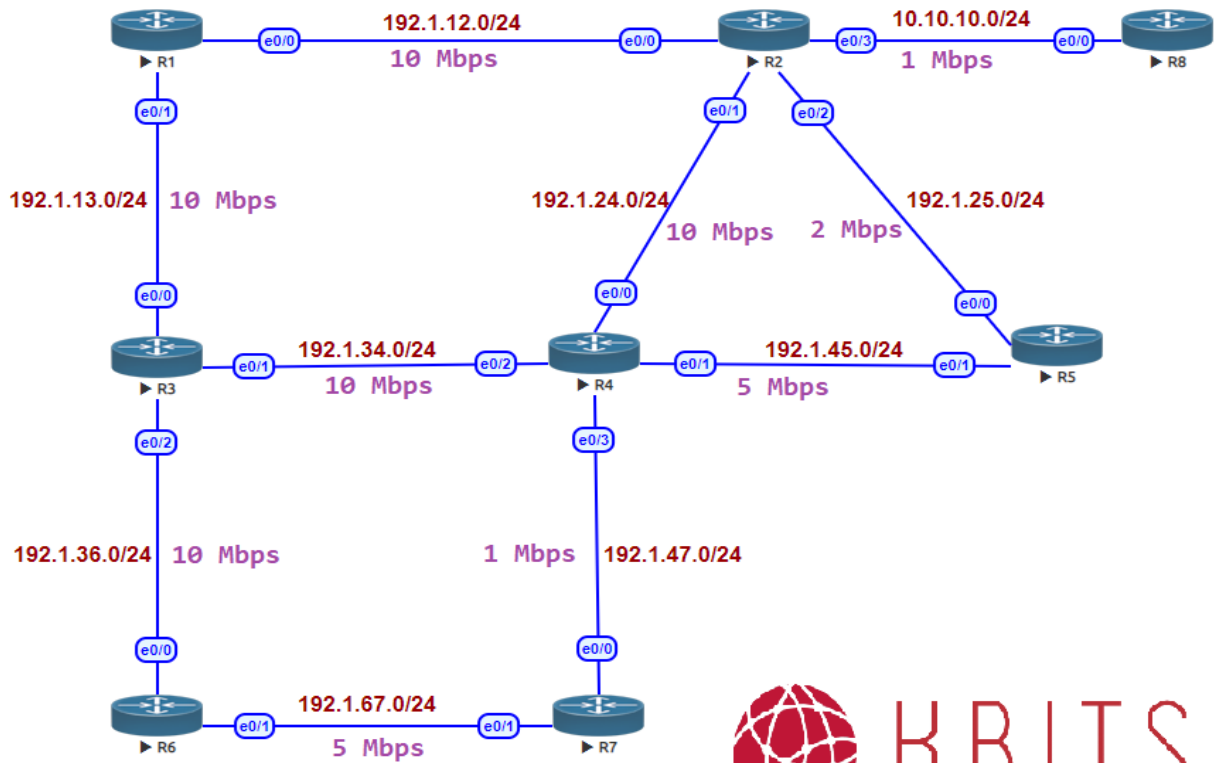
CCDE # 20110020

**Configuring EIGRP**





# Lab 1 – Initializing EIGRP – Network Statement



## Interface Configuration

### R1

Interface	IP Address	Subnet Mask
E 0/0	192.1.12.1	255.255.255.0
E 0/1	192.1.13.1	255.255.255.0
Loopback1	101.1.4.1	255.255.255.0
Loopback2	101.1.5.1	255.255.255.0
Loopback3	101.1.6.1	255.255.255.0
Loopback4	101.1.7.1	255.255.255.0

## R2

Interface	IP Address	Subnet Mask
E 0/0	192.1.12.2	255.255.255.0
E 0/1	192.1.24.2	255.255.255.0
E 0/2	192.1.25.2	255.255.255.0
E 0/3	10.10.10.2	255.255.255.0
Loopback1	202.1.4.1	255.255.255.0
Loopback2	202.1.5.1	255.255.255.0
Loopback3	202.1.6.1	255.255.255.0
Loopback4	202.1.7.1	255.255.255.0
Loopback5	10.1.4.1	255.255.255.0
Loopback6	10.1.5.1	255.255.255.0
Loopback7	10.1.6.1	255.255.255.0
Loopback8	10.1.7.1	255.255.255.0

## R3

Interface	IP Address	Subnet Mask
E 0/0	192.1.13.3	255.255.255.0
E 0/1	192.1.34.3	255.255.255.0
E 0/2	192.1.36.3	255.255.255.0
Loopback1	203.1.4.1	255.255.255.0
Loopback2	203.1.5.1	255.255.255.0
Loopback3	203.1.6.1	255.255.255.0
Loopback4	203.1.7.1	255.255.255.0

## R4

Interface	IP Address	Subnet Mask
E 0/0	192.1.24.4	255.255.255.0
E 0/1	192.1.45.4	255.255.255.0
E 0/2	192.1.34.4	255.255.255.0
E 0/3	192.1.47.4	255.255.255.0
Loopback1	104.1.8.1	255.255.255.0
Loopback2	104.1.9.1	255.255.255.0
Loopback3	104.1.10.1	255.255.255.0
Loopback4	104.1.11.1	255.255.255.0

## R5

Interface	IP Address	Subnet Mask
E 0/0	192.1.25.5	255.255.255.0

E 0/1	192.1.45.5	255.255.255.0
Loopback1	205.1.4.1	255.255.255.0
Loopback2	205.1.5.1	255.255.255.0
Loopback3	205.1.6.1	255.255.255.0
Loopback4	205.1.7.1	255.255.255.0

## R6

Interface	IP Address	Subnet Mask
E 0/0	192.1.36.6	255.255.255.0
E 0/1	192.1.67.6	255.255.255.0
Loopback1	101.1.60.1	255.255.255.0
Loopback2	101.1.61.1	255.255.255.0
Loopback3	101.1.62.1	255.255.255.0
Loopback4	101.1.63.1	255.255.255.0

## R7

Interface	IP Address	Subnet Mask
E 0/0	192.1.47.7	255.255.255.0
E 0/1	192.1.67.7	255.255.255.0
Loopback1	101.1.72.1	255.255.255.0
Loopback2	101.1.73.1	255.255.255.0
Loopback3	101.1.74.1	255.255.255.0
Loopback4	101.1.75.1	255.255.255.0

## R8

Interface	IP Address	Subnet Mask
E 0/0	10.10.10.8	255.255.255.0
Loopback1	10.1.8.1	255.255.255.0
Loopback2	10.1.9.1	255.255.255.0
Loopback3	10.1.10.1	255.255.255.0
Loopback4	10.1.11.1	255.255.255.0

### **Task 1 – Configure R1 in EIGRP AS 111.**

- Configure the Interface based on the Interface Configuration Table.
- Run EIGRP in EIGRP 111.
- Configure the Network statement based on Major networks only.

#### **R1**

```
Hostname R1
!
Interface E 0/0
ip address 192.1.12.1 255.255.255.0
no shut
!
Interface E 0/1
ip address 192.1.13.1 255.255.255.0
no shut
!
Interface Loopback1
ip address 101.1.4.1 255.255.255.0
!
Interface Loopback2
ip address 101.1.5.1 255.255.255.0
!
Interface Loopback3
ip address 101.1.6.1 255.255.255.0
!
Interface Loopback4
ip address 101.1.7.1 255.255.255.0
!
router eigrp 111
network 192.1.12.0
network 192.1.13.0
network 101.0.0.0
```

## Task 2 – Configure R2 in EIGRP AS 111.

- Configure the Interface based on the Interface Configuration Table.
- Run EIGRP in EIGRP 111.
- Configure the Network statement using a wild card mask for the Network 10.0.0.0. Use the Major networks for the rest of the networks.

### R2

```
Hostname R2
!
Interface E 0/0
ip address 192.1.12.2 255.255.255.0
no shut
!
Interface E 0/1
ip address 192.1.24.2 255.255.255.0
no shut
!
Interface E 0/2
ip address 192.1.25.2 255.255.255.0
no shut
!
Interface E 0/3
ip address 10.10.10.2 255.255.255.0
no shut
!
Interface Loopback1
ip address 202.1.4.1 255.255.255.0
!
Interface Loopback2
ip address 202.1.5.1 255.255.255.0
!
Interface Loopback3
ip address 202.1.6.1 255.255.255.0
!
Interface Loopback4
ip address 202.1.7.1 255.255.255.0
!
Interface Loopback5
ip address 10.1.4.1 255.255.255.0
!
Interface Loopback6
ip address 10.1.5.1 255.255.255.0
```

```
!  
Interface Loopback7  
ip address 10.1.6.1 255.255.255.0  
!  
Interface Loopback8  
ip address 10.1.7.1 255.255.255.0  
!  
router eigrp 111  
network 192.1.12.0  
network 192.1.24.0  
network 192.1.25.0  
network 10.10.10.0 0.0.0.255  
network 10.1.0.0 0.0.255.255  
network 202.1.4.0  
network 202.1.5.0  
network 202.1.6.0  
network 202.1.7.0
```

### **Task 3 – Configure R3 in EIGRP AS 111.**

- Configure the Interface based on the Interface Configuration Table.
- Run EIGRP in EIGRP 111.
- Configure the Network statement using a wild card mask to minimize the Network statements starting with 203.X.X.0. Use the Major networks for the rest of the networks.

### **R3**

```
Hostname R3  
!  
Interface E 0/0  
ip address 192.1.13.3 255.255.255.0  
no shut  
!  
Interface E 0/1  
ip address 192.1.34.3 255.255.255.0  
no shut  
!  
Interface E 0/2  
ip address 192.1.36.3 255.255.255.0  
no shut  
!  
Interface Loopback1  
ip address 203.1.4.1 255.255.255.0  
!
```

```
Interface Loopback2
ip address 203.1.5.1 255.255.255.0
!
Interface Loopback3
ip address 203.1.6.1 255.255.255.0
!
Interface Loopback4
ip address 203.1.7.1 255.255.255.0
!
router eigrp 111
network 192.1.13.0
network 192.1.34.0
network 192.1.36.0
network 203.1.0.0 0.0.255.255
```

#### **Task 4 – Configure R4 in EIGRP AS 111.**

- Configure the Interface based on the Interface Configuration Table.
- Run EIGRP in EIGRP 111.
- Configure the Network statement using a wild card mask to minimize the Network statements starting with 104.X.X.0. Use the Major networks for the rest of the networks.

#### **R4**

```
Hostname R4
!
Interface E 0/0
ip address 192.1.24.4 255.255.255.0
no shut
!
Interface E 0/1
ip address 192.1.45.4 255.255.255.0
no shut
!
Interface E 0/2
ip address 192.1.34.4 255.255.255.0
no shut
!
Interface E 0/3
ip address 192.1.47.4 255.255.255.0
no shut
!
Interface Loopback1
```

```
ip address 104.1.8.1 255.255.255.0
!  
Interface Loopback2  
ip address 104.1.9.1 255.255.255.0  
!  
Interface Loopback3  
ip address 104.1.10.1 255.255.255.0  
!  
Interface Loopback4  
ip address 104.1.11.1 255.255.255.0  
!  
router eigrp 111  
network 192.1.24.0  
network 192.1.34.0  
network 192.1.45.0  
network 192.1.47.0  
network 104.1.0.0 0.0.255.255
```

### **Task 5 – Configure R5 in EIGRP AS 111.**

- Configure the Interface based on the Interface Configuration Table.
- Run EIGRP in EIGRP 111.
- Configure the Network statement using a wild card mask to minimize the Network statements starting with 205.X.X.0. Use the Major networks for the rest of the networks.

### **R5**

```
Hostname R5  
!  
Interface E 0/0  
ip address 192.1.25.5 255.255.255.0  
no shut  
!  
Interface E 0/1  
ip address 192.1.45.5 255.255.255.0  
no shut  
!  
Interface Loopback1  
ip address 205.1.4.1 255.255.255.0  
!  
Interface Loopback2  
ip address 205.1.5.1 255.255.255.0  
!  
Interface Loopback3
```



```
ip address 205.1.6.1 255.255.255.0
!  
Interface Loopback4  
ip address 205.1.7.1 255.255.255.0  
!  
router eigrp 111  
network 192.1.25.0  
network 192.1.45.0  
network 205.1.0.0 0.0.255.255
```

### **Task 6 – Configure R6 in EIGRP AS 111.**

- Configure the Interface based on the Interface Configuration Table.
- Run EIGRP in EIGRP 111.
- Configure the Network statement using a wild card mask to minimize the Network statements starting with 101.X.X.0. Use the Major networks for the rest of the networks.

#### **R6**

```
Hostname R6  
!  
Interface E 0/0  
ip address 192.1.36.6 255.255.255.0  
no shut  
!  
Interface E 0/1  
ip address 192.1.67.6 255.255.255.0  
no shut  
!  
Interface Loopback1  
ip address 101.1.60.1 255.255.255.0  
!  
Interface Loopback2  
ip address 101.1.61.1 255.255.255.0  
!  
Interface Loopback3  
ip address 101.1.62.1 255.255.255.0  
!  
Interface Loopback4  
ip address 101.1.63.1 255.255.255.0  
!  
router eigrp 111  
network 192.1.36.0  
network 192.1.67.0
```

```
network 101.1.0.0 0.0.255.255
```

### **Task 7 – Configure R7 in EIGRP AS 111.**

- Configure the Interface based on the Interface Configuration Table.
- Run EIGRP in EIGRP 111.
- Configure the Network statement using a wild card mask to minimize the Network statements starting with 101.X.X.0. Use the Major networks for the rest of the networks.

#### **R7**

```
Hostname R7
!  
Interface E 0/0  
ip address 192.1.47.7 255.255.255.0  
no shut  
!  
Interface E 0/1  
ip address 192.1.67.7 255.255.255.0  
no shut  
!  
Interface Loopback1  
ip address 101.1.72.1 255.255.255.0  
!  
Interface Loopback2  
ip address 101.1.73.1 255.255.255.0  
!  
Interface Loopback3  
ip address 101.1.74.1 255.255.255.0  
!  
Interface Loopback4  
ip address 101.1.75.1 255.255.255.0  
!  
router eigrp 111  
network 192.1.47.0  
network 192.1.67.0  
network 101.1.0.0 0.0.255.255
```

### Task 8 – Configure R8 in EIGRP AS 111.

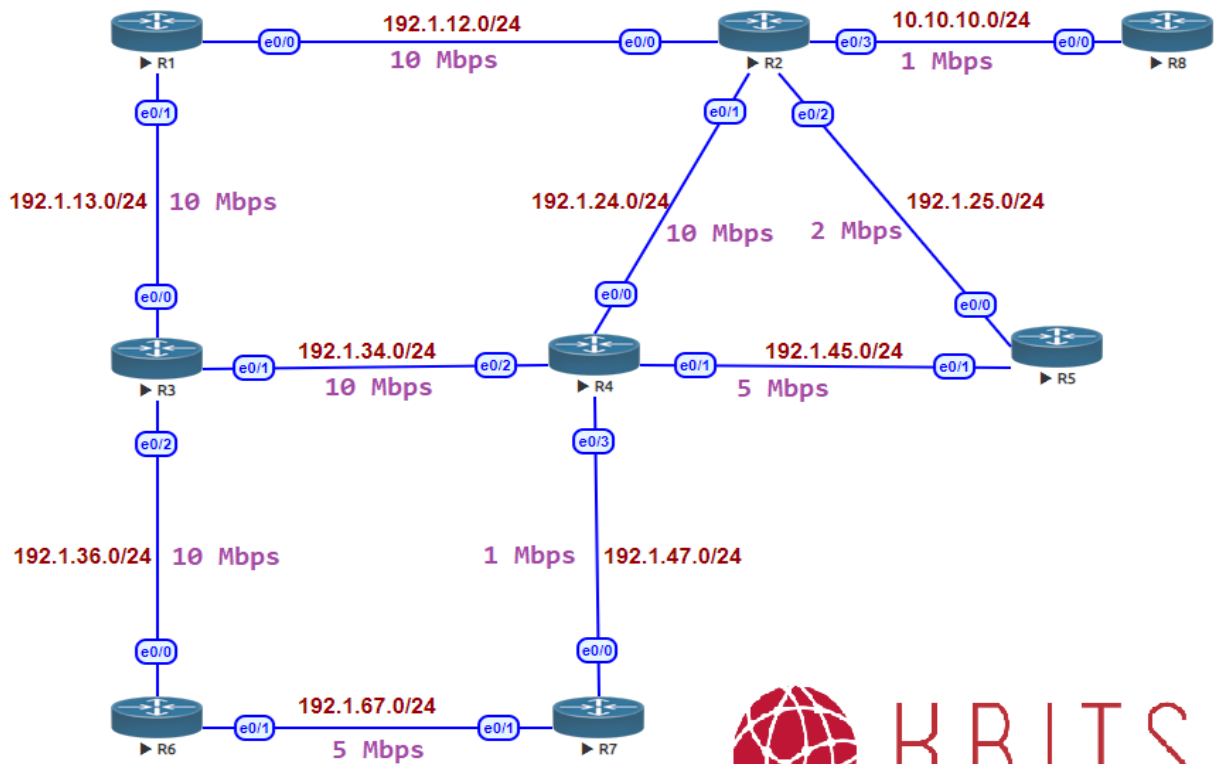
- Configure the Interface based on the Interface Configuration Table.
- Run EIGRP in EIGRP 111.
- Configure the Network statement such that all directly connected interfaces are enabled in EIGRP. This should also take care of any new interfaces configured in the future.

#### R8

```
Hostname R8
!
Interface E 0/0
ip address 10.10.10.8 255.255.255.0
no shut
!
Interface Loopback1
ip address 10.1.8.1 255.255.255.0
!
Interface Loopback2
ip address 10.1.9.1 255.255.255.0
!
Interface Loopback3
ip address 10.1.10.1 255.255.255.0
!
Interface Loopback4
ip address 10.1.11.1 255.255.255.0
!
router eigrp 111
network 0.0.0.0
```

# Lab 2 – EIGRP – Passive Interfaces

**Note: It builds on the topology created in the previous lab.**



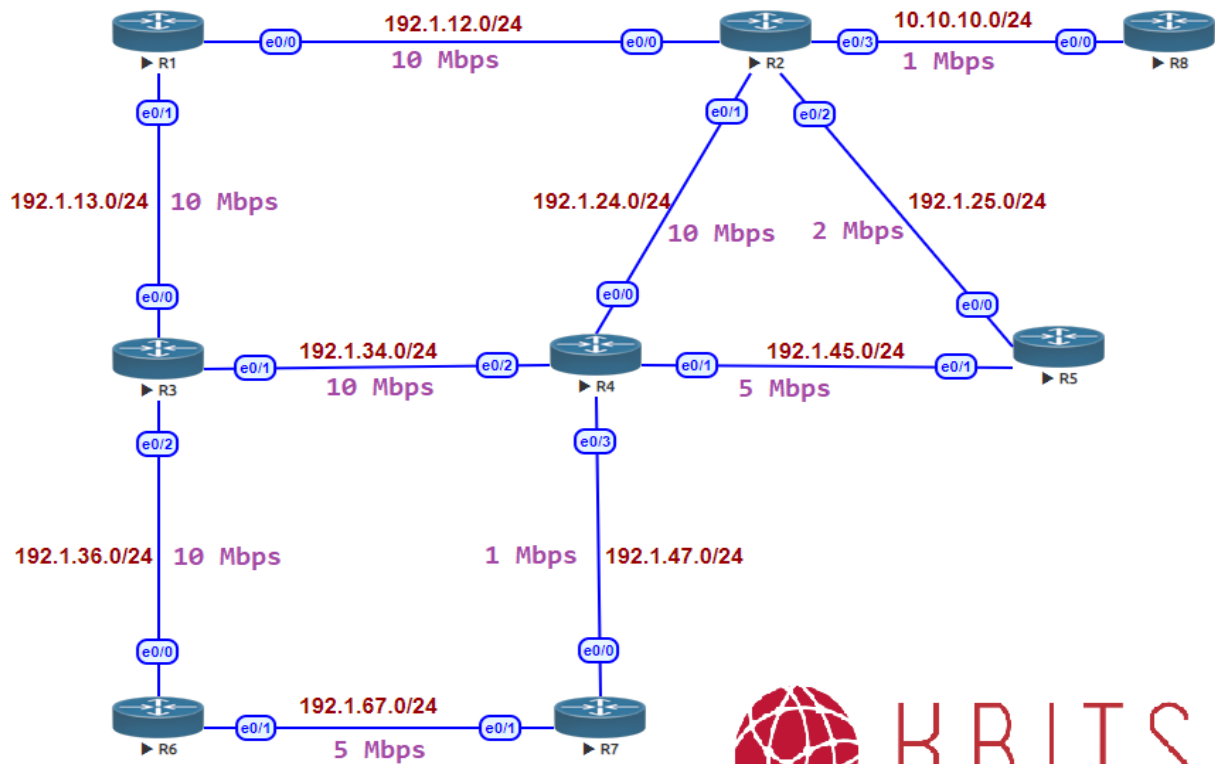
### Task 1 - Configure Passive-Interface on Routers in EIGRP AS 111.

- Configure all routers in EIGRP 111 such that they do not send updates on links that do not have other routers, basically Loopbacks.
- Use the minimum number of passive interface commands to accomplish the task.

<b>R1</b>  router eigrp 111 passive-interface default no passive-interface E 0/0 no passive-interface E 0/1	<b>R2</b>  router eigrp 111 passive-interface default no passive-interface E 0/0 no passive-interface E 0/1 no passive-interface E 0/2 no passive-interface E 0/3
<b>R3</b>  router eigrp 111 passive-interface default no passive-interface E 0/0 no passive-interface E 0/1 no passive-interface E 0/2	<b>R4</b>  router eigrp 111 passive-interface default no passive-interface E 0/0 no passive-interface E 0/1 no passive-interface E 0/2 no passive-interface E 0/3
<b>R5</b>  router eigrp 111 passive-interface default no passive-interface E 0/0 no passive-interface E 0/1	<b>R6</b>  router eigrp 111 passive-interface default no passive-interface E 0/0 no passive-interface E 0/1
<b>R7</b>  router eigrp 111 passive-interface default no passive-interface E 0/0 no passive-interface E 0/1	<b>R8</b>  router eigrp 111 passive-interface default no passive-interface E 0/0

# Lab 3 – EIGRP – Unicast Neighbors

**Note: It builds on the topology created in the previous lab.**



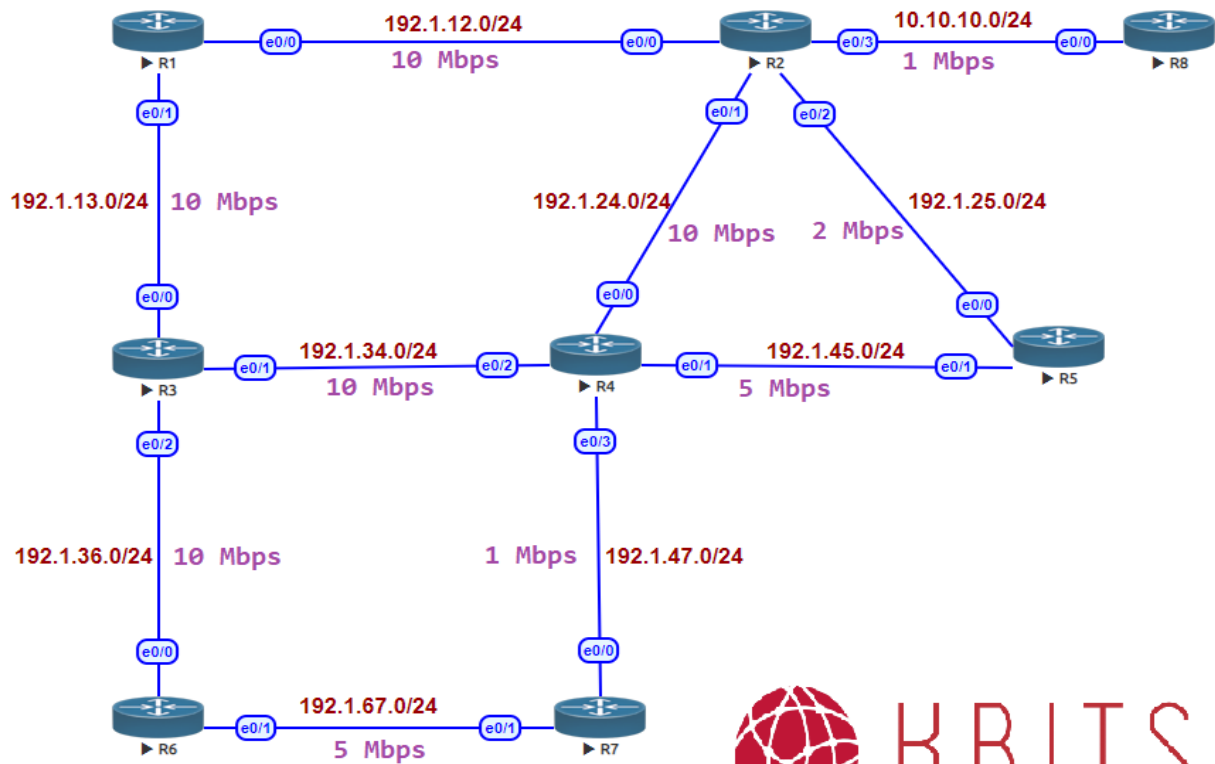
**Task 1 – Configure Unicast EIGRP on specific interfaces in AS 111.**

- Configure the neighbor relationship between R1 and R2 to be Unicast-based.
- Configure the neighbor relationship between R2 and R8 to be Unicast-based.

<b>R1</b> router eigrp 111 neighbor 192.1.12.2 E0/0	<b>R2</b> router eigrp 111 neighbor 192.1.12.1 E0/0 neighbor 10.10.10.8 E0/3
<b>R8</b> router eigrp 111 neighbor 10.10.10.2 E0/0	

# Lab 4 – EIGRP – Metric Calculations

**Note: It builds on the topology created in the previous lab.**



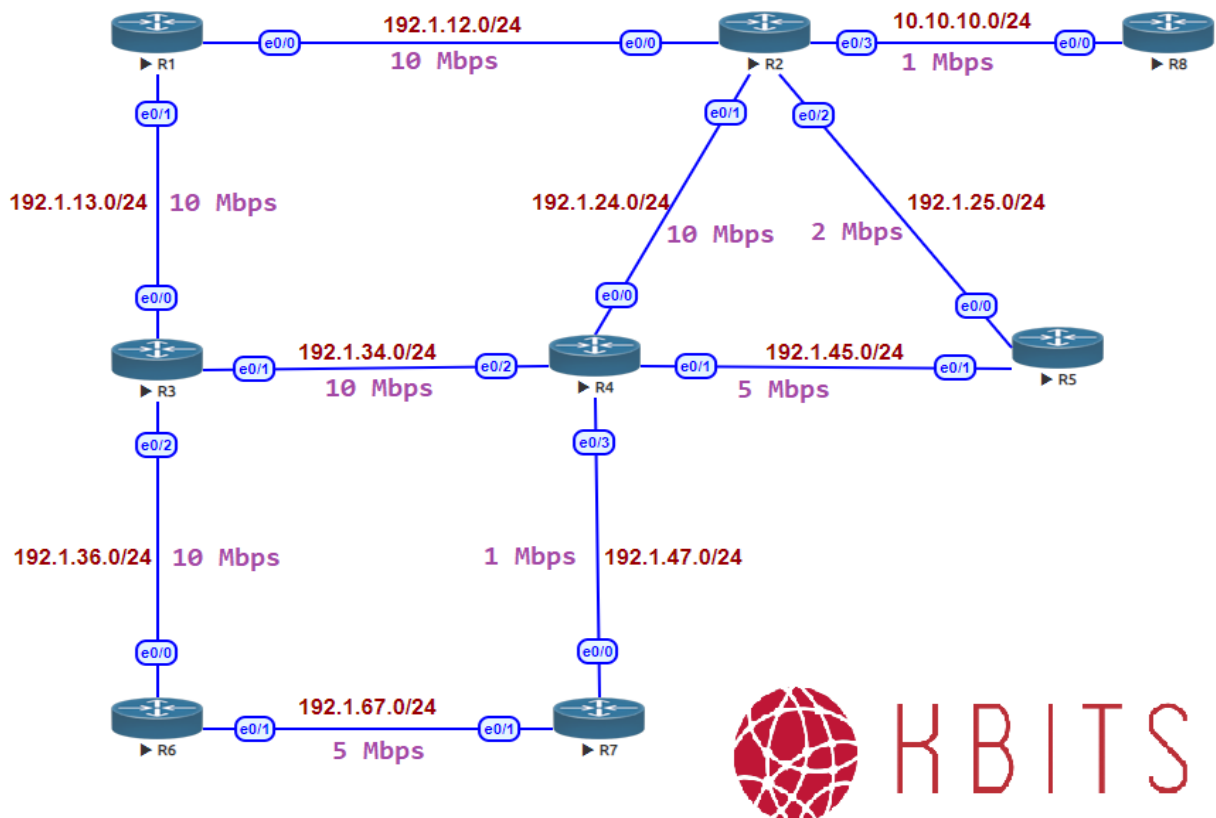


**Task 1 – Configure interface bandwidth based on the topology diagram.**

- Configure the Interfaces on the routers based on the bandwidth shown in the diagram. Don't change the delay.
- Calculate the metrics from R2 towards the 205.1.4.0/24 network. Make sure that it matches the values mentioned in the video.

<p><b>R1</b></p> <p>Interface E 0/0 Bandwidth 10000 Interface E 0/1 Bandwidth 10000</p>	<p><b>R2</b></p> <p>Interface E 0/0 Bandwidth 10000 Interface E 0/1 Bandwidth 10000 Interface E 0/2 Bandwidth 2000 Interface E 0/3 Bandwidth 1000</p>
<p><b>R3</b></p> <p>Interface E 0/0 Bandwidth 10000 Interface E 0/1 Bandwidth 10000 Interface E 0/2 Bandwidth 10000</p>	<p><b>R4</b></p> <p>Interface E 0/0 Bandwidth 10000 Interface E 0/1 Bandwidth 5000 Interface E 0/2 Bandwidth 10000 Interface E 0/3 Bandwidth 1000</p>
<p><b>R5</b></p> <p>Interface E 0/0 Bandwidth 2000 Interface E 0/1 Bandwidth 5000</p>	<p><b>R6</b></p> <p>Interface E 0/0 Bandwidth 10000 Interface E 0/1 Bandwidth 5000</p>
<p><b>R7</b></p> <p>Interface E 0/0 Bandwidth 1000 Interface E 0/1 Bandwidth 5000</p>	<p><b>R5</b></p> <p>Interface E 0/0 Bandwidth 1000</p>

# Lab 5 – Load Balancing – Equal & Unequal Load Balancing



## Task 1 – Verifying Equal Cost Load Balancing between R1, R2, R3 & R4.

- The Interface bandwidth between R1, R2, R3 & R4 is the same. This results in equal costs to get from diagonally across routers ( R1-R4 and vice versa & R2-R3 and vice versa).
- Verify the dual path from R1 towards the loopbacks of R4 and vice versa.
- Verify the dual path from R2 towards the loopbacks of R3 and vice versa.

<b>R1</b> Show IP Route eigrp	<b>R2</b> Show IP Route eigrp
<b>R3</b> Show IP Route eigrp	<b>R4</b> Show IP Route eigrp

### Task 2 – Configuring Unequal Load Balancing on R2

- Configure R2 to use both path towards the Loopback Interfaces of R5.
- Calculate the Variance and implement it on R2.
- This can be done by finding the Composite Metric for the Successor and Feasible successors in the EIGRP topology table.
- Divide the Feasible Successor Metric (1433600) by Successor Metric (691200).
- You will get a result of 2.07. Round up the number to 3.
- That is the variance.
- Verify the Traffic share count by using the Show Ip route command with the route option (show ip route 205.1.4.0)

#### R2

```
Router eigrp 111
Variance 3
```

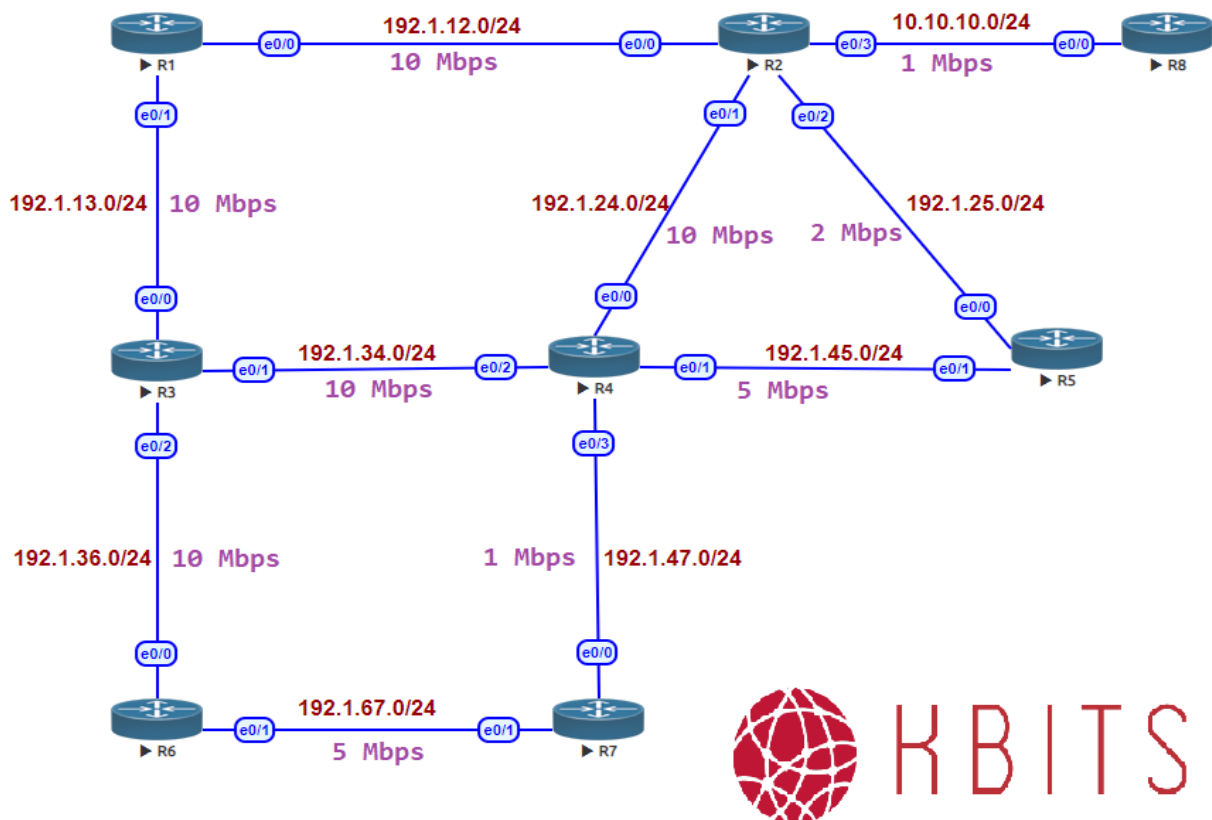
### Task 3 – Configuring Unequal Load Balancing on R5

- Configure R5 to use both path towards the Loopback Interfaces of R2.
- Calculate the Variance and implement it on R5.
- This can be done by finding the Composite Metric for the Successor and Feasible successors in the EIGRP topology table.
- Divide the Feasible Successor Metric (1433600) by Successor Metric (691200).
- You will get a result of 2.07. Round up the number to 3.
- That is the variance.
- Verify the Traffic share count by using the Show Ip route command with the route option (show ip route 205.1.4.0)

#### R5

```
Router eigrp 111
Variance 3
```

# Lab 6 – Route Summarization – Auto Summary



## Task 1 – Configuring Auto-Summary on R1 & R6

R1	R6
Router eigrp 111 Auto-summary	Router eigrp 111 Auto-summary

## Task 2 – Verifying the results of Auto-Summary

- Ping a R1 Loopback IP (101.1.4.1) from R6 or R7.
- Are you successful?
- What is the reason for that?

### Task 3 – Configuring Auto-Summary on R2 & R8

<b>R2</b> Router eigrp 111 Auto-summary	<b>R8</b> Router eigrp 111 Auto-summary
---	---

### Task 4 – Verifying Auto-Summarization between R2 & R8

- Are the Network 10.0.0.0/8 subnets summarized between R2 & R8?

### Task 5 – Configuring the following Interfaces on R2 & R8 & Enable them in EIGRP 111.

<b>R2</b> Interface loopback401 Ip address 102.1.4.1 255.255.255.0 Interface loopback402 Ip address 102.1.5.1 255.255.255.0 Interface loopback403 Ip address 102.1.6.1 255.255.255.0 Interface loopback404 Ip address 102.1.7.1 255.255.255.0 ! Router eigrp 111 Network 102.0.0.0	<b>R8</b> Interface loopback401 Ip address 108.1.4.1 255.255.255.0 Interface loopback402 Ip address 108.1.5.1 255.255.255.0 Interface loopback403 Ip address 108.1.6.1 255.255.255.0 Interface loopback404 Ip address 108.1.7.1 255.255.255.0 ! Router eigrp 111 Network 108.0.0.0
---	---

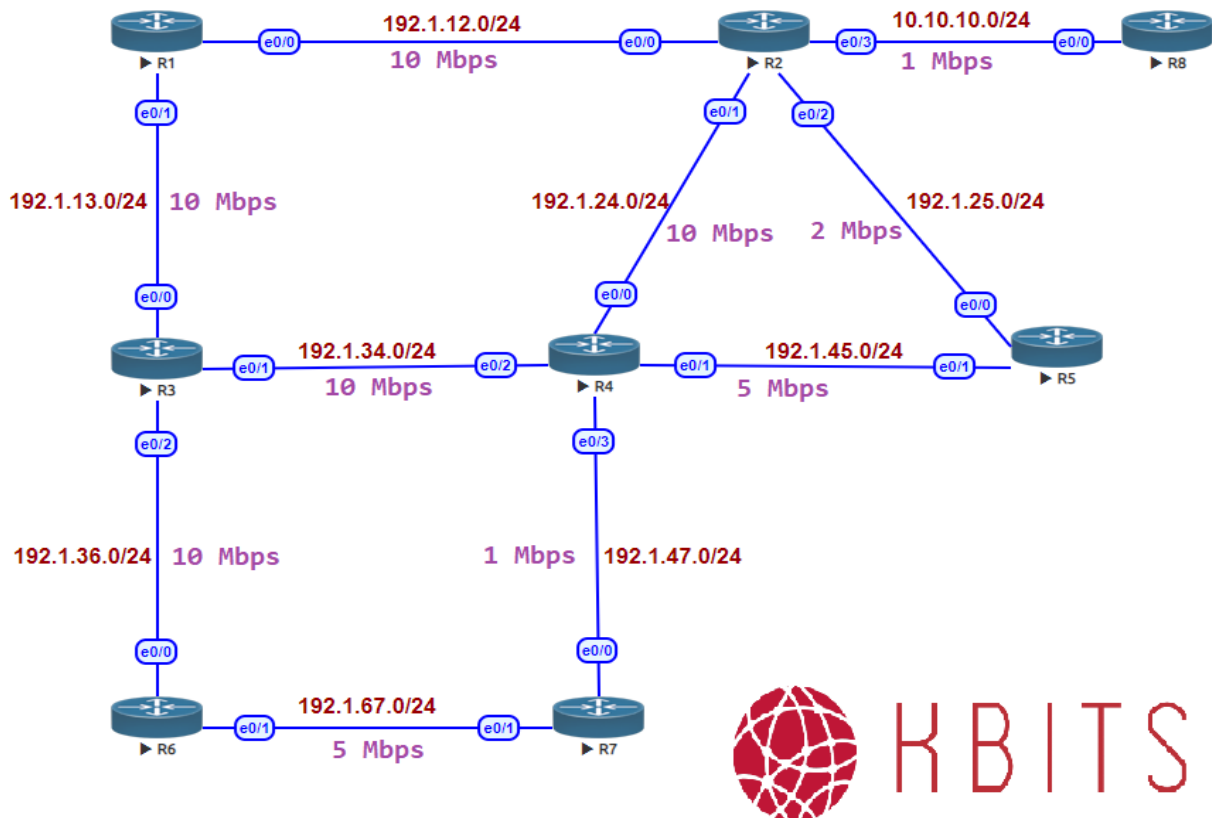
### Task 6 – Verifying Auto-Summarization between R2 & R8

- Are the new Loopback Networks summarized between R2 & R8?
- Why?

### Task 7 – Disable Auto-Summarization on R1, R2, R6 & R8

<b>R1</b> Router eigrp 111 No Auto-summary	<b>R2</b> Router eigrp 111 No Auto-summary
<b>R6</b> Router eigrp 111 No Auto-summary	<b>R8</b> Router eigrp 111 No Auto-summary

# Lab 7 – Route Summarization – Manual Summarization



**Task 1 – Configuring Manual Summarization on R1 for all the 101.0.0.0/8 subnets using the Longest Summary Mask. Summarization should be configured towards all the neighbors.**

## R1

```
Interface E 0/0
Ip summary-address eigrp 111 101.1.4.0 255.255.252.0
!
Interface E 0/1
Ip summary-address eigrp 111 101.1.4.0 255.255.252.0
```

**Task 2 – Configuring Manual Summarization on R2 for all the 10.0.0.0/8 subnets, 102.0.0.0/8 subnets & the 202.X.X.0/24 major networks using the Longest Summary Mask. Summarization should be configured towards all the neighbors.**

**R2**

```
Interface E 0/0
Ip summary-address eigrp 111 10.1.4.0 255.255.252.0
Ip summary-address eigrp 111 102.1.4.0 255.255.252.0
Ip summary-address eigrp 111 202.1.4.0 255.255.252.0
!
Interface E 0/1
Ip summary-address eigrp 111 10.1.4.0 255.255.252.0
Ip summary-address eigrp 111 102.1.4.0 255.255.252.0
Ip summary-address eigrp 111 202.1.4.0 255.255.252.0
!
Interface E 0/2
Ip summary-address eigrp 111 10.1.4.0 255.255.252.0
Ip summary-address eigrp 111 102.1.4.0 255.255.252.0
Ip summary-address eigrp 111 202.1.4.0 255.255.252.0
!
Interface E 0/3
Ip summary-address eigrp 111 10.1.4.0 255.255.252.0
Ip summary-address eigrp 111 102.1.4.0 255.255.252.0
Ip summary-address eigrp 111 202.1.4.0 255.255.252.0
```

**Task 3 – Configuring Manual Summarization on R3 for the 203.X.X.0/24 major networks using the Longest Summary Mask. Summarization should be configured towards all the neighbors.**

**R3**

```
Interface E 0/0
Ip summary-address eigrp 111 203.1.4.0 255.255.252.0
!
Interface E 0/1
Ip summary-address eigrp 111 203.1.4.0 255.255.252.0
!
Interface E 0/2
Ip summary-address eigrp 111 203.1.4.0 255.255.252.0
```

**Task 4 – Configuring Manual Summarization on R4 for the 104.0.0.0/24 subnets using the Longest Summary Mask. Summarization should be configured towards all the neighbors.**

**R4**

```
Interface E 0/0
Ip summary-address eigrp 111 104.1.8.0 255.255.252.0
!
Interface E 0/1
Ip summary-address eigrp 111 104.1.8.0 255.255.252.0
!
Interface E 0/2
Ip summary-address eigrp 111 104.1.8.0 255.255.252.0
```

**Task 5 – Configuring Manual Summarization on R5 for the 205.X.X.0/24 major networks using the Longest Summary Mask. Summarization should be configured towards all the neighbors.**

**R5**

```
Interface E 0/0
Ip summary-address eigrp 111 205.1.4.0 255.255.252.0
!
Interface E 0/1
Ip summary-address eigrp 111 205.1.4.0 255.255.252.0
```

**Task 6 – Configuring Manual Summarization on R6 for the 101.0.0.0/24 subnets using the Longest Summary Mask. Summarization should be configured towards all the neighbors.**

**R6**

```
Interface E 0/0
Ip summary-address eigrp 111 101.1.60.0 255.255.252.0
!
Interface E 0/1
Ip summary-address eigrp 111 101.1.60.0 255.255.252.0
```



**Task 7 – Configuring Manual Summarization on R7 for the 107.0.0.0/24 subnets using the Longest Summary Mask. Summarization should be configured towards all the neighbors.**

**R7**

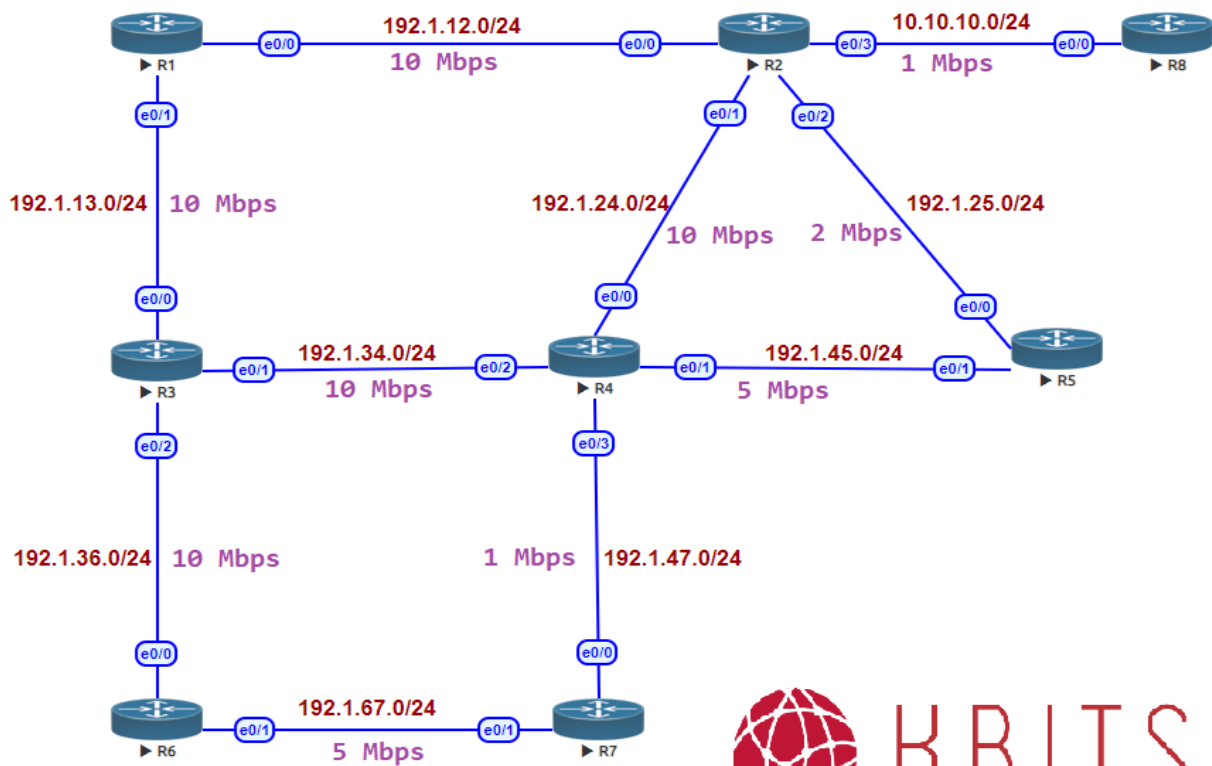
```
Interface E 0/0
Ip summary-address eigrp 111 107.1.72.0 255.255.252.0
!
Interface E 0/1
Ip summary-address eigrp 111 107.1.72.0 255.255.252.0
```

**Task 8 – Configuring Manual Summarization on R8 for all the 10.0.0.0/8 subnets & the 108.0.0.0/8 subnets using the Longest Summary Mask. Summarization should be configured towards all the neighbors.**

**R8**

```
Interface E 0/0
Ip summary-address eigrp 111 10.1.8.0 255.255.252.0
Ip summary-address eigrp 111 108.1.4.0 255.255.252.0
```

# Lab 8 – Route Summarization – Leak Maps

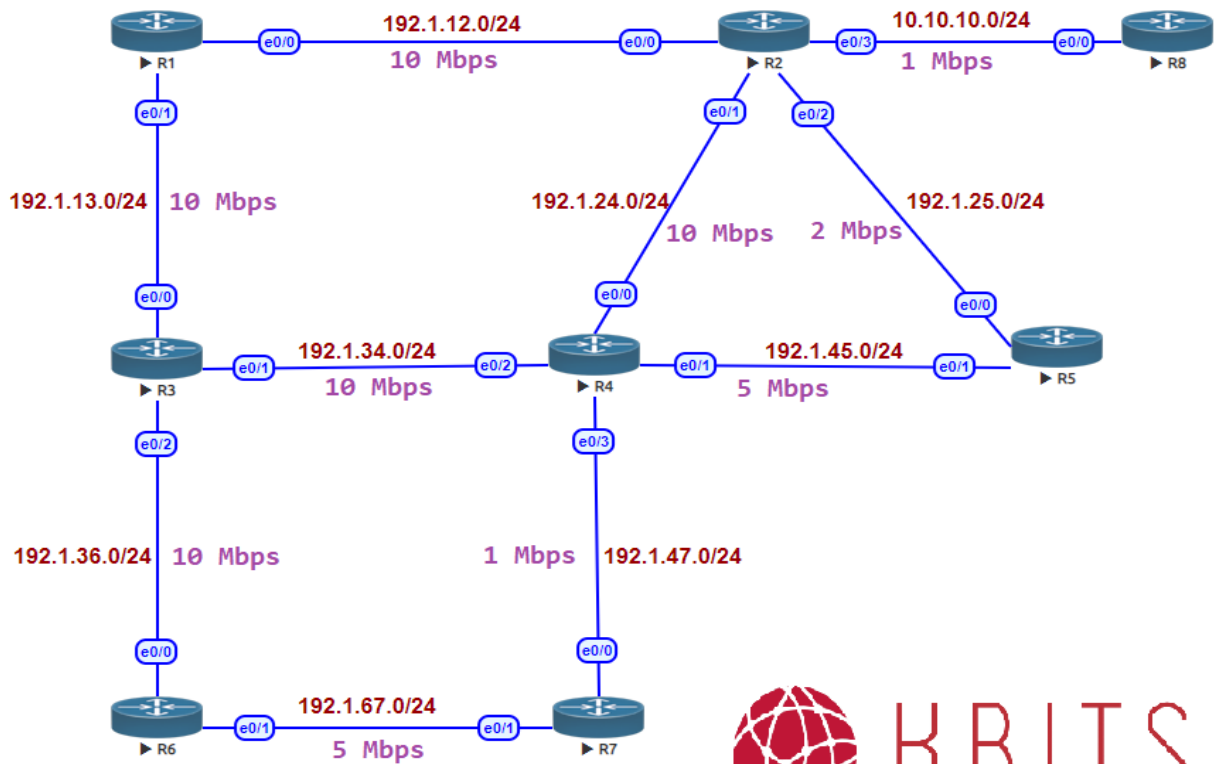


**Task 1 – Configure R7 such that traffic for the 101.1.75.0/24 network from R3 should use R4 as the Next-Hop whereas the other networks should continue to use the Summary Route based on the best metric.**

**R7**

```
Access-list 77 permit 101.1.75.0 0.0.0.255
!  
Route-map LM  
  Match ip address 77  
!  
Interface E 0/0  
Ip summary-address eigrp 111 101.1.72.0 255.255.252.0 leak-map LM
```

# Lab 9 – Route Filtering using ACLs in EIGRP



### Task 1

Configure the following Loopback Interfaces on R8:

Loopback 21: 178.1.1.1/24  
Loopback 22: 178.1.2.1/24  
Loopback 23: 178.1.3.1/24  
Loopback 24: 178.1.4.1/24

Enable them in EIGRP 111

#### R8

```
Interface loopback401
Ip address 178.1.1.1 255.255.255.0
Interface loopback402
Ip address 178.1.2.1 255.255.255.0
Interface loopback403
Ip address 178.1.3.1 255.255.255.0
Interface loopback404
Ip address 178.1.4.1 255.255.255.0
!
Router eigrp 111
Network 178.1.0.0
```

### Task 2

R2 should block the 178.1.1.0/24 & 178.1.4.0/24 networks from coming in from R8.

#### R2

```
Access-list 1 deny 178.1.1.0 0.0.0.255
Access-list 1 deny 178.1.4.0 0.0.0.255
Access-list 1 permit any
!
Router EIGRP 111
Distribute-list 1 in E 0/3
```

### Task 3

Configure the following Loopback Interfaces on R3:

Loopback 21: 173.1.1.1/24  
Loopback 22: 173.1.2.1/24  
Loopback 23: 173.1.3.1/24  
Loopback 24: 173.1.4.1/24  
Loopback 25: 173.1.5.1/24

Loopback 26: 173.1.6.1/24  
Loopback 27: 173.1.7.1/24  
Loopback 28: 173.1.8.1/24

Enable them in EIGRP 111

### **R3**

```
Interface loopback401
Ip address 173.1.1.1 255.255.255.0
Interface loopback402
Ip address 173.1.2.1 255.255.255.0
Interface loopback403
Ip address 173.1.3.1 255.255.255.0
Interface loopback404
Ip address 173.1.4.1 255.255.255.0
Interface loopback405
Ip address 173.1.5.1 255.255.255.0
Interface loopback406
Ip address 173.1.6.1 255.255.255.0
Interface loopback407
Ip address 173.1.7.1 255.255.255.0
Interface loopback408
Ip address 173.1.8.1 255.255.255.0
!
Router eigrp 111
Network 173.1.0.0
```

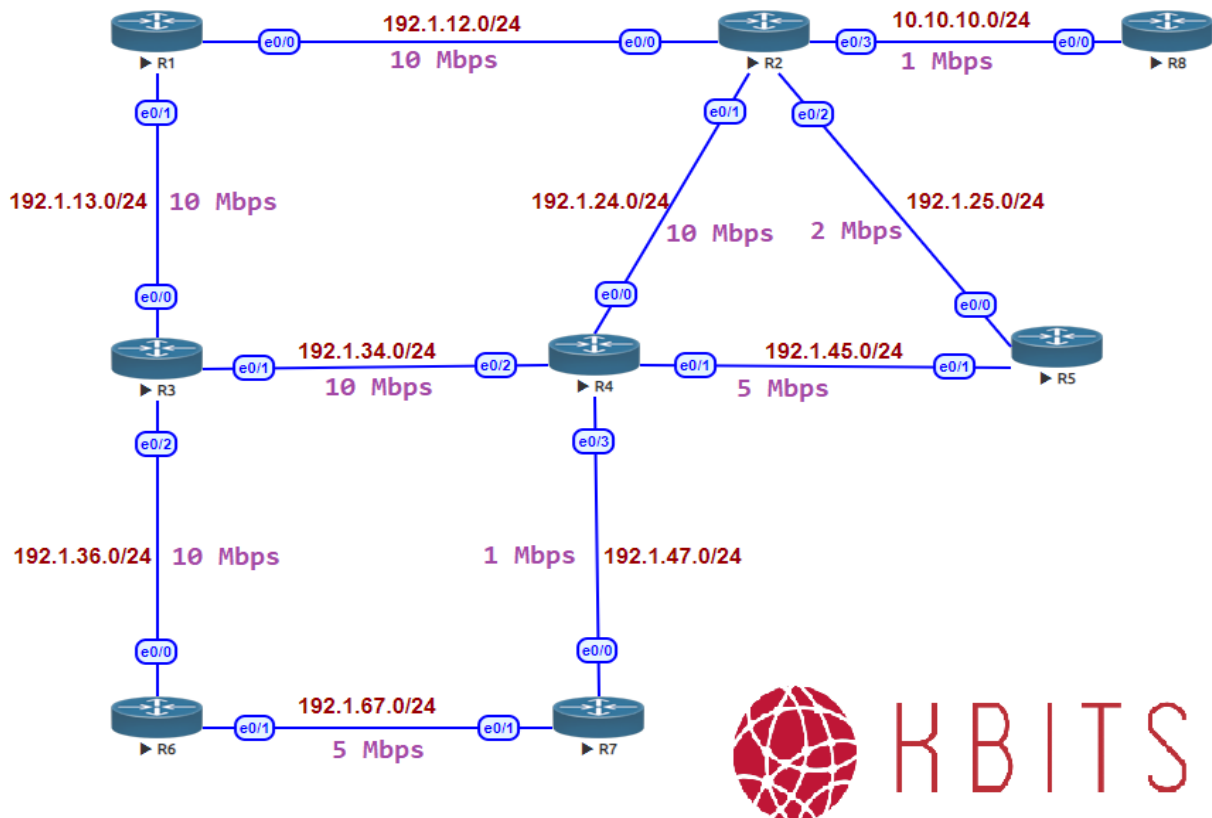
### **Task 4**

R3 should only send routes from the 173.1.X.0 range that have an even number in the 3<sup>rd</sup> Octet to all its neighbors. Use the minimum number of lines possible to accomplish this task.

### **R3**

```
Access-list 1 deny 173.1.1.0 0.0.254.255
Access-list 1 permit any
!
Router EIGRP 111
Distribute-list 1 out
```

# Lab 10 – Route Filtering using Prefix-Lists in EIGRP



## Task 1

Configure the following Loopback Interfaces on R5 and advertise them under EIGRP 111:

Loopback 401: 175.50.1.1/24  
Loopback 402: 175.50.2.1/24  
Loopback 403: 175.50.3.1/24  
Loopback 404: 205.1.1.33/27  
Loopback 405: 205.1.1.65/28

## R5

```
Interface loopback401
```

```
Ip address 175.50.1.1 255.255.255.0
Interface loopback402
Ip address 175.50.2.1 255.255.255.0
Interface loopback403
Ip address 175.50.3.1 255.255.255.0
Interface loopback404
Ip address 205.1.1.33 255.255.255.224
Interface loopback405
Ip address 205.1.1.65 255.255.255.240
!
Router eigrp 111
Network 175.50.0.0
Network 205.1.1.0
```

### **Task 2**

Configure R2 & R4 such that they receive prefixes with a prefix-length of 8 to 24 from R5. Do not configure the filtering on R5.

#### **R2**

```
ip prefix-list ABC permit 0.0.0.0/0 ge 8 le 24
!
Router EIGRP 111
distribute-list prefix ABC in E0/2
```

#### **R4**

```
ip prefix-list ABC permit 0.0.0.0/0 ge 8 le 24
!
Router EIGRP 111
distribute-list prefix ABC in E0/1
```

### **Task 3**

Configure the following Loopback Interfaces on R6 and advertise them under EIGRP 111:

```
Loopback 401: 176.1.32.1/19
Loopback 402: 176.1.64.1/20
Loopback 403: 176.1.80.1/21
Loopback 403: 176.1.96.1/24
```

#### **R6**

```
Interface loopback401
Ip address 176.1.32.1 255.255.224.0
```



```
Interface loopback402
Ip address 176.1.64.1 255.255.240.0
Interface loopback403
Ip address 176.1.80.1 255.255.248.0
Interface loopback404
Ip address 176.1.96.1 255.255.255.0
!
Router eigrp 111
Network 176.1.0.0
```

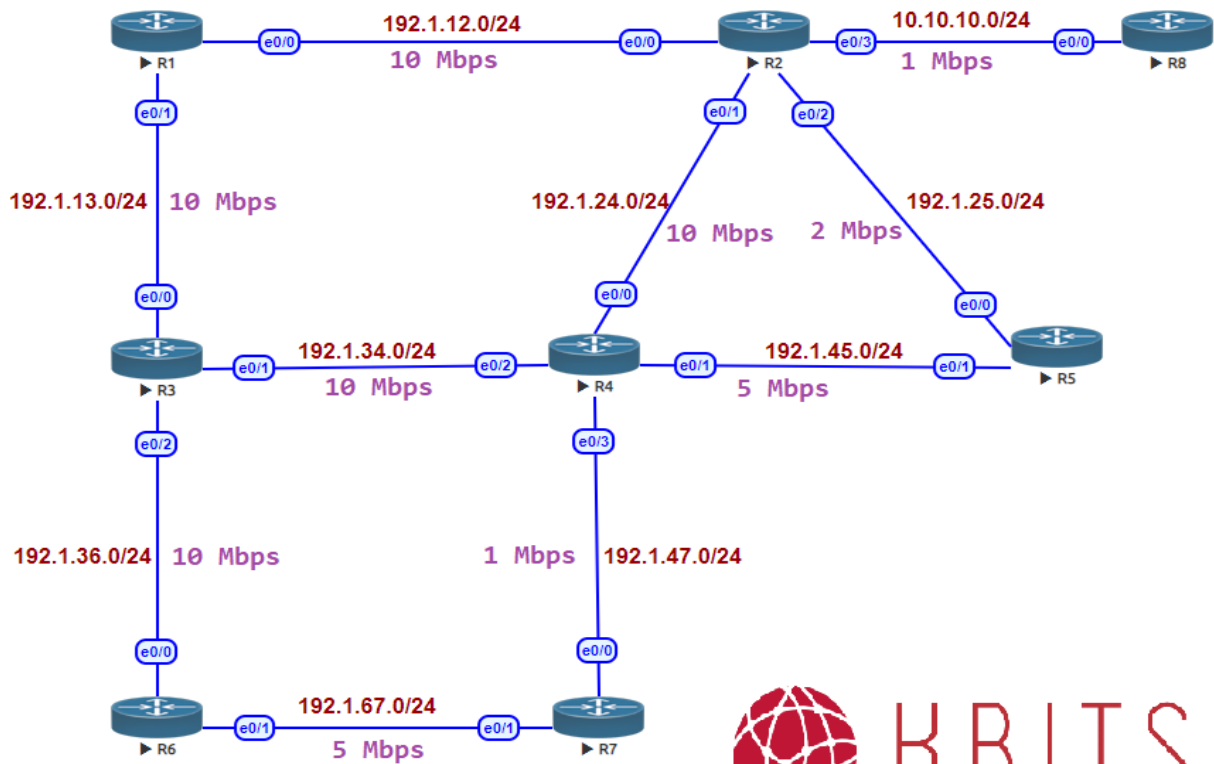
#### **Task 4**

Configure R6 such it does not send networks in the 176.1.0.0/16 major network range that have a mask greater than 20.

#### **R6**

```
ip prefix-list ABC deny 176.1.0.0/16 ge 21
ip prefix-list ABC permit 0.0.0.0/0 le 32
!
Router EIGRP 111
distribute-list prefix ABC out
```

# Lab 11 – Authenticating EIGRP Neighbors using MD5



### Task 1

Configure MD5 authentication for all links between R1, R2, R3 & R4. Use Cisco@123 as the key-string with a key-id of 1.

#### R1

```
Key chain AUTH
Key 1
  Key-string Cisco@123
!
Interface E 0/0
Ip authentication key-chain eigrp 111 AUTH
Ip authentication mode eigrp 111 MD5
!
Interface E 0/1
Ip authentication key-chain eigrp 111 AUTH
Ip authentication mode eigrp 111 MD5
```

#### R2

```
Key chain AUTH
Key 1
  Key-string Cisco@123
!
Interface E 0/0
Ip authentication key-chain eigrp 111 AUTH
Ip authentication mode eigrp 111 MD5
!
Interface E 0/1
Ip authentication key-chain eigrp 111 AUTH
Ip authentication mode eigrp 111 MD5
```

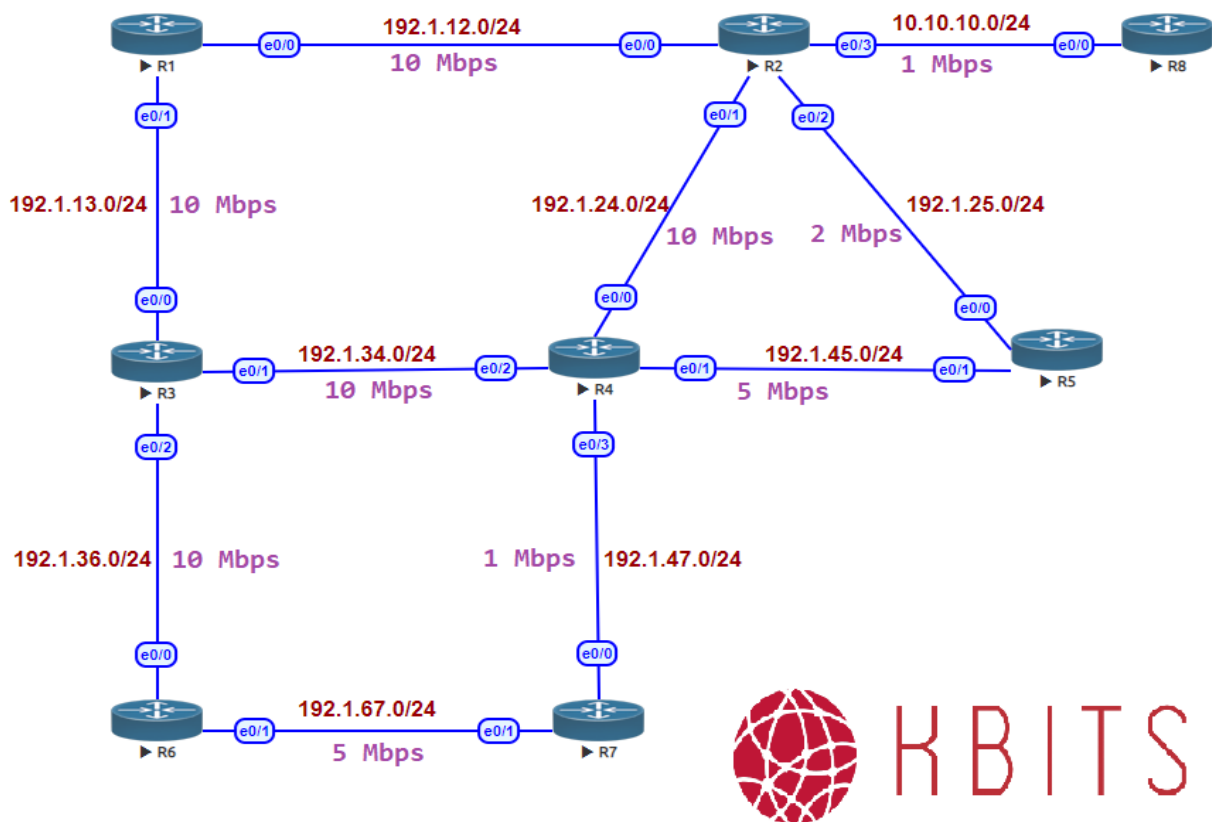
#### R3

```
Key chain AUTH
Key 1
  Key-string Cisco@123
!
Interface E 0/0
Ip authentication key-chain eigrp 111 AUTH
Ip authentication mode eigrp 111 MD5
!
Interface E 0/1
Ip authentication key-chain eigrp 111 AUTH
Ip authentication mode eigrp 111 MD5
```

#### R4

```
Key chain AUTH
Key 1
  Key-string Cisco@123
!
Interface E 0/0
  Ip authentication key-chain eigrp 111 AUTH
  Ip authentication mode eigrp 111 MD5
!
Interface E 0/2
  Ip authentication key-chain eigrp 111 AUTH
  Ip authentication mode eigrp 111 MD5
```

# Lab 12 – Configuring a Basic Named Mode Configuration



## Task 1 – Re-Configure R1, R2, R3, R4, R6 & R7 Using Named-Mode EIGRP.

- Re-configure EIGRP on the specified routers using EIGRP Named-Mode. Name the EIGRP process as KBITS. Continue to use AS as 111.
- Make sure to maintain the Summarization and Filtering Configurations previously configured on the Routers.
- Ignore configuring Neighbor Authentication as it will be done in a later lab.

**R1**

```
No router eigrp 111
!  
Router eigrp KBITS  
Address-family ipv4 autonomous-system 111  
  network 101.0.0.0  
  network 192.1.12.0  
  network 192.1.13.0  
  neighbor 192.1.12.2 Ethernet0/0  
!  
  af-interface default  
  passive-interface default  
!  
  af-interface E0/0  
  no passive-interface  
  summary-address 101.1.4.0 255.255.252.0  
!  
  af-interface E0/1  
  no passive-interface  
  summary-address 101.1.4.0 255.255.252.0
```

**R2**

```
No router eigrp 111
!  
Router eigrp KBITS  
Address-family ipv4 autonomous-system 111  
  network 10.1.4.0 0.0.0.255  
  network 10.1.5.0 0.0.0.255  
  network 10.1.6.0 0.0.0.255  
  network 10.1.7.0 0.0.0.255  
  network 10.10.10.0 0.0.0.255  
  network 102.0.0.0  
  network 192.1.12.0  
  network 192.1.24.0  
  network 192.1.25.0  
  network 202.1.4.0  
  network 202.1.5.0  
  network 202.1.6.0  
  network 202.1.7.0  
  neighbor 192.1.12.1 Ethernet0/0  
  neighbor 10.10.10.8 Ethernet0/3  
!  
Topology base  
distribute-list 1 in Ethernet0/3
```

```
distribute-list prefix ABC in Ethernet0/2
variance 3
!
af-interface default
  passive-interface default
!
af-interface E0/0
  no passive-interface
  summary-address 10.1.4.0 255.255.252.0
  summary-address 102.1.4.0 255.255.252.0
  summary-address 202.1.4.0 255.255.252.0
!
af-interface E0/1
  no passive-interface
  summary-address 10.1.4.0 255.255.252.0
  summary-address 102.1.4.0 255.255.252.0
  summary-address 202.1.4.0 255.255.252.0
!
af-interface E0/2
  no passive-interface
  summary-address 10.1.4.0 255.255.252.0
  summary-address 102.1.4.0 255.255.252.0
  summary-address 202.1.4.0 255.255.252.0
!
af-interface E0/3
  no passive-interface
  summary-address 10.1.4.0 255.255.252.0
  summary-address 102.1.4.0 255.255.252.0
  summary-address 202.1.4.0 255.255.252.0
```

### R3

```
No router eigrp 111
!
Router eigrp KBITS
Address-family ipv4 autonomous-system 111
  network 173.1.0.0
  network 192.1.13.0
  network 192.1.34.0
  network 192.1.36.0
  network 203.1.0.0 0.0.255.255
!
Topology base
  distribute-list 1 out
!
af-interface default
  passive-interface default
!
af-interface E0/0
  no passive-interface
  summary-address 203.1.4.0 255.255.252.0
!
af-interface E0/1
  no passive-interface
  summary-address 203.1.4.0 255.255.252.0
!
af-interface E0/2
  no passive-interface
  summary-address 203.1.4.0 255.255.252.0
```



## R4

```
No router eigrp 111
!
Router eigrp KBITS
Address-family ipv4 autonomous-system 111
  network 104.1.0.0 0.0.255.255
  network 192.1.24.0
  network 192.1.34.0
  network 192.1.45.0
  network 192.1.47.0
!
Topology base
  distribute-list prefix ABC in Ethernet0/1
!
af-interface default
  passive-interface default
!
af-interface E0/0
  no passive-interface
  summary-address 104.1.8.0 255.255.252.0
!
af-interface E0/1
  no passive-interface
  summary-address 104.1.8.0 255.255.252.0
!
af-interface E0/2
  no passive-interface
  summary-address 104.1.8.0 255.255.252.0
!
af-interface E0/3
  no passive-interface
  summary-address 104.1.8.0 255.255.252.0
```

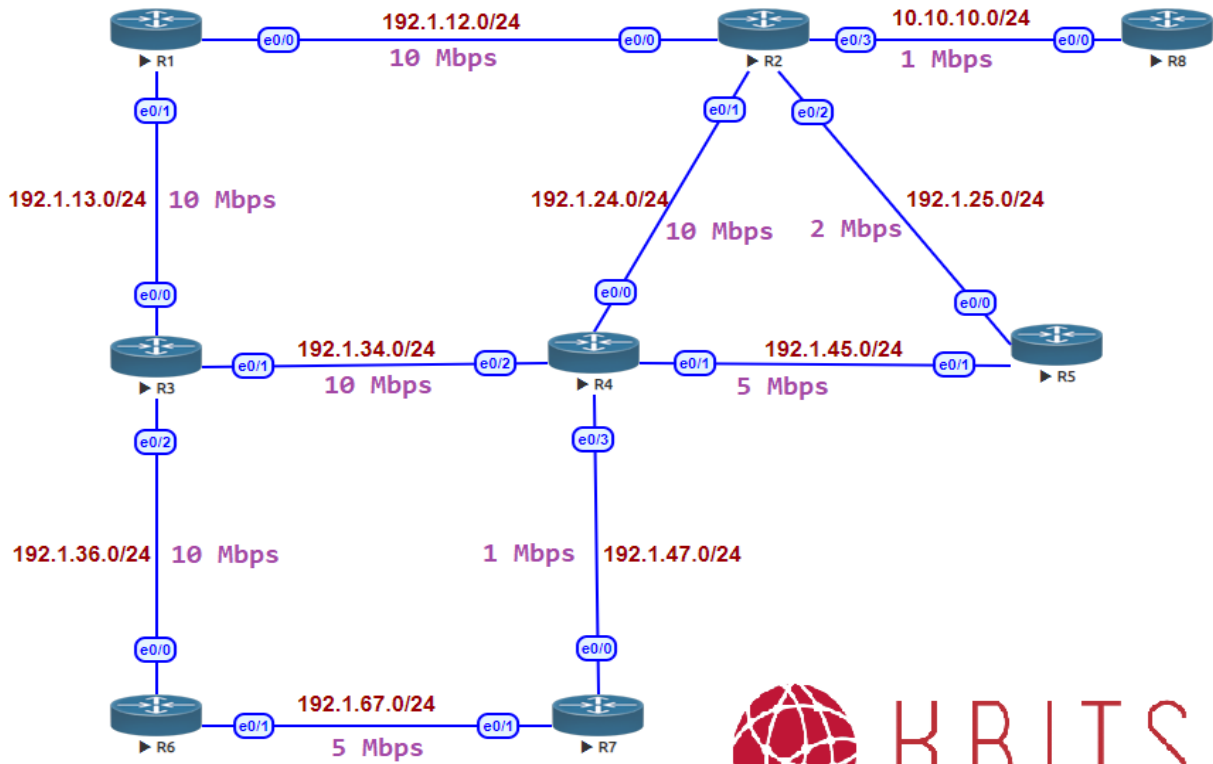
**R6**

```
No router eigrp 111
!
Router eigrp KBITS
Address-family ipv4 autonomous-system 111
  network 101.1.0.0 0.0.255.255
  network 176.1.0.0
  network 192.1.36.0
  network 192.1.67.0
!
Topology base
  distribute-list prefix ABC out
!
af-interface default
  passive-interface default
!
af-interface E0/0
  no passive-interface
  summary-address 101.1.60.0 255.255.252.0
!
af-interface E0/1
  no passive-interface
  summary-address 101.1.60.0 255.255.252.0
```

**R7**

```
No router eigrp 111
!
Router eigrp KBITS
Address-family ipv4 autonomous-system 111
  network 101.1.0.0 0.0.255.255
  network 192.1.47.0
  network 192.1.67.0
!
af-interface default
  passive-interface default
!
af-interface E0/0
  no passive-interface
  summary-address 101.1.72.0 255.255.252.0 leak-map LM
!
af-interface E0/1
  no passive-interface
  summary-address 101.1.72.0 255.255.252.0 leak-map LM
```

# Lab 13 – Configuring Authentication - SHA



### Task 1

Configure SHA authentication for all links between R1, R2, R3 & R4. Use Cisco@123 as the key-string.

#### R1

```
Router eigrp KBITS
Address-family ipv4 autonomous-system 111
!
Af-interface E 0/0
  authentication mode hmac-sha-256 Cisco@123
!
Af-interface E 0/1
  authentication mode hmac-sha-256 Cisco@123
```

#### R2

```
Router eigrp KBITS
Address-family ipv4 autonomous-system 111
!
Af-interface E 0/0
  authentication mode hmac-sha-256 Cisco@123
!
Af-interface E 0/1
  authentication mode hmac-sha-256 Cisco@123
```

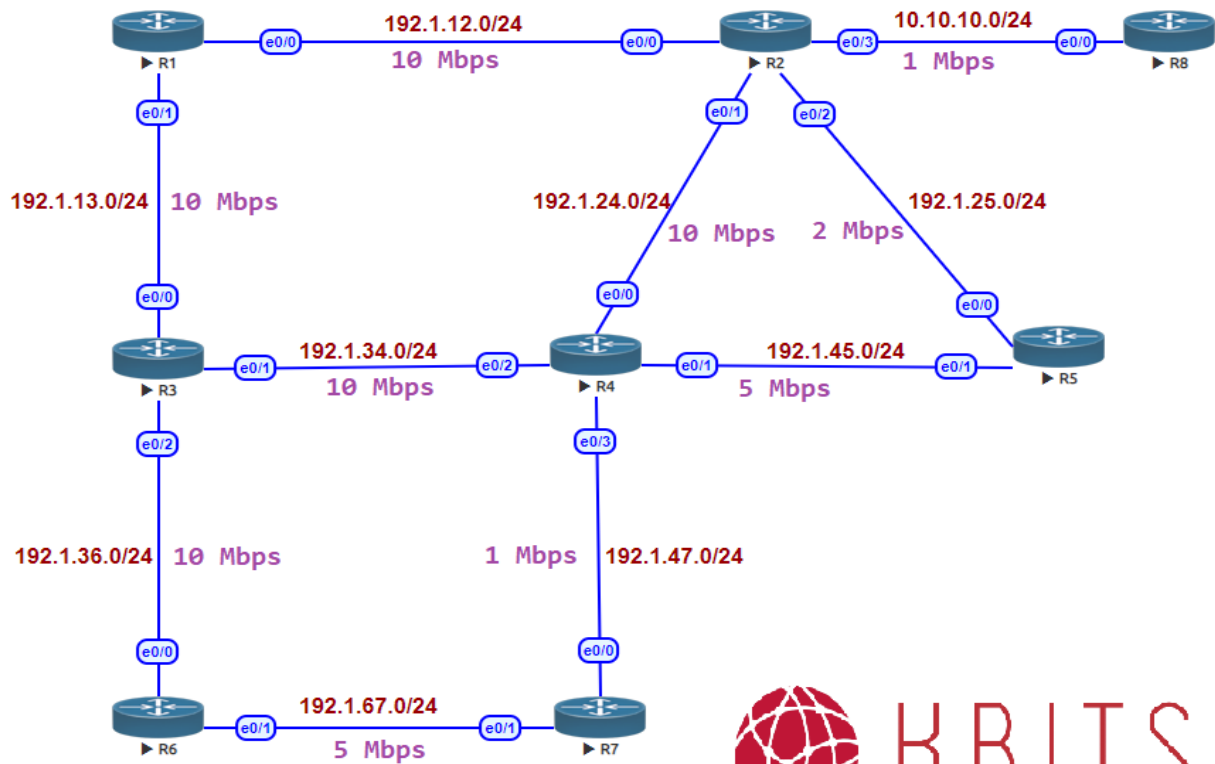
#### R3

```
Router eigrp KBITS
Address-family ipv4 autonomous-system 111
!
Af-interface E 0/0
  authentication mode hmac-sha-256 Cisco@123
!
Af-interface E 0/1
  authentication mode hmac-sha-256 Cisco@123
```

#### R4

```
Router eigrp KBITS
Address-family ipv4 autonomous-system 111
!
Af-interface E 0/0
  authentication mode hmac-sha-256 Cisco@123
!
Af-interface E 0/2
  authentication mode hmac-sha-256 Cisco@123
```

# Lab 14 - Configuring Authentication - MD5



### Task 1

Configure MD5 authentication between R4 & R7. Use a Key Chain name of R4R7 with a key string of 47 and a key string of Cisco@47.

#### R4

```
Key chain R4R7
Key 47
  Key-string Cisco@47
!
Router eigrp KBITS
Address-family ipv4 autonomous-system 111
!
Af-interface E 0/3
  authentication mode md5
  authentication key-chain R4R7
```

#### R7

```
Key chain R4R7
Key 47
  Key-string Cisco@47
!
Router eigrp KBITS
Address-family ipv4 autonomous-system 111
!
Af-interface E 0/0
  authentication mode md5
  authentication key-chain R4R7
```

### Task 2

Configure MD5 authentication between R2 & R8. Use a Key Chain R2R8 with a key string of 28 and a key string of Cisco@28.

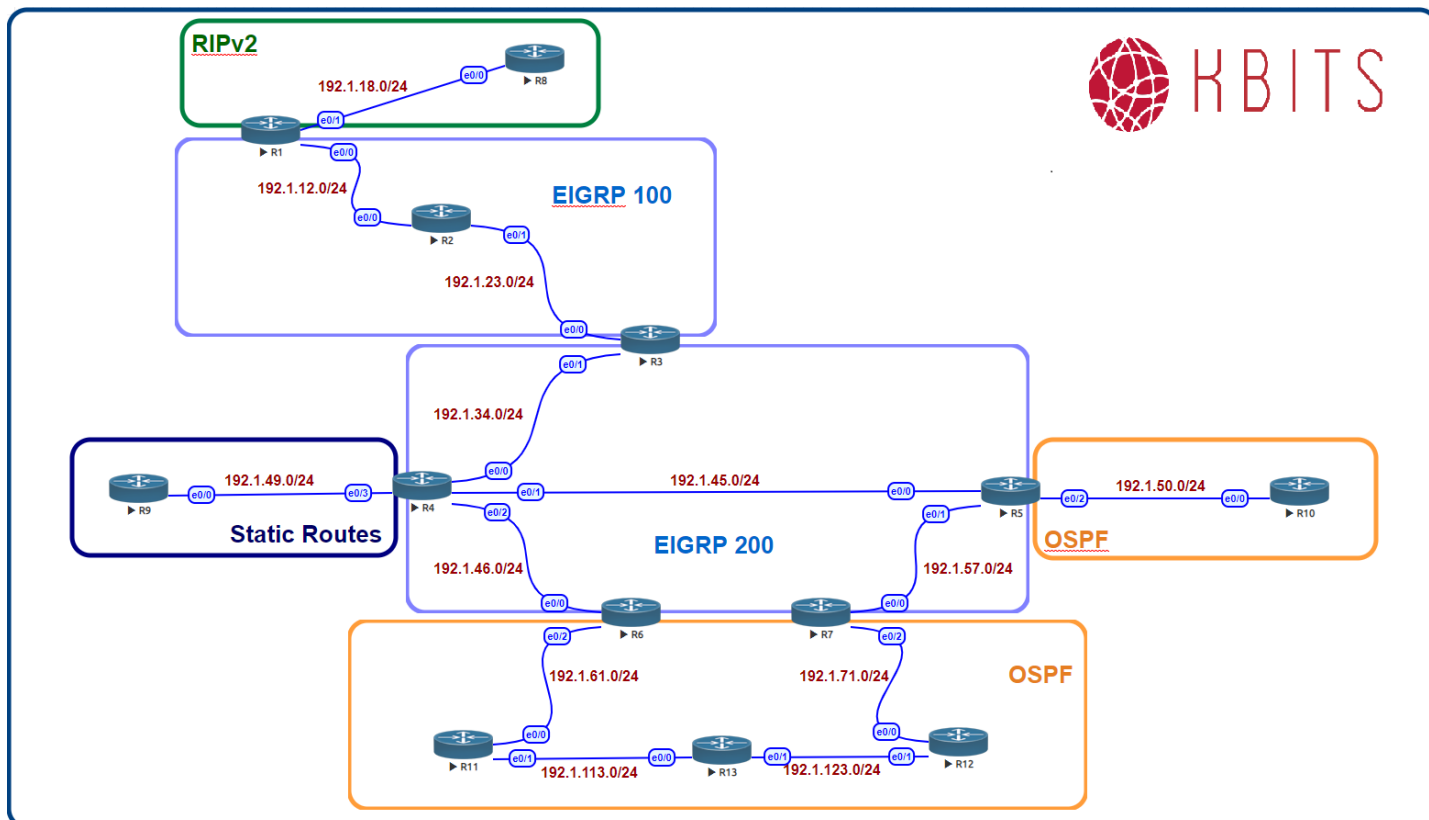
#### R2

```
Key chain R2R8
Key 28
  Key-string Cisco@28
!
Router eigrp KBITS
Address-family ipv4 autonomous-system 111
!
Af-interface E 0/3
  authentication mode md5
  authentication key-chain R2R8
```

**R8**

```
Key chain R2R8
Key 28
  Key-string Cisco@28
!
Router eigrp 111
ip authentication mode eigrp 111 md5
ip authentication key-chain eigrp 111 R4R7
```

# Lab 15 – Configuring a Multi-Domain Network



## Interface Configuration

### R1

Interface	IP Address	Subnet Mask
E 0/0	192.1.12.1	255.255.255.0
E 0/1	192.1.18.1	255.255.255.0
Loopback1	201.1.4.1	255.255.255.0
Loopback2	201.1.5.1	255.255.255.0
Loopback3	201.1.6.1	255.255.255.0
Loopback4	201.1.7.1	255.255.255.0

### R2

Interface	IP Address	Subnet Mask
E 0/0	192.1.12.2	255.255.255.0
E 0/1	192.1.23.2	255.255.255.0



Loopback1	202.1.4.1	255.255.255.0
Loopback2	202.1.5.1	255.255.255.0
Loopback3	202.1.6.1	255.255.255.0
Loopback4	202.1.7.1	255.255.255.0

### R3

Interface	IP Address	Subnet Mask
E 0/0	192.1.23.3	255.255.255.0
E 0/1	192.1.34.3	255.255.255.0
Loopback1	203.1.4.1	255.255.255.0
Loopback2	203.1.5.1	255.255.255.0
Loopback3	203.1.6.1	255.255.255.0
Loopback4	203.1.7.1	255.255.255.0
Loopback5	10.1.32.1	255.255.255.0
Loopback6	10.1.33.1	255.255.255.0
Loopback7	10.1.34.1	255.255.255.0
Loopback8	10.1.35.1	255.255.255.0

### R4

Interface	IP Address	Subnet Mask
E 0/0	192.1.34.4	255.255.255.0
E 0/1	192.1.45.4	255.255.255.0
E 0/2	192.1.46.4	255.255.255.0
E 0/3	192.1.49.4	255.255.255.0
Loopback1	204.1.4.1	255.255.255.0
Loopback2	204.1.5.1	255.255.255.0
Loopback3	204.1.6.1	255.255.255.0
Loopback4	204.1.7.1	255.255.255.0

### R5

Interface	IP Address	Subnet Mask
E 0/0	192.1.45.5	255.255.255.0
E 0/1	192.1.57.5	255.255.255.0
E 0/2	192.1.50.5	255.255.255.0
Loopback1	205.1.4.1	255.255.255.0
Loopback2	205.1.5.1	255.255.255.0
Loopback3	205.1.6.1	255.255.255.0
Loopback4	205.1.7.1	255.255.255.0
Loopback5	10.1.56.1	255.255.255.0
Loopback6	10.1.57.1	255.255.255.0

Loopback7	10.1.58.1	255.255.255.0
Loopback8	10.1.59.1	255.255.255.0

## R6

Interface	IP Address	Subnet Mask
E 0/0	192.1.46.6	255.255.255.0
E 0/1	192.1.61.6	255.255.255.0
Loopback1	206.1.4.1	255.255.255.0
Loopback2	206.1.5.1	255.255.255.0
Loopback3	206.1.6.1	255.255.255.0
Loopback4	206.1.7.1	255.255.255.0
Loopback5	10.1.60.1	255.255.255.0
Loopback6	10.1.61.1	255.255.255.0
Loopback7	10.1.62.1	255.255.255.0
Loopback8	10.1.63.1	255.255.255.0

## R7

Interface	IP Address	Subnet Mask
E 0/0	192.1.57.7	255.255.255.0
E 0/1	192.1.71.7	255.255.255.0
Loopback1	207.1.4.1	255.255.255.0
Loopback2	207.1.5.1	255.255.255.0
Loopback3	207.1.6.1	255.255.255.0
Loopback4	207.1.7.1	255.255.255.0
Loopback5	10.1.72.1	255.255.255.0
Loopback6	10.1.73.1	255.255.255.0
Loopback7	10.1.74.1	255.255.255.0
Loopback8	10.1.75.1	255.255.255.0

## R8

Interface	IP Address	Subnet Mask
E 0/0	192.1.18.8	255.255.255.0
Loopback1	10.1.80.1	255.255.255.0
Loopback2	10.1.81.1	255.255.255.0
Loopback3	10.1.82.1	255.255.255.0
Loopback4	10.1.83.1	255.255.255.0

**R9**

<b>Interface</b>	<b>IP Address</b>	<b>Subnet Mask</b>
E 0/0	192.1.49.9	255.255.255.0
Loopback1	10.1.96.1	255.255.255.0
Loopback2	10.1.97.1	255.255.255.0
Loopback3	10.1.98.1	255.255.255.0
Loopback4	10.1.99.1	255.255.255.0

**R10**

<b>Interface</b>	<b>IP Address</b>	<b>Subnet Mask</b>
E 0/0	192.1.50.10	255.255.255.0
Loopback1	101.1.100.1	255.255.255.0
Loopback2	101.1.101.1	255.255.255.0
Loopback3	101.1.102.1	255.255.255.0
Loopback4	101.1.103.1	255.255.255.0

**R11**

<b>Interface</b>	<b>IP Address</b>	<b>Subnet Mask</b>
E 0/0	192.1.61.11	255.255.255.0
E 0/1	192.1.113.11	255.255.255.0
Loopback1	10.1.112.1	255.255.255.0
Loopback2	10.1.113.1	255.255.255.0
Loopback3	10.1.114.1	255.255.255.0
Loopback4	10.1.115.1	255.255.255.0

**R12**

<b>Interface</b>	<b>IP Address</b>	<b>Subnet Mask</b>
E 0/0	192.1.71.12	255.255.255.0
E 0/1	192.1.123.12	255.255.255.0
Loopback1	10.1.120.1	255.255.255.0
Loopback2	10.1.121.1	255.255.255.0
Loopback3	10.1.122.1	255.255.255.0
Loopback4	10.1.123.1	255.255.255.0

**R13**

<b>Interface</b>	<b>IP Address</b>	<b>Subnet Mask</b>
E 0/0	192.1.113.13	255.255.255.0
E 0/1	192.1.123.13	255.255.255.0
Loopback1	10.1.132.1	255.255.255.0

Loopback2	10.1.133.1	255.255.255.0
Loopback3	10.1.134.1	255.255.255.0
Loopback4	10.1.135.1	255.255.255.0

### **Task 1 - Configure RIPv2 between R1 & R8.**

- Run RIPv2 between R1 & R8.
- Enable all interfaces on R8 under RIPv2.
- Enable all interfaces on R1 under RIPv2 except the R1-R2 link (192.1.12.0/24 Network).
- Disable Auto-summary for both routers.

#### **R1**

```
router rip
version 2
no auto-summary
network 192.1.18.0
network 201.1.4.0
network 201.1.5.0
network 201.1.6.0
network 201.1.7.0
```

#### **R8**

```
router rip
version 2
no auto-summary
network 192.1.18.0
network network 10.0.0.0
```

## Task 2 – Configure EIGRP 100 between R1, R2 & R3.

- Run EIGRP 100 between R1, R2 & R3.
- Enable the R1-R2 link in EIGRP 100 on R1.
- Enable all interface on R2 under EIGRP 100.
- Enable the R2-R3 link and all the 10.0.0.0/8 subnets on R3 under EIGRP 100.

### R1

```
router eigrp 100
network 192.1.12.0
```

### R2

```
router eigrp 100
network 192.1.12.0
network 192.1.23.0
network 202.1.4.0
network 202.1.5.0
network 202.1.6.0
network 202.1.7.0
```

### R3

```
router eigrp 100
network 192.1.23.0
network 10.0.0.0
```

## Task 3 – Configure EIGRP 200 between R3, R4, R5, R6 & R7.

- Run EIGRP 200 between R3, R4, R5, R6 & R7.
- Enable the R3-R4 link & the 203.1.X.0/24 networks in EIGRP 200 on R3.
- Enable all interface on R4 under EIGRP 200 except the R4-R9 Link. Enable Passive-interface on all interfaces except on E0/0, E0/1 & E0/2.
- Enable the R4-R5 & R5-R7 link and all the 10.0.0.0/8 subnets on R5 under EIGRP 200.
- Enable the R4-R6 link and all the 10.0.0.0/8 subnets on R6 under EIGRP 200.
- Enable the R5-R7 link and all the 10.0.0.0/8 subnets on R7 under EIGRP 200.

### R3

```
router eigrp 100
```

```
network 192.1.34.0
network 203.1.4.0
network 203.1.5.0
network 203.1.6.0
network 203.1.7.0
```

#### **R4**

```
router eigrp 200
network 192.1.34.0
network 192.1.45.0
network 192.1.46.0
network 204.1.4.0
network 204.1.5.0
network 204.1.6.0
network 204.1.7.0
passive-interface default
no passive-interface E0/0
no passive-interface E0/1
no passive-interface E0/2
```

#### **R5**

```
router eigrp 200
network 192.1.45.0
network 192.1.57.0
network 10.0.0.0
```

#### **R6**

```
router eigrp 200
network 192.1.46.0
network 10.0.0.0
```

#### **R7**

```
router eigrp 200
network 192.1.57.0
network 10.0.0.0
```

### **Task 4 – Configure Static Routing between R4 & R9.**

- Configure a default route on R9 pointing towards R4.
- Create static routes on R4 for the R9 Loopback networks using R9 as the next-hop.

#### **R4**

```
Ip route 10.1.96.0 255.255.255.0 192.1.49.9
```

```
Ip route 10.1.97.0 255.255.255.0 192.1.49.9
Ip route 10.1.98.0 255.255.255.0 192.1.49.9
Ip route 10.1.99.0 255.255.255.0 192.1.49.9
```

### **R9**

```
Ip route 0.0.0.0 0.0.0.0 192.1.49.4
```

### **Task 5 – Configure OSPF in Area 0 between R5 & R10.**

- Configure all interfaces on R10 under OSPF in Area 0.
- Configure the R5-R10 Link and the 205.1.X.0/24 networks on R5 under OSPF in Area 0.

### **R5**

```
Router ospf 1
Router-id 0.0.0.5
Network 192.1.50.0 0.0.0.255 area 0
Network 205.1.4.0 0.0.0.255 area 0
Network 205.1.5.0 0.0.0.255 area 0
Network 205.1.6.0 0.0.0.255 area 0
Network 205.1.7.0 0.0.0.255 area 0
```

### **R10**

```
Router ospf 1
Router-id 0.0.0.10
Network 192.1.50.0 0.0.0.255 area 0
Network 10.1.0.0 0.0.255.255 area 0
```

### **Task 6 – Configure OSPF in Area 0 between R6, R7, R11, R12 & R13.**

- Configure the R6-R11 Link and the 206.1.X.0/24 networks on R6 under OSPF in Area 0.
- Configure all interfaces on R11 under OSPF in Area 0.
- Configure all interfaces on R12 under OSPF in Area 0.
- Configure all interfaces on R13 under OSPF in Area 0.
- Configure the R7-R12 Link and the 207.1.X.0/24 networks on R7 under OSPF in Area 0.

### **R6**

```
Router ospf 1
Router-id 0.0.0.6
Network 192.1.61.0 0.0.0.255 area 0
Network 206.1.4.0 0.0.0.255 area 0
```

```
Network 206.1.5.0 0.0.0.255 area 0
Network 206.1.6.0 0.0.0.255 area 0
Network 206.1.7.0 0.0.0.255 area 0
```

### **R7**

```
Router ospf 1
Router-id 0.0.0.7
Network 192.1.71.0 0.0.0.255 area 0
Network 207.1.4.0 0.0.0.255 area 0
Network 207.1.5.0 0.0.0.255 area 0
Network 207.1.6.0 0.0.0.255 area 0
Network 207.1.7.0 0.0.0.255 area 0
```

### **R11**

```
Router ospf 1
Router-id 0.0.0.11
Network 192.1.61.0 0.0.0.255 area 0
Network 192.1.113.0 0.0.0.255 area 0
Network 10.1.0.0 0.0.255.255 area 0
```

### **R12**

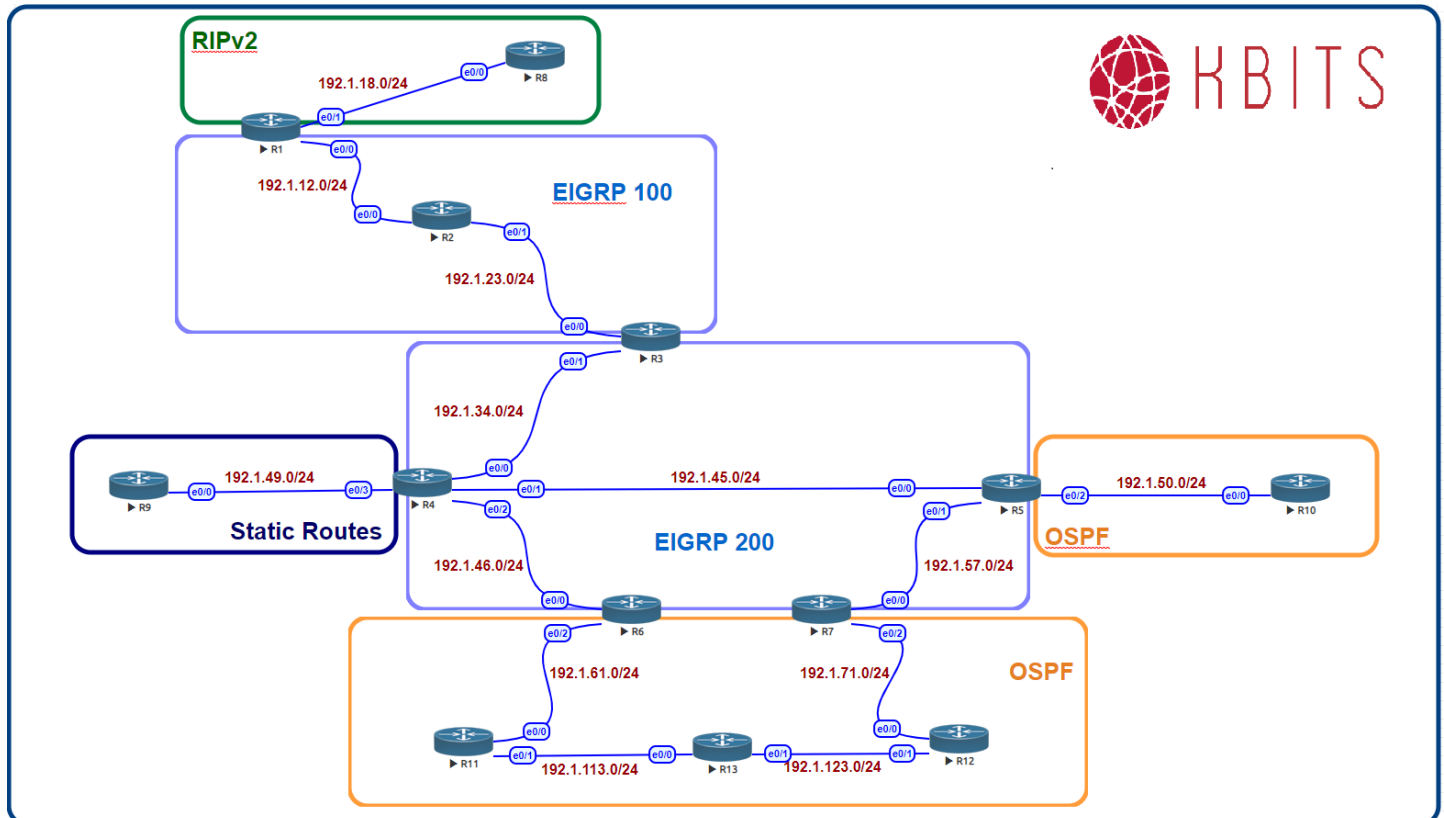
```
Router ospf 1
Router-id 0.0.0.12
Network 192.1.71.0 0.0.0.255 area 0
Network 192.1.123.0 0.0.0.255 area 0
Network 10.1.0.0 0.0.255.255 area 0
```

### **R13**

```
Router ospf 1
Router-id 0.0.0.13
Network 192.1.113.0 0.0.0.255 area 0
Network 192.1.123.0 0.0.0.255 area 0
Network 10.1.0.0 0.0.255.255 area 0
```



# Lab 16 – Redistributing Connected & Static Routes



## Task 1

You would like to provide reachability between the Static Routing and EIGRP 200 domains. Configure Route Redistribution of Static Routes on R4. R9 is already configured with a Default Route towards R4.

### R4

```
Router eigrp 200
Redistribute static
```

### Verification:

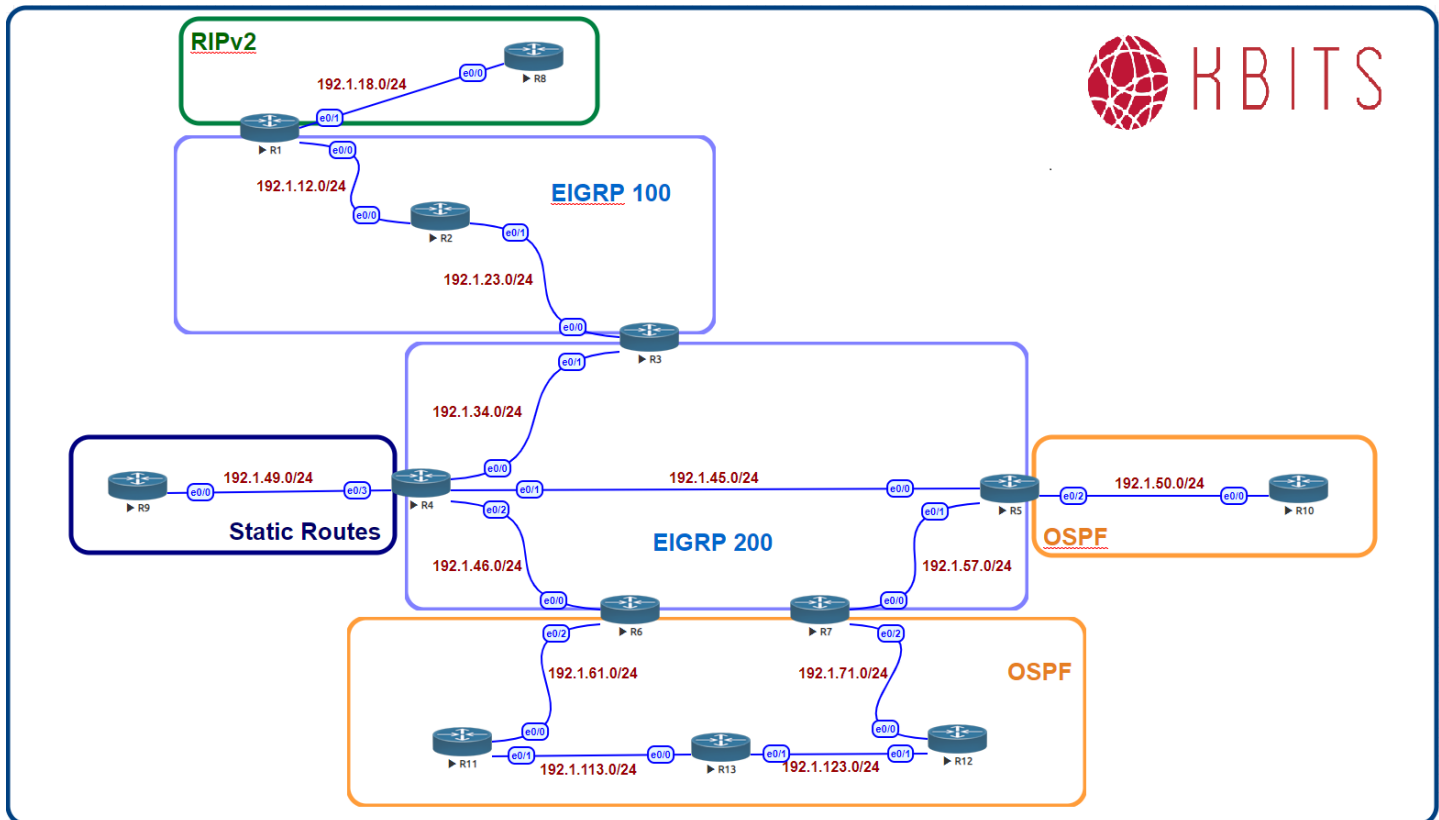
- Try Pinging 10.1.96.1 (R9's Loopback) from R5. Are you successful?
- Try Pinging 192.1.49.9 (R9's E0/0 IP) from R5. Are you successful?

- The reason is that 192.1.49.0/24 is a directly connected interface on R4. It is not enabled in EIGRP on R4. Redistribute Static does not include the Connected Route.
- Redistribute the connected route into EIGRP. Make sure to only redistribute the R4-R9 directly connected Interface.

#### **R4**

```
Route-map RC
Match interface E0/3
!
Router eigrp 200
Redistribute connected route-map RC
```

# Lab 17 - Redistributing between RIP & EIGRP



## Task 1

You would like to provide reachability between the RIPv2 and EIGRP 100 domains. Configure Mutual Route Redistribution of RIPv2 & EIGRP on R1. Use a metric of your choice.

### R1

```

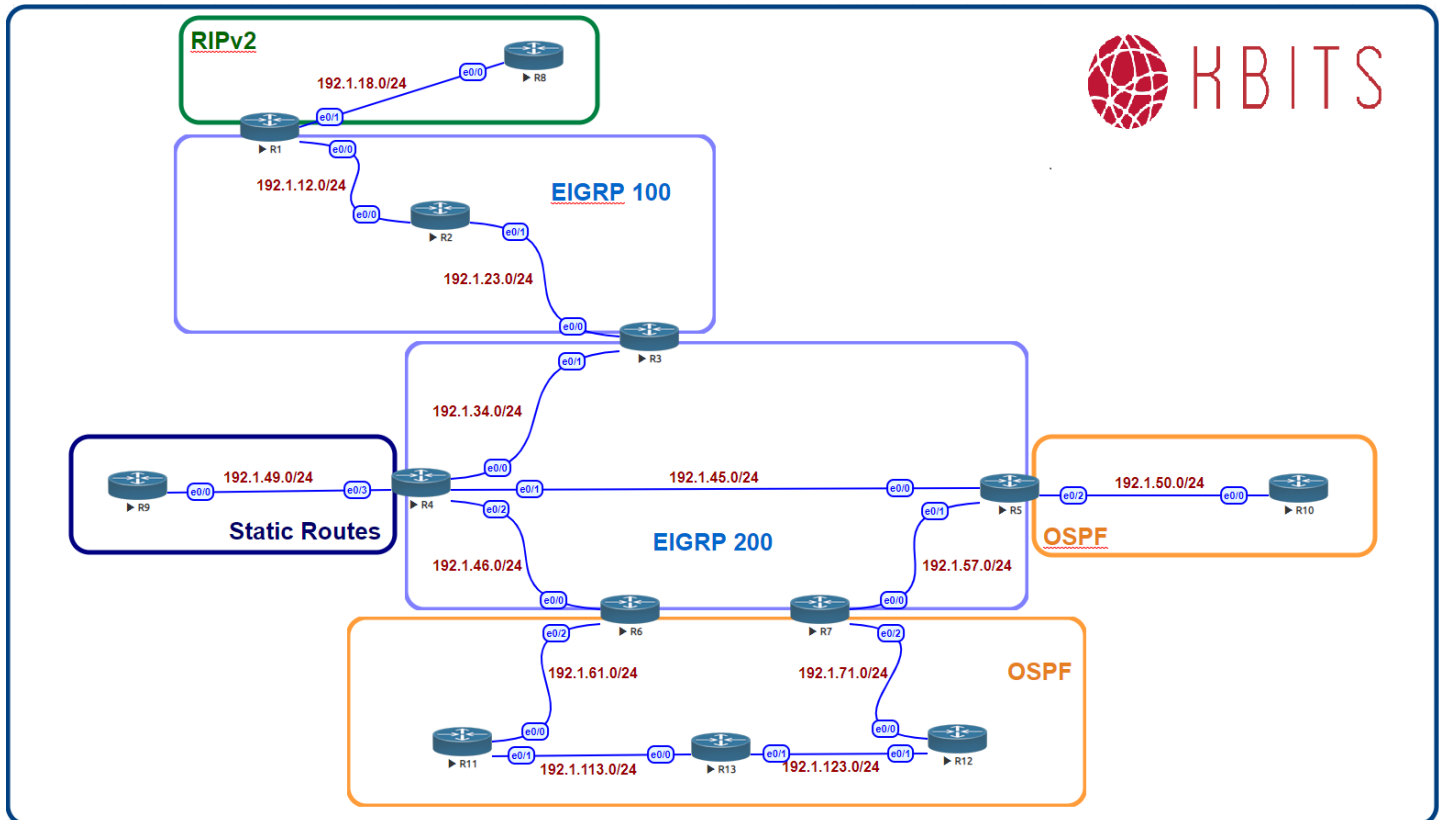
Router eigrp 100
  Redistribute rip metric 10000 1000 255 1 1500
!
Router rip
  Redistribute eigrp 100 metric 1

OR

Router eigrp 100
  Redistribute rip
  Default-metric 10000 1000 255 1 1500
    
```

```
!  
Router rip  
  Redistribute eigrp 100  
  Default-metric 1
```

# Lab 18 - Redistributing between 2 different EIGRP Autonomous-Systems



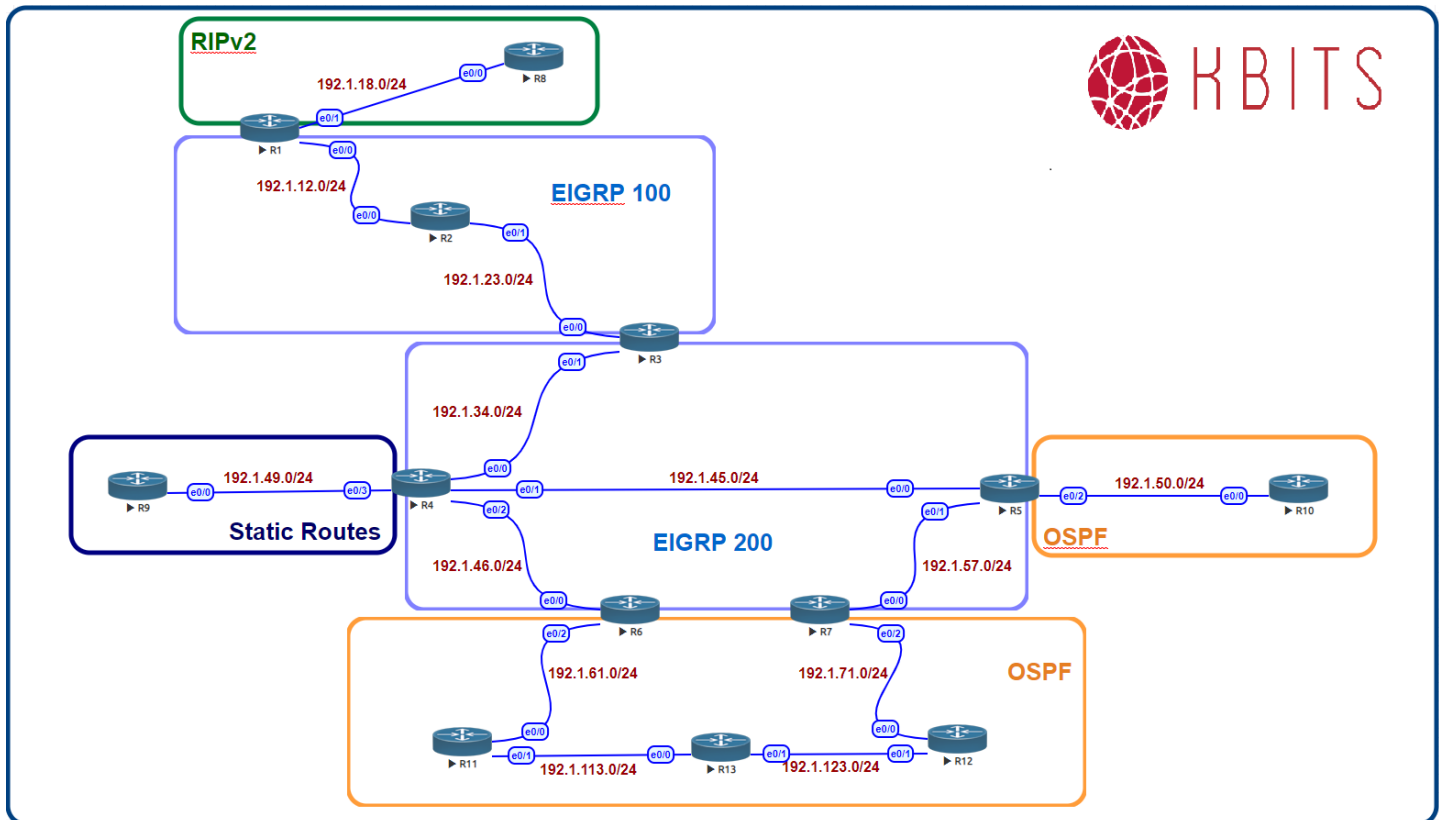
## Task 1

You would like to provide reachability between the EIGRP 100 and EIGRP 200 domains. Configure Mutual Route Redistribution on R3 to redistribute between the 2 domains.

### R3

```
Router eigrp 100
  Redistribute eigrp 200
!
Router eigrp 200
  Redistribute eigrp 100
```

# Lab 19 - Redistributing between OSPF & EIGRP



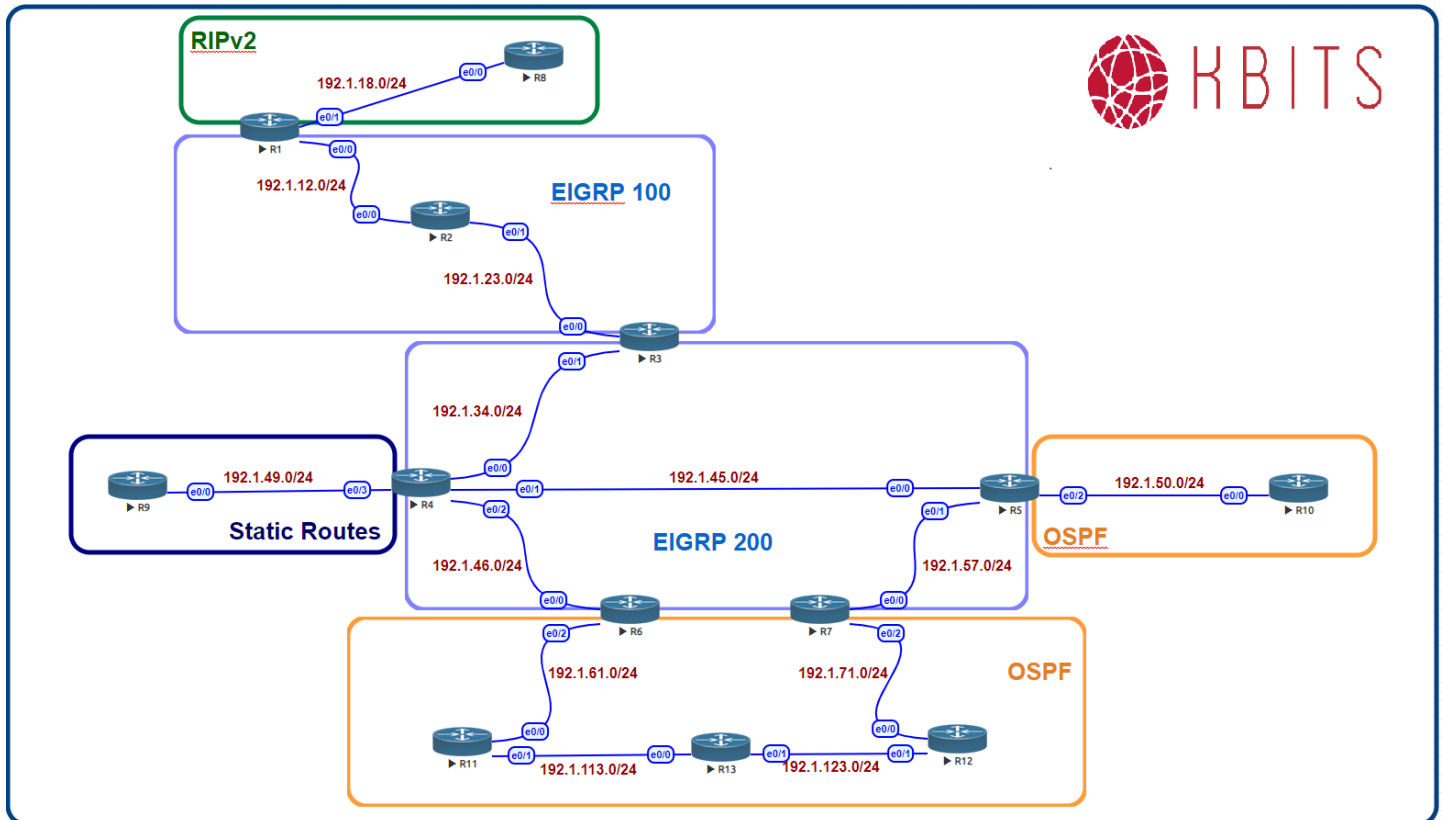
## Task 1

You would like to provide reachability between the EIGRP 200 and OSPF by performing redistribution on R5. We will look at the redistribution for the bigger OSPF domain in a later lab.

## R5

```
Router eigrp 200
  Redistribute ospf 1 metric 10 10 10 10 10
!
Router ospf 1
  Redistribute eigrp 200
```

# Lab 20 – Redistribution with Route Filtering



## Task 1

Networks 202.1.4.0/24 & 202.1.6.0/24 networks should not be redistributed into EIGRP 200. Re-configure Redistribution on R3 to fulfill the requirement.

### R3

```
Access-list 1 deny 202.1.4.0 0.0.0.255
Access-list 1 deny 202.1.6.0 0.0.0.255
Access-list 1 permit any
!
Route-map E2E
 Match ip address 1
!
Router eigrp 200
 Redistribute eigrp 100 route-map E2E
```

## Task 2

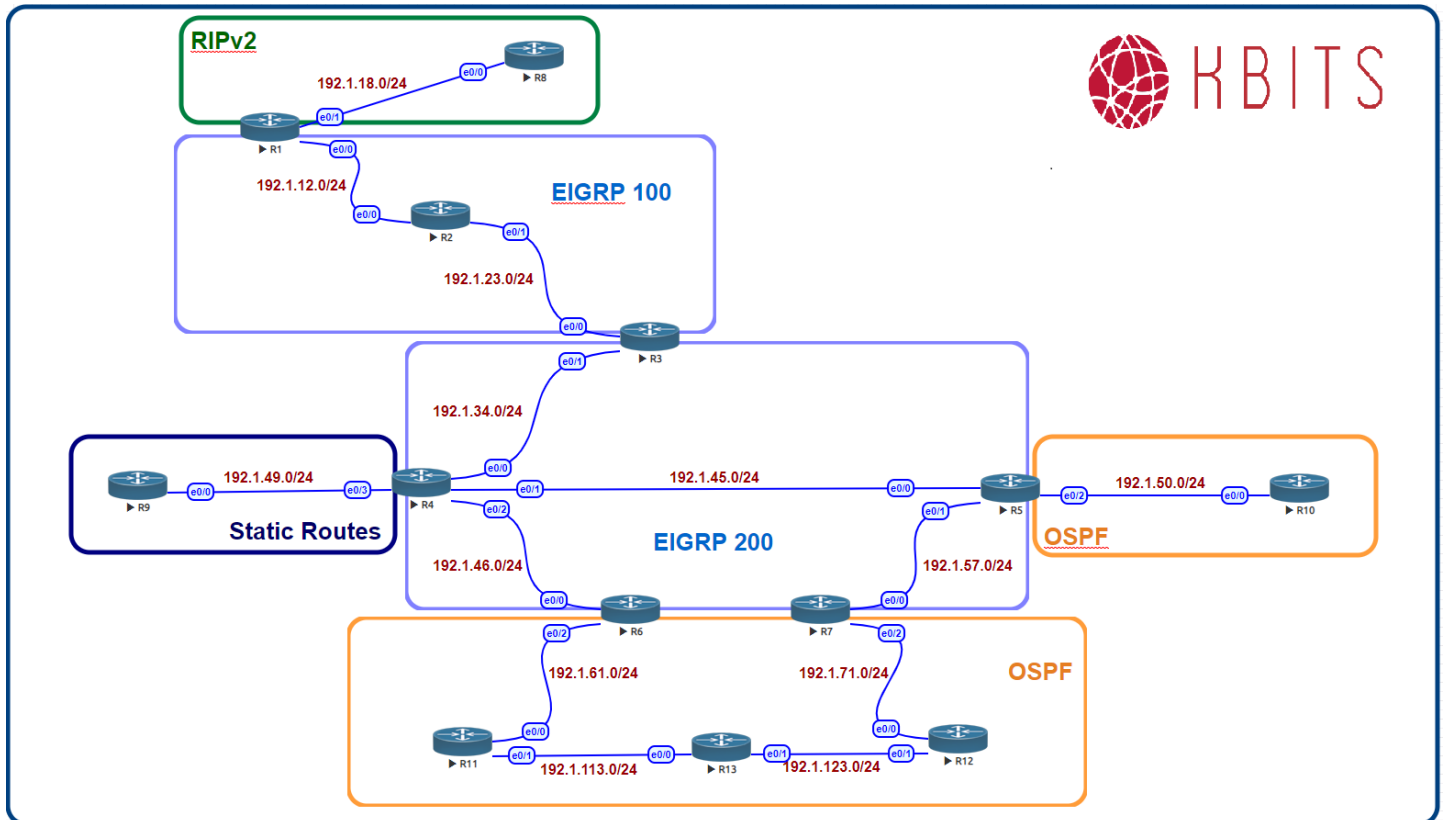
Networks 10.1.97.0/24 & 10.1.98.0/24 networks should not be redistributed into EIGRP 200. Re-configure Redistribution on R4 to fulfill the requirement.

### R4

```
Ip prefix-list PL1 deny 10.1.97.0/24
Ip prefix-list PL1 deny 10.1.98.0/24
Ip prefix-list PL1 permit 0.0.0.0/0 le 32
!
Route-map S2E
  Match ip address prefix PL1
!
Router eigrp 200
  Redistribute static route-map S2E
```



# Lab 21 – Redistribution with Route Tagging



## Task 1

You are required to block the RIPv2 routes from propagating into the OSPF domain connected to R5 and vice-versa. Use a Mechanism that ensures that if new routes are added into the RIP or OSPF domains, they continue to get blocked from propagating to each other without having to do any further configurations.

### Blocking RIP to OSPF

#### R1

```

Route-map R2E
Set tag 123
!
Router eigrp 100
Redistribute rip metric 1000 100 255 1 1500 route-map R2E
    
```

#### R5

```
Route-map E2O deny 10
  Match tag 123
Route-map E2O permit 20
!
Router ospf 1
  Redistribute eigrp 200 route-map E2O
```

### **Blocking OSPF to RIP**

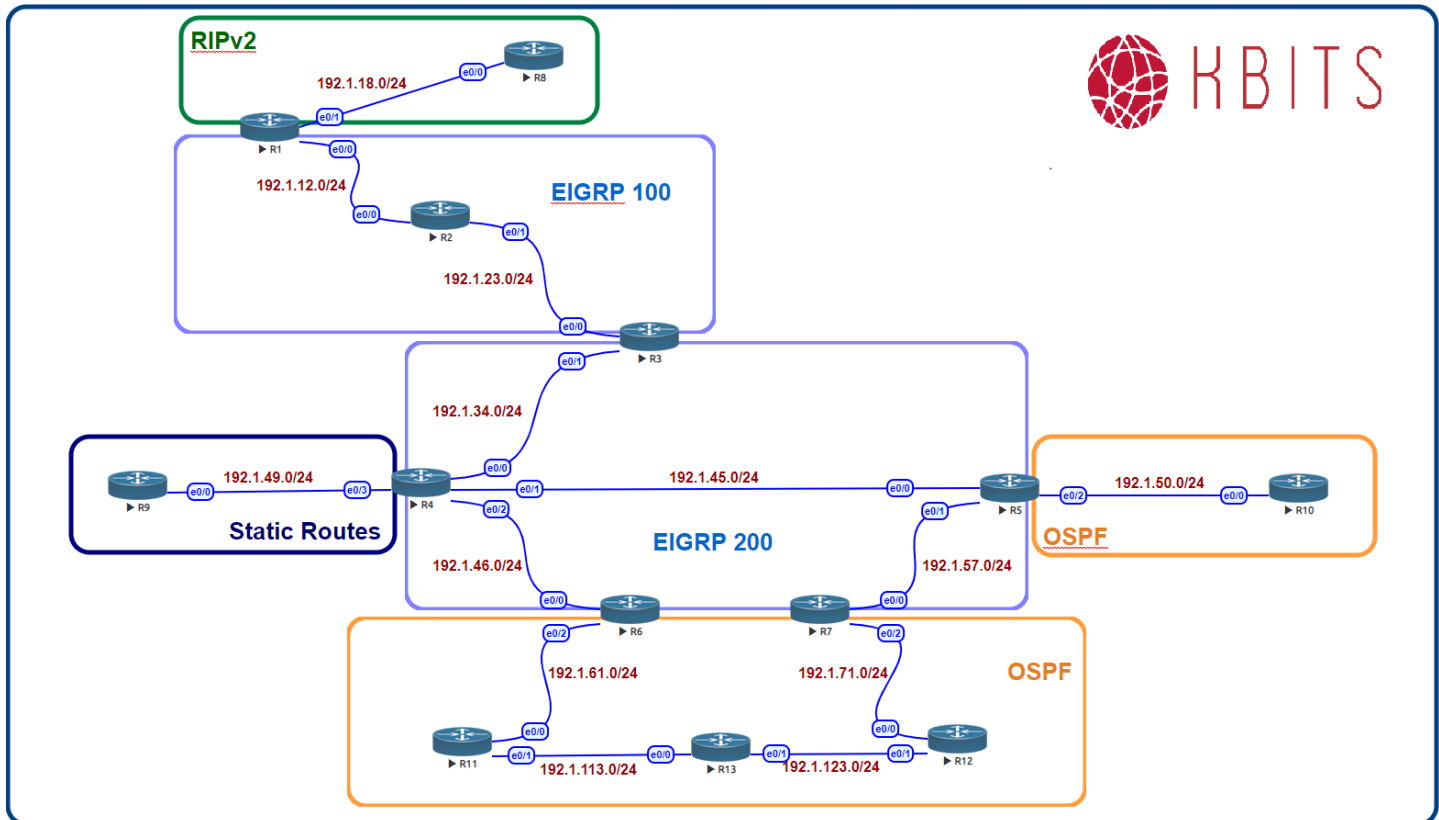
#### **R5**

```
Route-map O2E
  Set tag 456
!
Router eigrp 200
  Redistribute ospf 1 metric 10 10 10 10 10 route-map O2E
```

#### **R1**

```
Route-map E2R deny 10
  Match tag 456
Route-map E2R permit 20
!
Router rip
  Redistribute eigrp 100 route-map E2R
```

# Lab 22 – Multi-Point Redistribution with Route Tagging



## Task 1

EIGRP 200 & OSPF (Southbound) need to redundancy with Redistribution. Redistribution needs to be configured on R6 & R7. R6 should be the preferred router to connect the 2 domains. Make sure that the routes are blocked from coming back into the Source Domain.

### Route-Maps to Block EIGRP Routes coming back into OSPF

#### R6

```
Route-map E2O permit 20
Set tag 111
!
Route-map O2E deny 10
Match tag 222
```

#### R7

```
Route-map E2O permit 20
```

```
Set tag 222
!  
Route-map O2E deny 10  
Match tag 111
```

### **Route-Maps to Block OSPF Routes coming back into EIGRP**

#### **R6**

```
Route-map O2E permit 20  
Set tag 333  
!  
Route-map E2O deny 10  
Match tag 444
```

#### **R7**

```
Route-map E2O deny 10  
Match tag 333  
!  
Route-map O2E permit 20  
Set tag 444
```

### **Redistribute Commands**

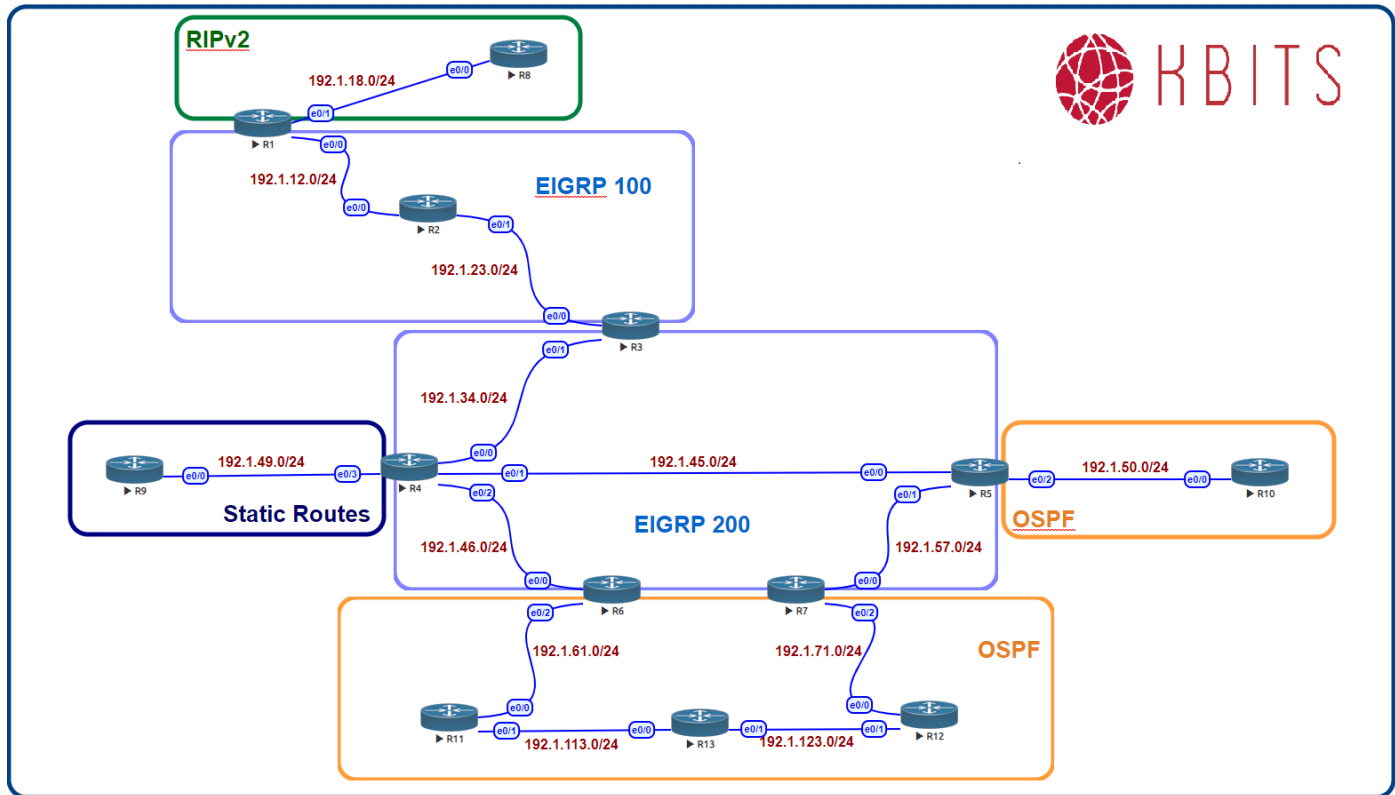
#### **R6**

```
Router eigrp 200  
Redistribute ospf 1 metric 1000 100 255 1 1500 route-map O2E  
!  
Router ospf 1  
Redistribute eigrp 200 metric 20 route-map E2O
```

#### **R7**

```
Router eigrp 200  
Redistribute ospf 1 metric 500 500 255 1 1500 route-map O2E  
!  
Router ospf 1  
Redistribute eigrp 200 metric 30 route-map E2O
```

# Lab 23 – Configuring BFD for EIGRP



## Task 1 – Configure BFD in AS 200

Configure routers in EIGRP in AS 200 such that all neighbor down events are detected in a Sub-second times to optimize reconvergence of a network. Use a BFD Hello Interval of 300 msec. The dead time should be 3 times the Hello Interval.

### R3

```
interface Ethernet0/1
  bfd interval 300 min_rx 300 multiplier 3
!
Router eigrp 200
  Bfd interface e 0/1
```

### R4

```
interface Ethernet0/0
  bfd interval 300 min_rx 300 multiplier 3
!
interface Ethernet0/1
```

```
bfd interval 300 min_rx 300 multiplier 3
!  
interface Ethernet0/2  
bfd interval 300 min_rx 300 multiplier 3
!  
Router eigrp 200  
Bfd all-interface
```

### **R5**

```
interface Ethernet0/0  
bfd interval 300 min_rx 300 multiplier 3
!  
interface Ethernet0/1  
bfd interval 300 min_rx 300 multiplier 3
!  
Router eigrp 200  
Bfd all-interface
```

### **R6**

```
interface Ethernet0/0  
bfd interval 300 min_rx 300 multiplier 3
!  
Router eigrp 200  
Bfd interface E0/0
```

### **R7**

```
interface Ethernet0/0  
bfd interval 300 min_rx 300 multiplier 3
!  
Router eigrp 200  
Bfd interface E0/0
```

# Configuring OSPF for IPv4 Networks

**Authored By:**

**Khawar Butt**

CCIE # 12353

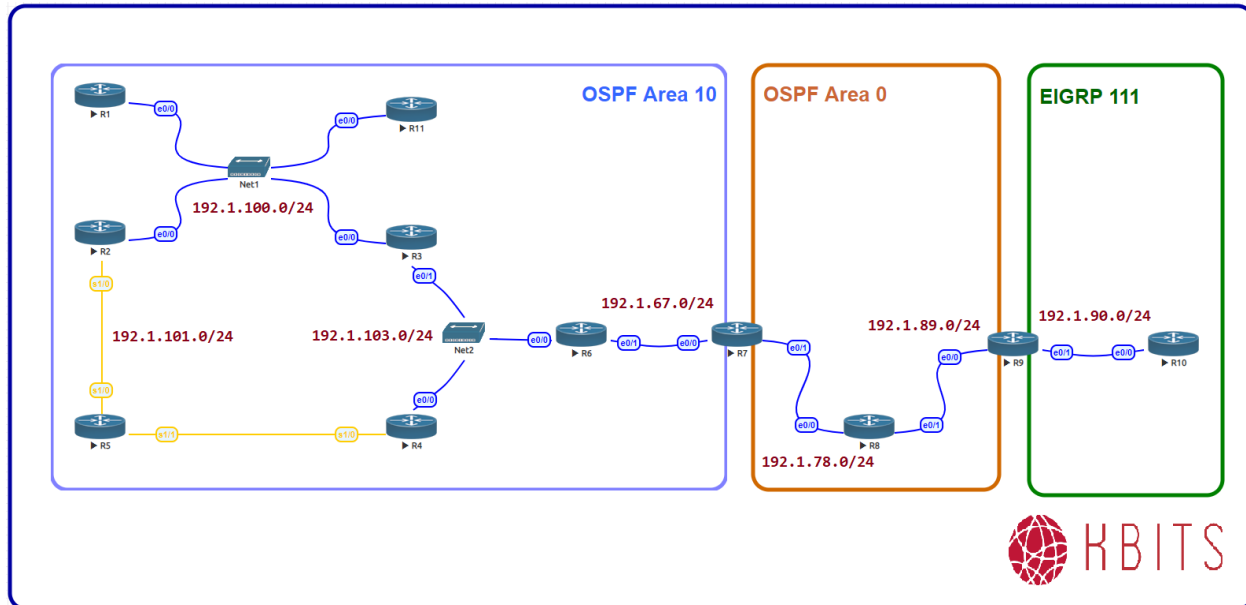
Hepta CCIE#12353

CCDE # 20110020

**Configuring OSPF**



# Lab 1 – Configure OSPF on Ethernet – Area 10



## Interface IP Address Configuration

### R1

Interface	IP Address	Subnet Mask
Loopback 0	1.1.1.1	255.0.0.0
E 0/0	192.1.100.1	255.255.255.0

### R2

Interface	IP Address	Subnet Mask
Loopback 0	2.2.2.2	255.0.0.0
E 0/0	192.1.100.2	255.255.255.0
S 1/0	192.1.101.2	255.255.255.0

### R3

Interface	IP Address	Subnet Mask
Loopback 0	3.3.3.3	255.0.0.0
E 0/0	192.1.100.3	255.255.255.0
E 0/1	192.1.103.3	255.255.255.0

### R4



Interface	IP Address	Subnet Mask
Loopback 0	4.4.4.4	255.0.0.0
E 0/0	192.1.103.4	255.255.255.0
S 1/0	192.1.102.4	255.255.255.0

## R5

Interface	IP Address	Subnet Mask
Loopback 0	5.5.5.5	255.0.0.0
S 1/0	192.1.101.5	255.255.255.0
S 1/1	192.1.102.5	255.255.255.0

## R6

Interface	IP Address	Subnet Mask
Loopback 0	6.6.6.6	255.0.0.0
E 0/0	192.1.103.6	255.255.255.0
E 0/1	192.1.67.6	255.255.255.0

## R7

Interface	IP Address	Subnet Mask
Loopback 0	7.7.7.7	255.0.0.0
E 0/0	192.1.67.7	255.255.255.0
E 0/1	192.1.78.7	255.255.255.0

## R8

Interface	IP Address	Subnet Mask
Loopback 0	8.8.8.8	255.0.0.0
E 0/0	192.1.78.8	255.255.255.0
E 0/1	192.1.89.8	255.255.255.0

## R9

Interface	IP Address	Subnet Mask
Loopback 0	9.9.9.9	255.0.0.0
E 0/0	192.1.89.9	255.255.255.0
E 0/1	192.1.90.9	255.255.255.0

## R10

Interface	IP Address	Subnet Mask
Loopback 0	10.10.10.10	255.0.0.0
E 0/0	192.1.90.10	255.255.255.0

## R11

Interface	IP Address	Subnet Mask
Loopback 0	11.11.11.11	255.0.0.0
E 0/0	192.1.100.11	255.255.255.0

### Task 1

Configure OSPF all the Broadcast Multi-Access (BMA) Ethernet network in Area 10. Enable OSPF on all loopbacks on all routers. Hard Code the Router-id based on the following:

- R1 – 0.0.0.1
- R2 – 0.0.0.2
- R3 – 0.0.0.3
- R4 – 0.0.0.4
- R6 – 0.0.0.6
- R7 – 0.0.0.7
- R11 – 0.0.0.11

<b>R1</b> Router OSPF 1 Router-id 0.0.0.1 Network 1.0.0.0 0.255.255.255 area 10 Network 192.1.100.0 0.0.0.255 area 10	<b>R2</b> Router OSPF 1 Router-id 0.0.0.2 Network 2.0.0.0 0.255.255.255 area 10 Network 192.1.100.0 0.0.0.255 area 10
<b>R3</b> Router OSPF 1 Router-id 0.0.0.3 Network 3.0.0.0 0.255.255.255 area 10 Network 192.1.100.0 0.0.0.255 area 10 Network 192.1.103.0 0.0.0.255 area 10	<b>R4</b> Router OSPF 1 Router-id 0.0.0.4 Network 4.0.0.0 0.255.255.255 area 10 Network 192.1.103.0 0.0.0.255 area 10
<b>R6</b> Router OSPF 1 Router-id 0.0.0.6 Network 6.0.0.0 0.255.255.255 area 10 Network 192.1.103.0 0.0.0.255 area 10 Network 192.1.67.0 0.0.0.255 area 10	<b>R7</b> Router OSPF 1 Router-id 0.0.0.7 Network 7.0.0.0 0.255.255.255 area 10 Network 192.1.67.0 0.0.0.255 area 10

### Task 2

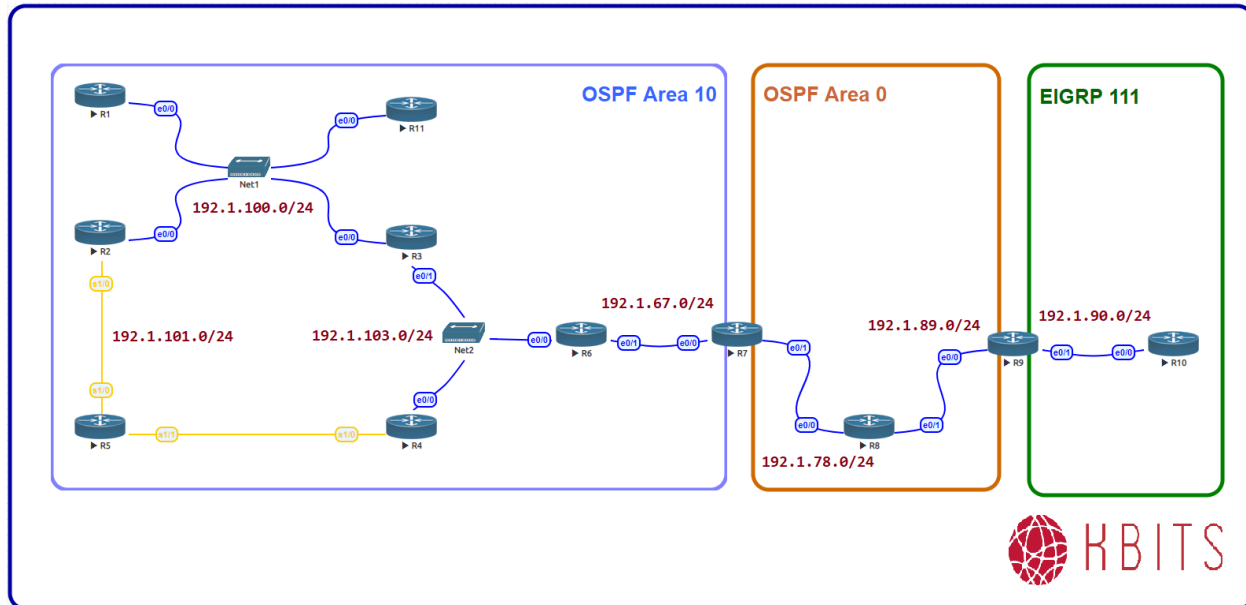
Configure the routers such that R1 becomes the DR and R2 as the BDR on the 192.1.100.0/24 Network. R3 should be the DR & R4 should be the BDR for the 192.1.103.0/24 network.

<b>R1</b>	<b>R2</b>
-----------	-----------

Interface E 0/0 Ip ospf priority 100	Interface E 0/0 Ip ospf priority 50
<b>R3</b>	<b>R4</b>
Interface E 0/1 Ip ospf priority 100	Interface E 0/0 Ip ospf priority 50

**Note:** Issue the **Clear ip ospf process** command to reset the OSPF process for the change to take effect.

# Lab 2 – Configuring OSPF on Serial Links – Area 10



## Task 1

Run OSPF as your Routing Protocol on the Serial Networks between R2, R4 & R5 in Area 10. Enable OSPF on the Loopback interface on R5. Configure the Router ID of R5 as 0.0.0.5.

### R2

```
router ospf 1
network 192.1.101.0 0.0.0.255 area 10
```

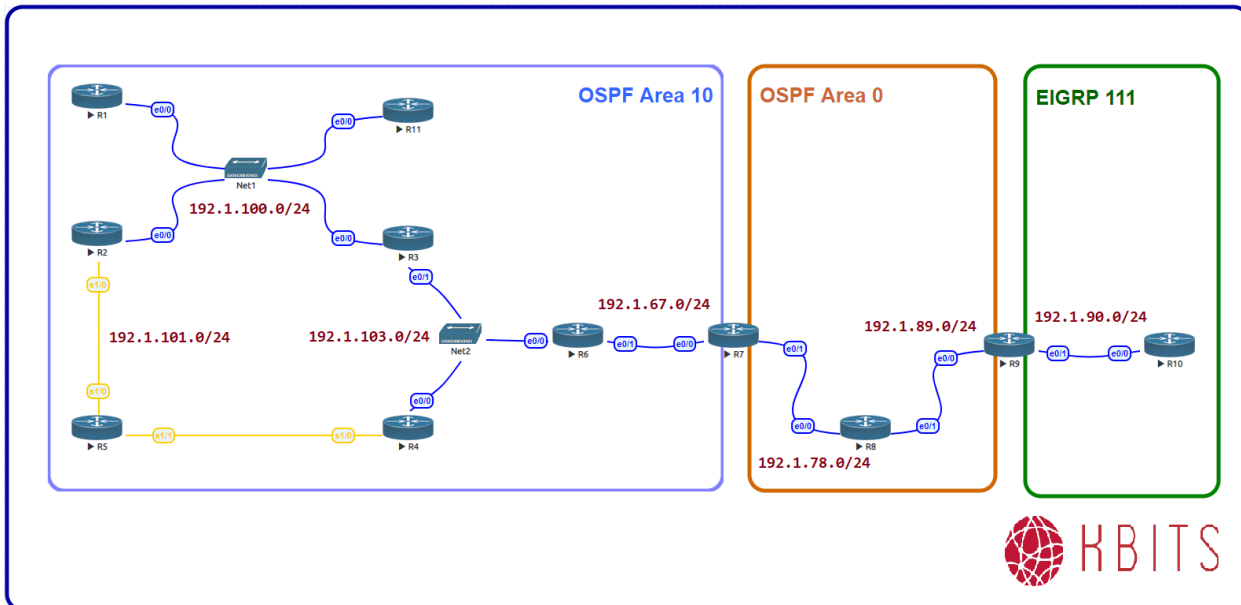
### R4

```
router ospf 1
network 192.1.102.0 0.0.0.255 area 10
```

### R5

```
router ospf 1
router-id 0.0.0.5
network 5.0.0.0 0.255.255.255 area 10
network 192.1.101.0 0.0.0.255 area 10
network 192.1.102.0 0.0.0.255 area 10
```

# Lab 3 – Configuring OSPF in Area 0



## Task 1

Configure R7, R8 & R9 in Area 0. Don't enable the Loopback Interface of R9 in OSPF. The Router ID's for R8 & R9 should be 0.0.0.8 & 0.0.0.9 respectively. Make sure that the neighbor relationships in Area 0 are established bypassing the DR & BDR election wait time.

### R7

```
router ospf 1
 network 192.1.78.0 0.0.0.255 area 0
!
Interface E 0/1
 Ip ospf network point-to-point
```

### R8

```
router ospf 1
 router-id 0.0.0.8
 network 8.0.0.0 0.255.255.255 area 0
 network 192.1.78.0 0.0.0.255 area 0
 network 192.1.89.0 0.0.0.255 area 0
!
Interface E 0/0
 Ip ospf network point-to-point
!
Interface E 0/1
```

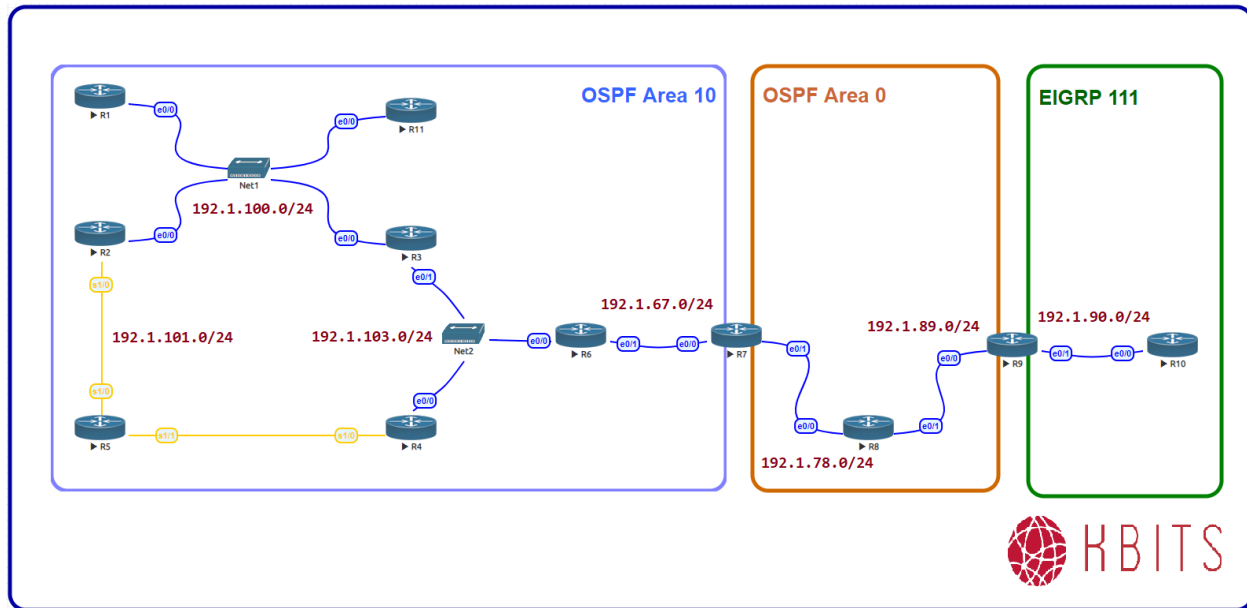
Ip ospf network point-to-point
<b>R9</b>
router ospf 1 router-id 0.0.0.9 network 192.1.89.0 0.0.0.255 area 0

**Task 2**

Make sure that all OSPF Loopbacks networks appear with the Interface mask. They should not appear as a Host Route.

<b>R1</b>	<b>R2</b>
Interface Loopback0 Ip ospf network point-to-point	Interface Loopback0 Ip ospf network point-to-point
<b>R3</b>	<b>R4</b>
Interface Loopback0 Ip ospf network point-to-point	Interface Loopback0 Ip ospf network point-to-point
<b>R5</b>	<b>R6</b>
Interface Loopback0 Ip ospf network point-to-point	Interface Loopback0 Ip ospf network point-to-point
<b>R7</b>	<b>R8</b>
Interface Loopback0 Ip ospf network point-to-point	Interface Loopback0 Ip ospf network point-to-point
<b>R9</b>	<b>R11</b>
Interface Loopback0 Ip ospf network point-to-point	Interface Loopback0 Ip ospf network point-to-point

# Lab 4 – Configuring Unicast-based OSPF



## Task 1

Configure Unicast-based OSPF between R6 & R7.

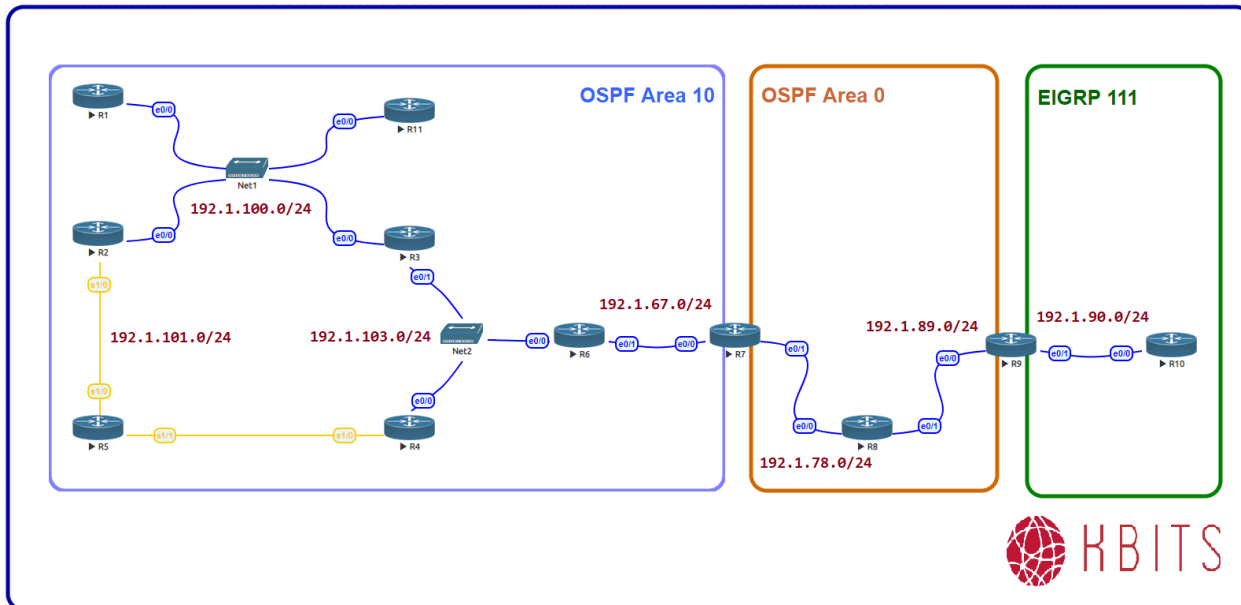
### R6

```
Interface E 0/1
Ip ospf network non-broadcast
!
Router ospf 1
Neighbor 192.1.67.7
```

### R7

```
Interface E 0/1
Ip ospf network non-broadcast
!
Router ospf 1
Neighbor 192.1.67.6
```

# Lab 5 – Configuring an OSPF ASBR



## Task 1

Configure EIGRP in AS 111 between R9 & R10. Enable all loopbacks on the 2 routers in EIGRP.

### R9

```
Router eigrp 111
Network 192.1.90.0
Network 9.0.0.0
```

### R10

```
Router eigrp 111
Network 192.1.90.0
Network 10.0.0.0
```

## Task 2

Configure Mutual Route Redistribution between OSPF & EIGRP on R9. Use Seed Metrics of your choice.

### R9

```
Router eigrp 111
Redistribute ospf 1 metric 10 10 10 10 10
!
Router ospf 1
```

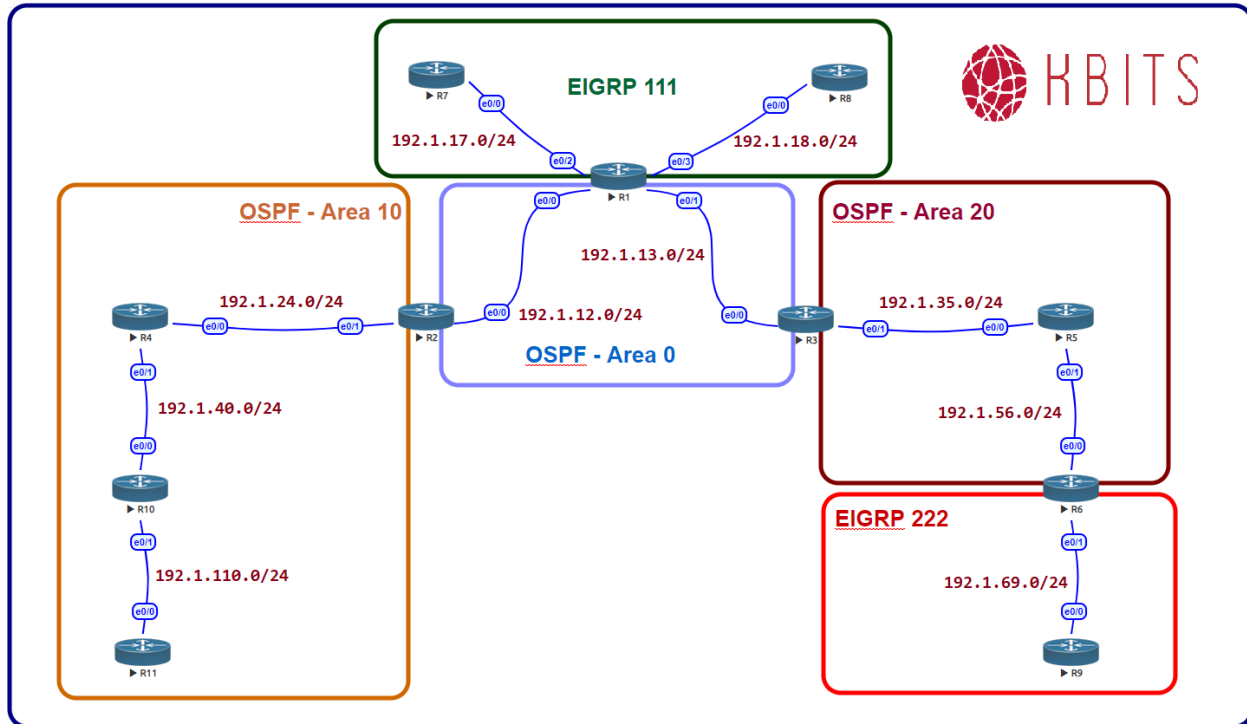


Redistribute eigrp 111 subnets

**Verification:**

Verify the OSPF Database for appropriate LSAs on the appropriate routers.

# Lab 6 – Configuring a Multi-Area / Multi-Domain Topology



## Interface IP Address Configuration

### R1

Interface	IP Address	Subnet Mask
Loopback 0	1.1.1.1	255.0.0.0
Loopback 1	11.11.11.11	255.0.0.0
E 0/0	192.1.12.1	255.255.255.0
E 0/1	192.1.13.1	255.255.255.0
E 0/2	192.1.17.1	255.255.255.0
E 0/3	192.1.18.1	255.255.255.0

### R2

Interface	IP Address	Subnet Mask
Loopback 0	2.2.2.2	255.0.0.0
Loopback 1	22.22.22.22	255.0.0.0
E 0/0	192.1.12.2	255.255.255.0
E 0/1	192.1.24.2	255.255.255.0

### R3

<b>Interface</b>	<b>IP Address</b>	<b>Subnet Mask</b>
Loopback 0	3.3.3.3	255.0.0.0
Loopback 1	33.33.33.33	255.0.0.0
E 0/0	192.1.13.3	255.255.255.0
E 0/1	192.1.35.3	255.255.255.0

### R4

<b>Interface</b>	<b>IP Address</b>	<b>Subnet Mask</b>
Loopback 0	4.4.4.4	255.0.0.0
Loopback 1	44.44.44.44	255.0.0.0
Loopback 11	105.1.4.1	255.255.255.0
Loopback 12	105.1.5.1	255.255.255.0
Loopback 13	105.1.6.1	255.255.255.0
Loopback 14	105.1.7.1	255.255.255.0
E 0/0	192.1.24.4	255.255.255.0
E 0/1	192.1.40.4	255.255.255.0

### R5

<b>Interface</b>	<b>IP Address</b>	<b>Subnet Mask</b>
Loopback 0	5.5.5.5	255.0.0.0
Loopback 1	55.55.55.55	255.0.0.0
E 0/0	192.1.35.5	255.255.255.0
E 0/1	192.1.56.5	255.255.255.0

### R6

<b>Interface</b>	<b>IP Address</b>	<b>Subnet Mask</b>
Loopback 0	6.6.6.6	255.0.0.0
Loopback 1	66.66.66.66	255.0.0.0
E 0/0	192.1.56.6	255.255.255.0
E 0/1	192.1.69.6	255.255.255.0

### R7

<b>Interface</b>	<b>IP Address</b>	<b>Subnet Mask</b>
Loopback 0	7.7.7.7	255.0.0.0
Loopback 1	107.7.72.1	255.255.255.0
Loopback 2	107.7.73.1	255.255.255.0
Loopback 3	107.7.74.1	255.255.255.0
Loopback 4	107.7.75.1	255.255.255.0

E 0/0	192.1.17.7	255.255.255.0
-------	------------	---------------

### R8

Interface	IP Address	Subnet Mask
Loopback 0	8.8.8.8	255.0.0.0
Loopback 1	88.88.88.88	255.0.0.0
E 0/0	192.1.18.8	255.255.255.0

### R9

Interface	IP Address	Subnet Mask
Loopback 0	9.9.9.9	255.0.0.0
Loopback 1	99.99.99.99	255.0.0.0
E 0/0	192.1.69.9	255.255.255.0

### R10

Interface	IP Address	Subnet Mask
Loopback 0	10.10.10.10	255.0.0.0
Loopback 11	100.1.1.1	255.0.0.0
Loopback 12	101.1.1.1	255.0.0.0
Loopback 13	102.1.1.1	255.0.0.0
Loopback 14	103.1.1.1	255.0.0.0
E 0/0	192.1.40.10	255.255.255.0
E 0/1	192.1.110.10	255.255.255.0

### R11

Interface	IP Address	Subnet Mask
Loopback 0	111.111.111.111	255.255.255.0
E 0/0	192.1.110.11	255.255.255.0

### Task 1

Configure OSPF in Area 0 between R1, R2 & R3. Besides the physical links, enable the Loopback 0 interfaces of all 3 routers in Area 0. Loopbacks should be advertised with the Interface Mask. Hard Code the Router-id based on the following:

- R1 – 0.0.0.1
- R2 – 0.0.0.2
- R3 – 0.0.0.3

<b>R1</b>	<b>R2</b>
Router OSPF 1	Router OSPF 1

<pre> Router-id 0.0.0.1 Network 1.0.0.0 0.255.255.255 area 0 Network 192.1.12.0 0.0.0.255 area 0 Network 192.1.13.0 0.0.0.255 area 0 ! Interface Loopback0 Ip ospf network point-to-point </pre>	<pre> Router-id 0.0.0.2 Network 2.0.0.0 0.255.255.255 area 0 Network 192.1.12.0 0.0.0.255 area 0 ! Interface Loopback0 Ip ospf network point-to-point </pre>
<p><b>R3</b></p> <pre> Router OSPF 1 Router-id 0.0.0.3 Network 3.0.0.0 0.255.255.255 area 0 Network 192.1.13.0 0.0.0.255 area 0 ! Interface Loopback0 Ip ospf network point-to-point </pre>	

### Task 2

Configure OSPF in Area 10 between R2, R4, R10 & R11. Besides the physical links, enable the Loopback 1 interface on R2 and all the loopbacks of the other 3 routers in Area 10. Loopbacks should be advertised with the Interface Mask. Hard Code the Router-id based on the following:

- R4 – 0.0.0.4
- R10 – 0.0.0.10
- R11 – 0.0.0.11

<p><b>R10</b></p> <pre> Router OSPF 1 Router-id 0.0.0.10 Network 0.0.0.0 255.255.255.255 10 ! Interface Loopback0 Ip ospf network point-to-point ! Interface Loopback11 Ip ospf network point-to-point ! Interface Loopback12 Ip ospf network point-to-point ! Interface Loopback13 Ip ospf network point-to-point ! </pre>	<p><b>R4</b></p> <pre> Router OSPF 1 Router-id 0.0.0.4 Network 4.0.0.0 0.255.255.255 area 10 Network 44.0.0.0 0.255.255.255 area 10 Network 105.0.0.0 0.255.255.255 area 10 Network 192.1.24.0 0.0.0.255 area 10 Network 192.1.40.0 0.0.0.255 area 10 ! Interface Loopback0 Ip ospf network point-to-point ! Interface Loopback1 Ip ospf network point-to-point ! Interface Loopback11 Ip ospf network point-to-point </pre>
---	--

<pre>Interface Loopback14 Ip ospf network point-to-point</pre>	<pre>! Interface Loopback12 Ip ospf network point-to-point ! Interface Loopback13 Ip ospf network point-to-point ! Interface Loopback14 Ip ospf network point-to-point</pre>
<p><b>R2</b></p> <pre>Router OSPF 1 Network 192.1.24.0 0.0.0.255 area 10 Network 22.0.0.0 0.255.255.255 area 10 ! Interface Loopback1 Ip ospf network point-to-point</pre>	<p><b>R11</b></p> <pre>Router OSPF 1 Router-id 0.0.0.11 Network 111.0.0.0 0.255.255.255 area 10 Network 192.1.110.0 0.0.0.255 area 10 ! Interface Loopback0 Ip ospf network point-to-point</pre>

### Task 3

Configure OSPF in Area 20 between R3, R5 & R6. Besides the physical links, enable the Loopback 0 interface on R3 & R6 and all the loopbacks on R5 in Area 20. Loopbacks should be advertised with the Interface Mask. Hard Code the Router-id based on the following:

R5 – 0.0.0.5

R6 – 0.0.0.6

<p><b>R3</b></p> <pre>Router OSPF 1 Network 192.1.35.0 0.0.0.255 area 20 Network 33.0.0.0 0.255.255.255 area 20 ! Interface Loopback1 Ip ospf network point-to-point</pre>	<p><b>R5</b></p> <pre>Router OSPF 1 Router-id 0.0.0.5 Network 5.0.0.0 0.255.255.255 area 20 Network 55.0.0.0 0.255.255.255 area 20 Network 192.1.35.0 0.0.0.255 area 20 Network 192.1.56.0 0.0.0.255 area 20 ! Interface Loopback0 Ip ospf network point-to-point ! Interface Loopback1 Ip ospf network point-to-point</pre>
<p><b>R6</b></p> <pre>Router OSPF 1 Router-id 0.0.0.6</pre>	

```
Network 6.0.0.0 0.255.255.255 area 20
Network 192.1.56.0 0.0.0.255 area 20
!
Interface Loopback0
Ip ospf network point-to-point
```

#### Task 4

Configure EIGRP is AS 111 between R1, R7 & R8. Enable all loopbacks on R7 & R8 in EIGRP 111. Enable Loopback 1 on R1 in EIGRP 111.

<b>R1</b> Router EIGRP 111 Network 192.1.17.0 Network 192.1.18.0 Network 11.0.0.0	<b>R7</b> Router EIGRP 111 Network 192.1.17.0 Network 7.0.0.0 Network 107.0.0.0
<b>R8</b> Router EIGRP 111 Network 192.1.18.0 Network 8.0.0.0 Network 88.0.0.0	

#### Task 5

Configure EIGRP is AS 222 between R6 & R9. Enable all loopbacks on R9 in EIGRP 222. Enable Loopback 1 on R6 in EIGRP 222.

<b>R6</b> Router EIGRP 222 Network 192.1.69.0 Network 66.0.0.0	<b>R9</b> Router EIGRP 222 Network 192.1.69.0 Network 9.0.0.0 Network 99.0.0.0
---	--

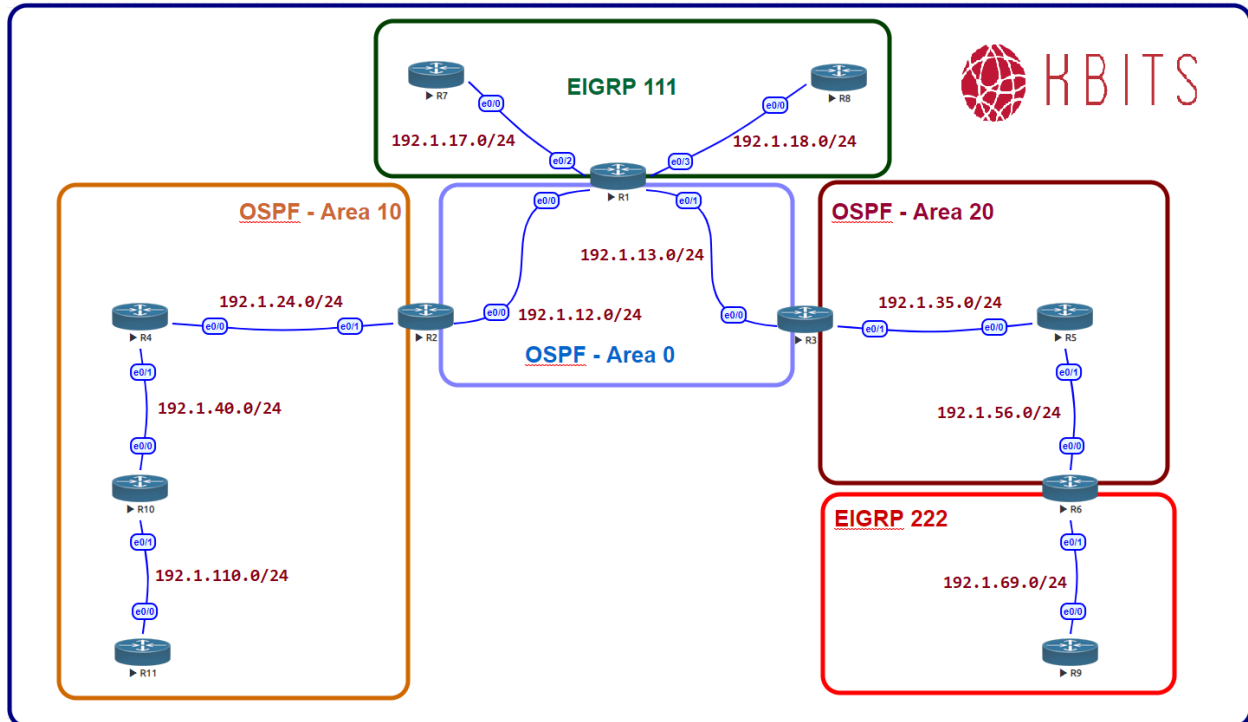
#### Task 6

Configure Mutual Redistribution between the appropriate routers to allow end-to-end connectivity between all routing domains. Use Seed metric of your choice.

<b>R1</b> Router ospf 1 Redistribute eigrp 111 subnets ! Router eigrp 111 Redistribute ospf 1 metric 10 10 10 10 10	<b>R6</b> Router ospf 1 Redistribute eigrp 222 subnets ! Router eigrp 222 Redistribute ospf 1 metric 10 10 10 10 10
--	--



# Lab 7 – Configuring Inter-Area Route Summarization



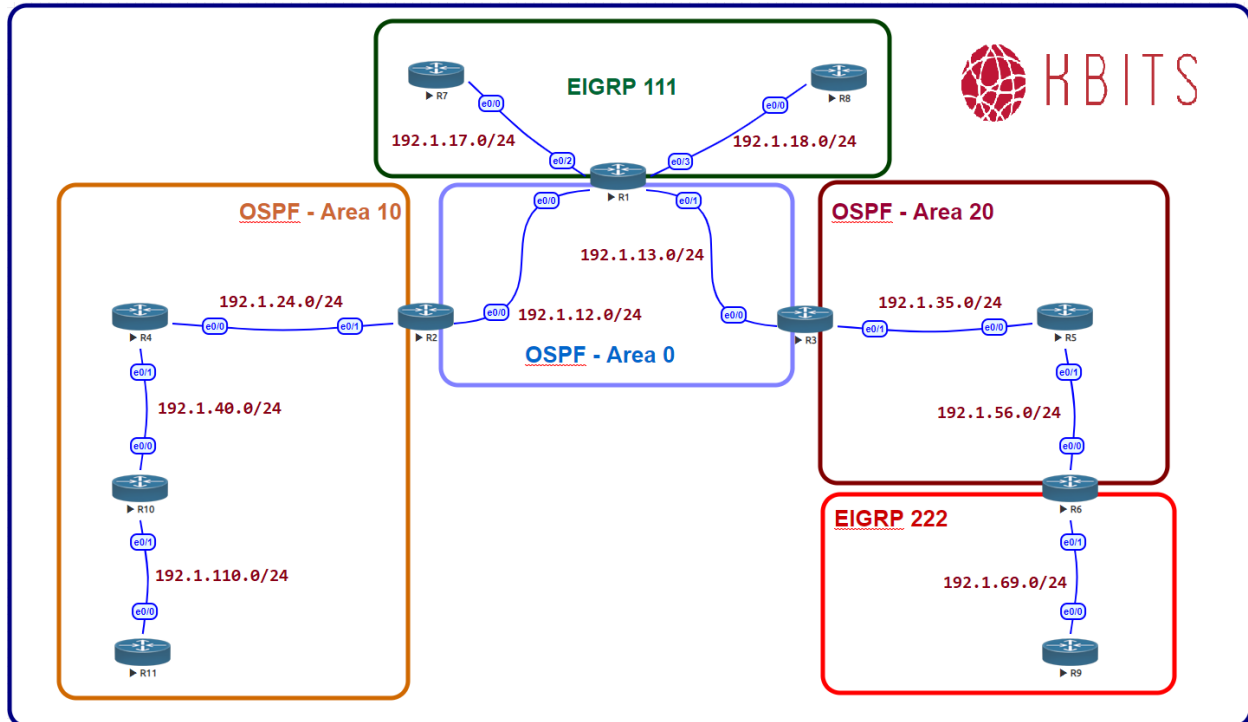
## Task 1

Configure Route Summarization on the appropriate ABR to summarize all the R11 Loopbacks.

## R2

```
Router ospf 1
Area 10 range 111.111.100.0 255.255.252.0
```

# Lab 8 – Configuring External Route Summarization



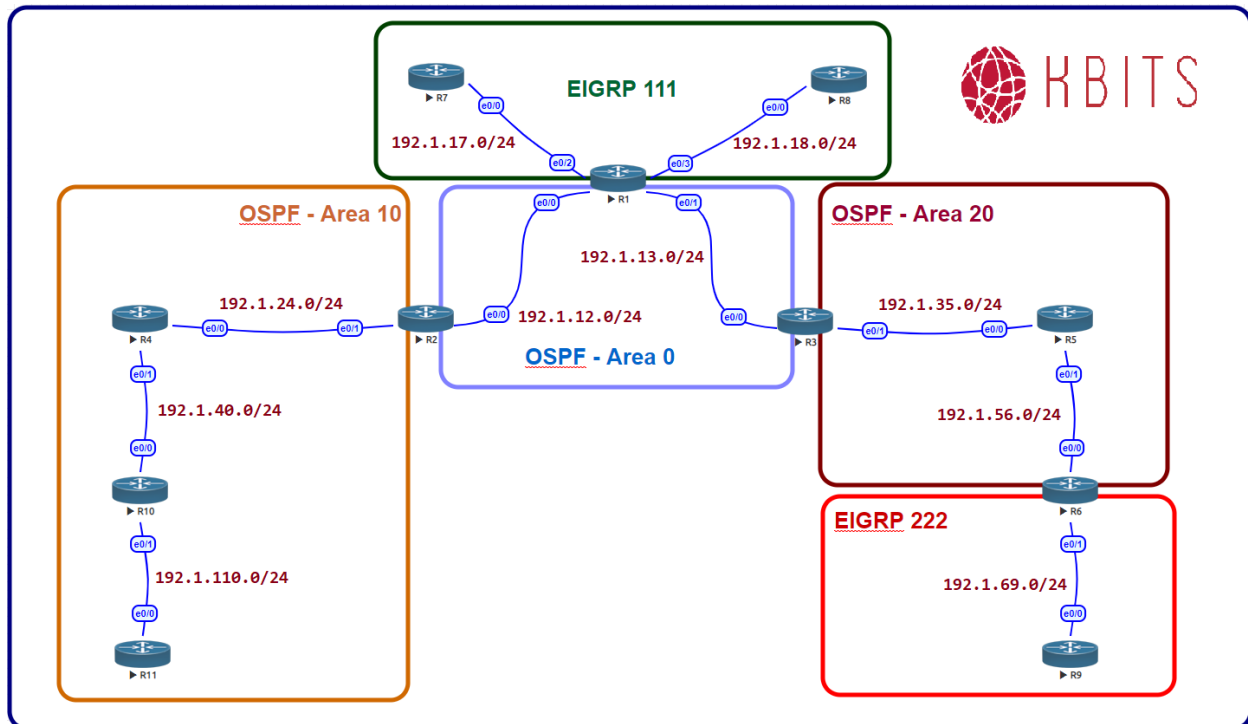
## Task 1

Configure Route Summarization on the appropriate ASBR to summarize all the routes from the 107.0.0.0/8 major network towards OSPF. Use the longest mask for Route Summarization.

### R1

```
Router ospf 1
Summary-address 107.7.72.0 255.255.252.0
```

# Lab 9 – Route Summarization and LSA Filtering



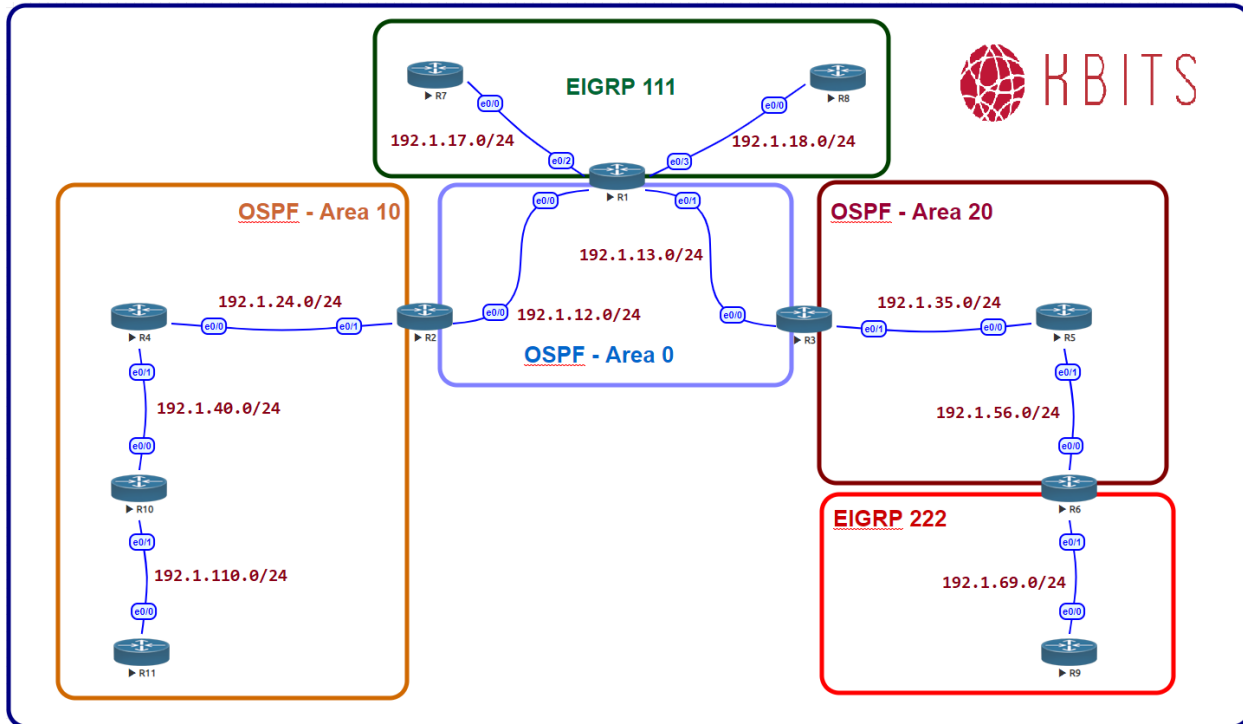
## Task 1

Configure LSA Filtering such that network 4.0.0.0/8 is not allowed to leave Area 10.

## R2

```
Ip prefix-list FILTER1 deny 4.0.0.0/8
Ip prefix-list FILTER1 permit 0.0.0.0/0 le 32
!
Router ospf 1
Area 10 filter-list prefix FILTER1 out
```

# Lab 10 – Configuring OSPF Authentication



## Task 1

Configure the most secure authentication on all routers in Area's 0. Use a key of 1 and a key-string **ccie123**.

### R1

```
interface E 0/0
ip ospf authentication message-digest
ip ospf message-digest-key 1 md5 ccie123
!
interface E 0/1
ip ospf authentication message-digest
ip ospf message-digest-key 1 md5 ccie123
```

### R2

```
interface E 0/0
ip ospf authentication message-digest
ip ospf message-digest-key 1 md5 ccie123
```

### R3

```
interface E 0/0
ip ospf authentication message-digest
ip ospf message-digest-key 1 md5 ccie123
```

## **Task 2**

Configure text authentication on all routers in 10. Use a key-string **cisco**.

### **R2**

```
interface E 0/1
ip ospf authentication
ip ospf authentication-key cisco
```

### **R4**

```
interface E 0/0
ip ospf authentication
ip ospf authentication-key cisco
!
interface E 0/1
ip ospf authentication
ip ospf authentication-key cisco
```

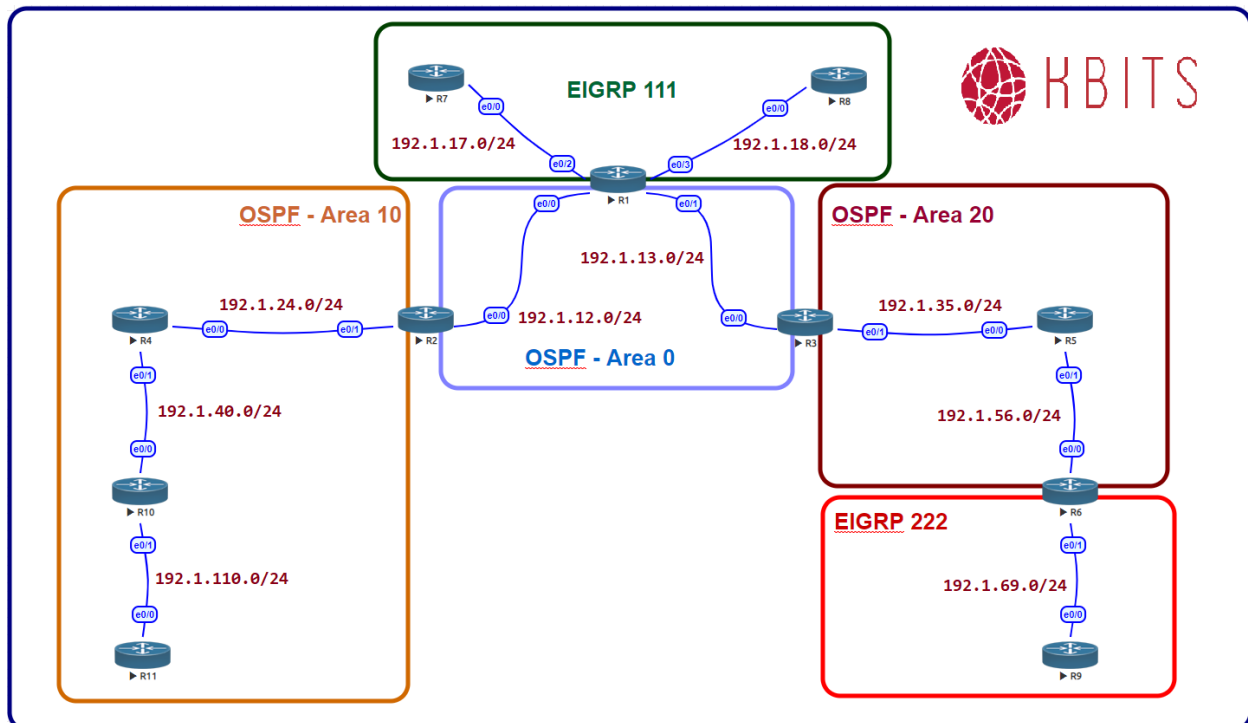
### **R10**

```
interface E 0/0
ip ospf authentication
ip ospf authentication-key cisco
!
interface E 0/1
ip ospf authentication
ip ospf authentication-key cisco
```

### **R11**

```
interface E 0/0
ip ospf authentication
ip ospf authentication-key cisco
```

# Lab 11 – Configuring OSPF Area Types



## Task 1

Configure Area 10 such that it does not receive any External Routes. It should maintain connectivity to the External Routes. **(Stub Area)**

<b>R2</b>	<b>R4</b>
Router ospf 1 Area 10 stub	Router ospf 1 Area 10 stub
<b>R10</b>	<b>R11</b>
Router ospf 1 Area 10 stub	Router ospf 1 Area 10 stub
<b>Note: The ABR will block the External Routes from EIGRP 111 &amp; EIGRP 222 from reaching Area 10 Internal Routers. R2 will inject a default route instead. This is a Stub Area. Verify it on R4, R10 &amp; R11 by checking the Routing table.</b>	

## Task 2

This step is a continuation of Task 1. Area 10 should also block Inter-Area routes maintaining reachability to them. **(Totally Stubby Area)**

**R2**

```
Router ospf 1
Area 10 stub no-summary
```

**Note: The ABR will block the Inter-Area Routes from getting propagating into Area 10. Instead R2 will inject a default route instead. This is a Totally Stubby Area. Verify it on R4, R10 & R11 by checking the Routing table.**

### Task 3

Configure Area 20 such that it does not receive any external routes from the backbone. The External routes from EIGRP 222 should continue to be received in Area 20 and propagated into the Backbone. **(NSSA Area)**

#### R3

```
Router ospf 1
Area 20 nssa
```

#### R5

```
Router ospf 1
Area 20 nssa
```

#### R6

```
Router ospf 1
Area 20 nssa
```

**Note: The ABR will block the External routes from the Backbone (EIGRP). Area 20 will continue to receive the external routes from EIGRP 222 as N routes. These routes will continue to be propagated towards the backbone. The ABR will convert the N routes into E routes as it propagates it into the Backbone. You will lose reachability to the External Routes from the Backbone as the ABR does not inject a default route in this configuration.**

### Task 4

This step is a continuation of Task 3. Configure Area 20 such that the previous requirement is maintained but Area 20 should also have reachability to the external routes from the backbone (EIGRP Routes). **(NSSA-Stub Area)**

#### R3

```
Router ospf 1
Area 20 nssa default-information-originate
```

**Note: This builds on the NSSA area by regaining reachability to the Backbone external routes. This is done by having the ABR injecting**

**the default route into Area 20.**

**Task 5**

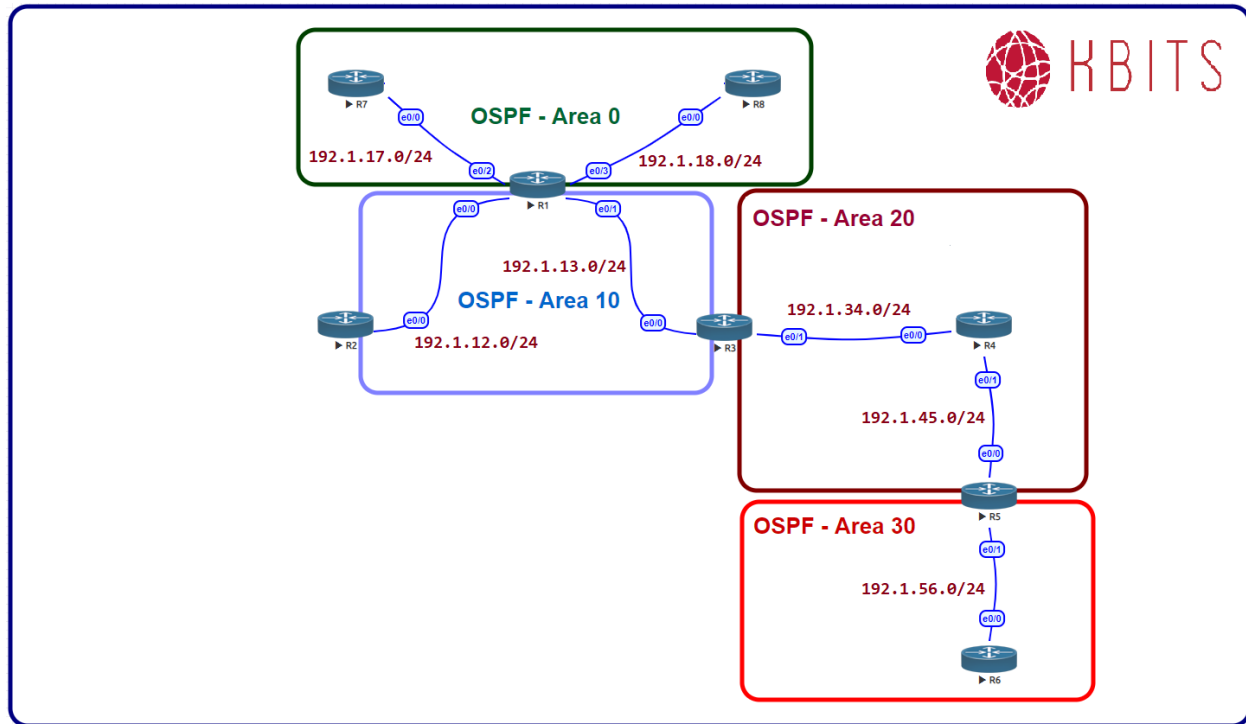
Configure Area 20 such that the Inter-Area routes are also blocked in addition to the external routes from the backbone. (**NSSA-Totally Stubby Area**)

**R3**

```
Router ospf 1
Area 20 nssa no-summary
```



# Lab 12 – Configuring Virtual Link



## Interface IP Address Configuration

### R1

Interface	IP Address	Subnet Mask
Loopback 0	1.1.1.1	255.0.0.0
Loopback 1	11.11.11.11	255.0.0.0
E 0/0	192.1.12.1	255.255.255.0
E 0/1	192.1.13.1	255.255.255.0
E 0/2	192.1.17.1	255.255.255.0
E 0/3	192.1.18.1	255.255.255.0

### R2

Interface	IP Address	Subnet Mask
Loopback 0	2.2.2.2	255.0.0.0
E 0/0	192.1.12.2	255.255.255.0

### R3

Interface	IP Address	Subnet Mask
Loopback 0	3.3.3.3	255.0.0.0
Loopback 1	33.33.33.33	255.0.0.0
E 0/0	192.1.13.3	255.255.255.0
E 0/1	192.1.34.3	255.255.255.0

### R4

Interface	IP Address	Subnet Mask
Loopback 0	4.4.4.4	255.0.0.0
E 0/0	192.1.34.4	255.255.255.0
E 0/1	192.1.45.4	255.255.255.0

### R5

Interface	IP Address	Subnet Mask
Loopback 0	5.5.5.5	255.0.0.0
Loopback 1	55.55.55.55	255.0.0.0
E 0/0	192.1.45.5	255.255.255.0
E 0/1	192.1.56.5	255.255.255.0

### R6

Interface	IP Address	Subnet Mask
Loopback 0	6.6.6.6	255.0.0.0
E 0/0	192.1.56.6	255.255.255.0

### R7

Interface	IP Address	Subnet Mask
Loopback 0	7.7.7.7	255.0.0.0
E 0/0	192.1.17.7	255.255.255.0

### R8

Interface	IP Address	Subnet Mask
Loopback 0	8.8.8.8	255.0.0.0
E 0/0	192.1.18.8	255.255.255.0

### Task 1

Configure OSPF in Area 0 between R1, R7 & R8. Besides the physical links, enable the Loopback 0 interfaces of all 3 routers in Area 0. Loopbacks should be advertised with the Interface Mask. Hard Code the Router-id based on the following:

R1 – 0.0.0.1  
 R7 – 0.0.0.7  
 R8 – 0.0.0.8

<p><b>R1</b></p> <pre>Router OSPF 1 Router-id 0.0.0.1 Network 1.0.0.0 0.255.255.255 area 0 Network 192.1.12.0 0.0.0.255 area 0 Network 192.1.13.0 0.0.0.255 area 0 ! Interface Loopback0 Ip ospf network point-to-point</pre>	<p><b>R7</b></p> <pre>Router OSPF 1 Router-id 0.0.0.7 Network 7.0.0.0 0.255.255.255 area 0 Network 192.1.17.0 0.0.0.255 area 0 ! Interface Loopback0 Ip ospf network point-to-point</pre>
<p><b>R8</b></p> <pre>Router OSPF 1 Router-id 0.0.0.8 Network 8.0.0.0 0.255.255.255 area 0 Network 192.1.18.0 0.0.0.255 area 0 ! Interface Loopback0 Ip ospf network point-to-point</pre>	

**Task 2**

Configure OSPF in Area 10 between R1, R2 & R3. Besides the physical links, enable the Loopback 0 interfaces of R2 & R3 in Area 10. R1 Loopback1 should also be enabled in area 10. Loopbacks should be advertised with the Interface Mask. Hard Code the Router-id based on the following:

R2 – 0.0.0.2  
 R3 – 0.0.0.3

<p><b>R1</b></p> <pre>Router OSPF 1 Network 192.1.12.0 0.0.0.255 area 10 Network 192.1.13.0 0.0.0.255 area 10 Network 11.0.0.0 0.255.255.255 area 10 ! Interface Loopback1 Ip ospf network point-to-point</pre>	<p><b>R2</b></p> <pre>Router OSPF 1 Router-id 0.0.0.2 Network 2.0.0.0 0.255.255.255 area 10 Network 192.1.12.0 0.0.0.255 area 10 ! Interface Loopback0 Ip ospf network point-to-point</pre>
<p><b>R3</b></p> <pre>Router OSPF 1</pre>	

```

Router-id 0.0.0.3
Network 3.0.0.0 0.255.255.255 area 10
Network 192.1.13.0 0.0.0.255 area 10
!
Interface Loopback0
Ip ospf network point-to-point

```

### Task 3

Configure OSPF in Area 20 between R3, R4 & R5. Besides the physical links, enable the Loopback 0 interfaces of R4 & R5 in Area 20. R3 Loopback1 should also be enabled in area 20. Loopbacks should be advertised with the Interface Mask. Hard Code the Router-id based on the following:

R4 – 0.0.0.4  
R5 – 0.0.0.5

#### R3

```

Router OSPF 1
Network 192.1.34.0 0.0.0.255 area 20
Network 33.0.0.0 0.255.255.255 area 20
!
Interface Loopback1
Ip ospf network point-to-point

```

#### R4

```

Router OSPF 1
Router-id 0.0.0.4
Network 4.0.0.0 0.255.255.255 area 20
Network 192.1.34.0 0.0.0.255 area 20
Network 192.1.45.0 0.0.0.255 area 20
!
Interface Loopback0
Ip ospf network point-to-point

```

#### R5

```

Router OSPF 1
Router-id 0.0.0.5
Network 5.0.0.0 0.255.255.255 area 20
Network 192.1.45.0 0.0.0.255 area 20
!
Interface Loopback0
Ip ospf network point-to-point

```

#### Task 4

Configure a Virtual Link between the appropriate devices to allow Area 20 to communicate to the rest of the network.

##### R1

```
router ospf 1
 area 10 virtual-link 0.0.0.3
```

##### R3

```
router ospf 1
 area 10 virtual-link 0.0.0.1
```

#### Task 5

Configure OSPF in Area 30 between R5 & R6. Besides the physical links, enable the Loopback 0 interfaces of R6 in Area 30. R5 Loopback1 should also be enabled in area 30. Loopbacks should be advertised with the Interface Mask. Hard Code the Router-id based on the following:

R6 – 0.0.0.6

##### R5

```
Router OSPF 1
 Network 192.1.56.0 0.0.0.255 area 30
 Network 55.0.0.0 0.255.255.255 area 30
 !
 Interface Loopback1
 Ip ospf network point-to-point
```

##### R6

```
Router OSPF 1
 Router-id 0.0.0.6
 Network 6.0.0.0 0.255.255.255 area 30
 Network 192.1.56.0 0.0.0.255 area 30
 !
 Interface Loopback0
 Ip ospf network point-to-point
```

#### Task 6

Configure a Virtual Link between the appropriate devices to allow Area 30 to communicate to the rest of the network.

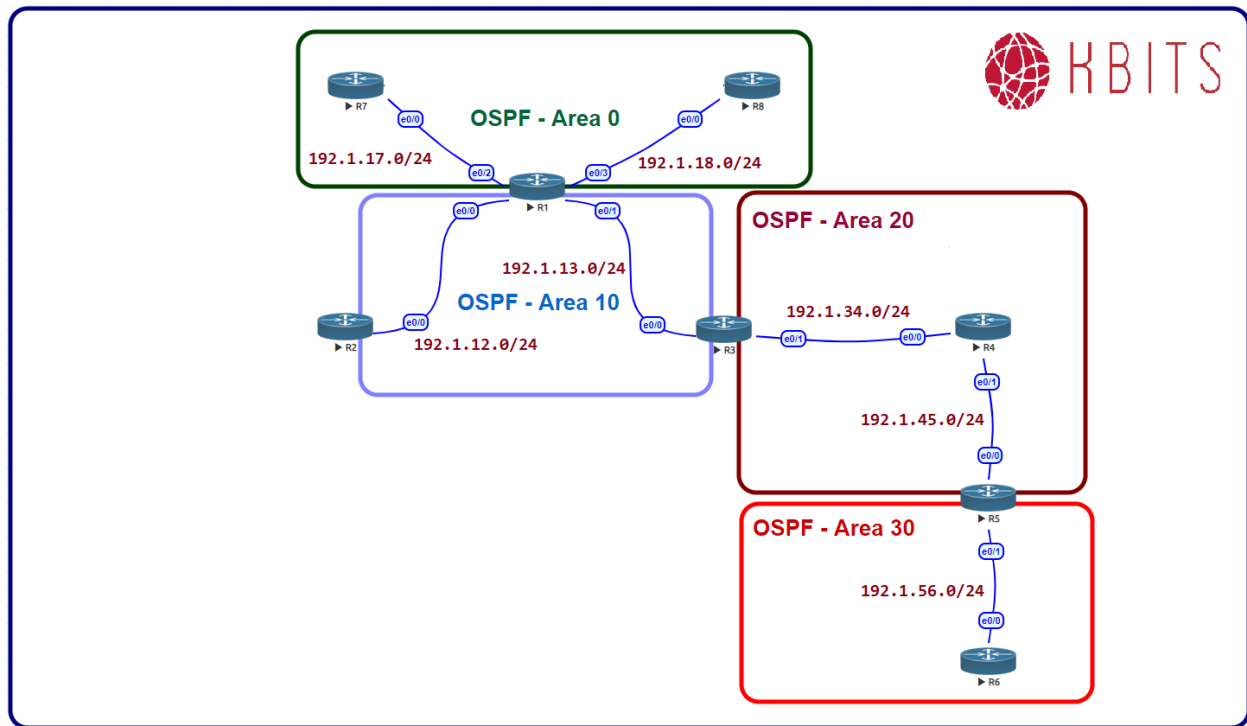
##### R3

```
router ospf 1
 area 20 virtual-link 0.0.0.5
```

##### R5

```
router ospf 1
 area 20 virtual-link 0.0.0.3
```

# Lab 13 – Configuring BFD for OSPF

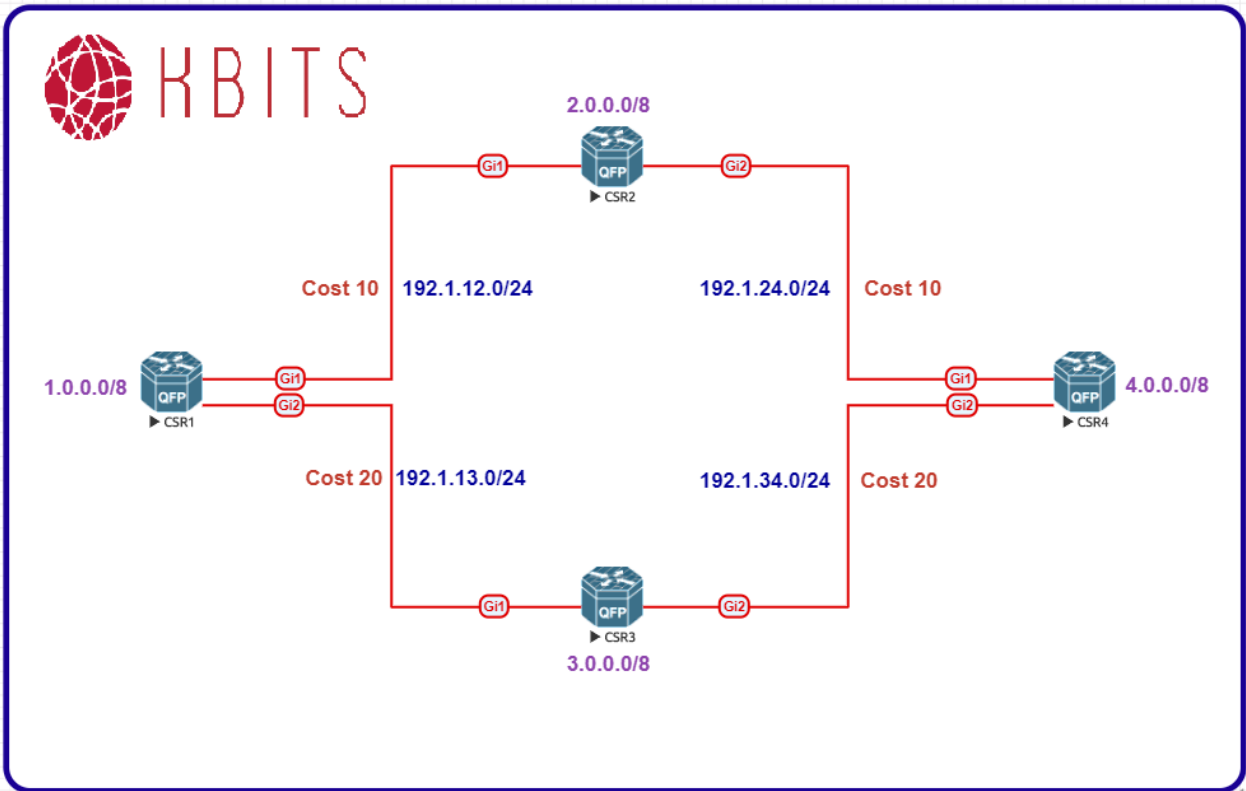


## Task 1

Configure BFD between all routers in area 0. Configure the BFD Interface interval to be 300 for sending and receiving. A neighbor should be deemed dead is the router misses 3 hellos.

<p><b>R1</b></p> <pre>Interface E 0/2 bfd interval 300 min_rx 300 multiplier 3 ! Router ospf 1 bfd all-interfaces</pre>	<p><b>R7</b></p> <pre>Interface E 0/0 bfd interval 300 min_rx 300 multiplier 3 ! Router ospf 1 bfd all-interfaces</pre>
<p><b>R8</b></p> <pre>Interface E 0/0 bfd interval 300 min_rx 300 multiplier 3 ! Router ospf 1 bfd all-interfaces</pre>	

# Lab 14 – Configuring IP FRR - OSPF



## Interface IP Address Configuration

### R1

Interface	IP Address	Subnet Mask
Loopback 0	1.1.1.1	255.0.0.0
G 1	192.1.12.1	255.255.255.0
G 2	192.1.13.1	255.255.255.0

### R2

Interface	IP Address	Subnet Mask
Loopback 0	2.2.2.2	255.0.0.0
G 1	192.1.12.2	255.255.255.0
G 2	192.1.24.2	255.255.255.0

### R3

Interface	IP Address	Subnet Mask
Loopback 0	3.3.3.3	255.0.0.0
G 1	192.1.13.3	255.255.255.0
G 2	192.1.34.3	255.255.255.0

### R4

Interface	IP Address	Subnet Mask
Loopback 0	4.4.4.4	255.0.0.0
G 1	192.1.24.4	255.255.255.0
G 2	192.1.34.4	255.255.255.0

### Task 1

Configure OSPF in Area 0 between R1, R2, R3 & R4. Besides the physical links, enable the Loopback 0 interfaces of all 4 routers in Area 0.

Loopbacks should be advertised with the Interface Mask. Hard Code the Router-id based on the following:

- R1 – 0.0.0.1
- R2 – 0.0.0.2
- R3 – 0.0.0.3
- R4 – 0.0.0.4

<b>R1</b>  Router OSPF 1 Router-id 0.0.0.1 Network 1.0.0.0 0.255.255.255 area 0 Network 192.1.12.0 0.0.0.255 area 0 Network 192.1.13.0 0.0.0.255 area 0 ! Interface Loopback0 Ip ospf network point-to-point	<b>R2</b>  Router OSPF 1 Router-id 0.0.0.2 Network 2.0.0.0 0.255.255.255 area 0 Network 192.1.12.0 0.0.0.255 area 0 Network 192.1.24.0 0.0.0.255 area 0 ! Interface Loopback0 Ip ospf network point-to-point
<b>R3</b>  Router OSPF 1 Router-id 0.0.0.3 Network 3.0.0.0 0.255.255.255 area 0 Network 192.1.13.0 0.0.0.255 area 0 Network 192.1.34.0 0.0.0.255 area 0 ! Interface Loopback0 Ip ospf network point-to-point	<b>R4</b>  Router OSPF 1 Router-id 0.0.0.4 Network 4.0.0.0 0.255.255.255 area 0 Network 192.1.24.0 0.0.0.255 area 0 Network 192.1.34.0 0.0.0.255 area 0 ! Interface Loopback0 Ip ospf network point-to-point



## Task 2

Configure the link cost based on the Diagram.

<b>R1</b>  Interface Gig1 Ip ospf cost 10 ! Interface Gig2 Ip ospf cost 20	<b>R2</b>  Interface Gig1 Ip ospf cost 10 ! Interface Gig2 Ip ospf cost 10
<b>R3</b>  Interface Gig1 Ip ospf cost 20 ! Interface Gig2 Ip ospf cost 20	<b>R4</b>  Interface Gig1 Ip ospf cost 10 ! Interface Gig2 Ip ospf cost 20

## Task 3

Verify the routing table and CEF on R1 for Network 4.0.0.0/8. It should have a single path via R2 (Lower cost)

<b>R1</b>  Show IP route 4.0.0.0  <b>Note:</b> It should have a single path via 192.1.12.2  Show ip cef 4.0.0.0  <b>Note:</b> It should have a single path via 192.1.12.2
---

#### Task 4

Enable Fast-reroute on all routers in area 0. Configure the Priority as low that creates the backup route for all networks in the OSPF Database.

##### R1

```
Router ospf 1
fast-reroute per-prefix enable area 0 prefix-priority low
```

##### R2

```
Router ospf 1
fast-reroute per-prefix enable area 0 prefix-priority low
```

##### R3

```
Router ospf 1
fast-reroute per-prefix enable area 0 prefix-priority low
```

##### R4

```
Router ospf 1
fast-reroute per-prefix enable area 0 prefix-priority low
```

#### Task 5

Verify the routing table and CEF on R1 for Network 4.0.0.0/8. It should have a repair path via R3 (higher cost) installed and ready in case the lower cost route goes down.

##### R1

```
Show IP route 4.0.0.0
```

**Note:** It should have a repair path via 192.1.13.3

```
Show ip cef 4.0.0.0
```

**Note:** It should have a repair path via 192.1.13.3

# Configuring BGP

**Authored By:**

**Khawar Butt**

CCIE # 12353

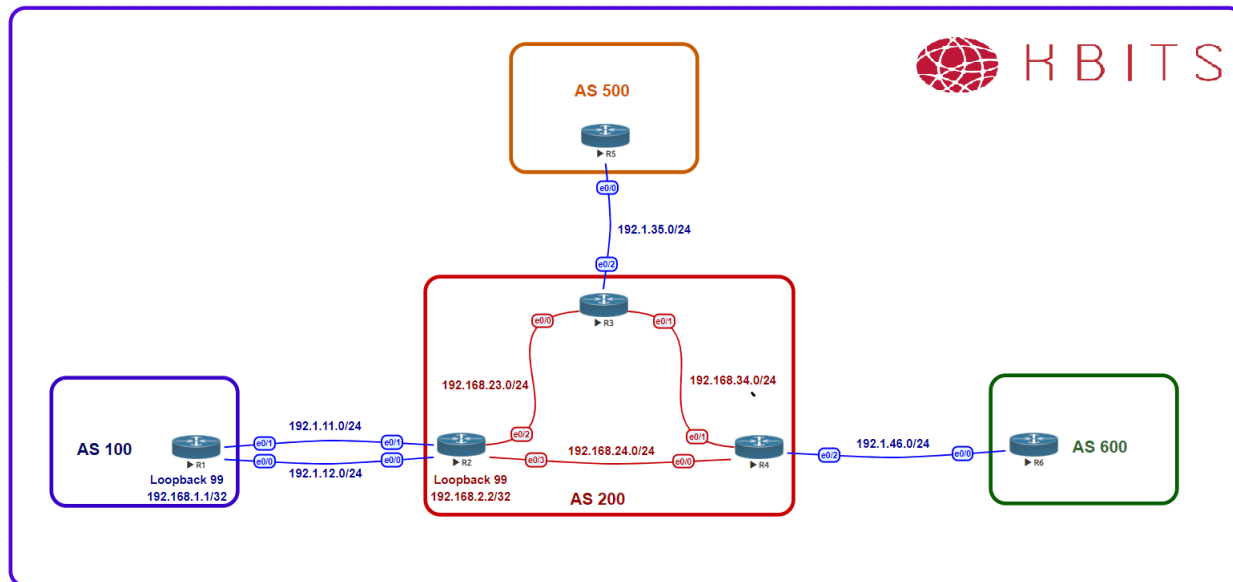
Hepta CCIE#12353

CCDE # 20110020

**Configuring BGP**



# Lab 1 - Configuring eBGP



## Interface IP Address Configuration

### R1

Interface	IP Address	Subnet Mask
Loopback 0	1.1.1.1	255.0.0.0
Loopback 1	11.1.1.1	255.255.255.0
Loopback 99	192.168.1.1	255.255.255.255
E 0/0	192.1.12.1	255.255.255.0
E 0/1	192.1.11.1	255.255.255.0

### R2

Interface	IP Address	Subnet Mask
Loopback 0	2.2.2.2	255.0.0.0
Loopback 1	22.2.2.2	255.255.255.0
Loopback 10	10.2.2.2	255.255.255.255
Loopback 99	192.168.2.2	255.255.255.255
E 0/0	192.1.12.2	255.255.255.0
E 0/1	192.1.11.2	255.255.255.0
E 0/2	192.168.23.2	255.255.255.0
E 0/3	192.168.24.2	255.255.255.0

**R3**

<b>Interface</b>	<b>IP Address</b>	<b>Subnet Mask</b>
Loopback 0	3.3.3.3	255.0.0.0
Loopback 1	33.3.3.3	255.255.255.0
Loopback 10	10.3.3.3	255.255.255.255
E 0/0	192.168.23.3	255.255.255.0
E 0/1	192.168.34.3	255.255.255.0
E 0/2	192.1.35.3	255.255.255.0

**R4**

<b>Interface</b>	<b>IP Address</b>	<b>Subnet Mask</b>
Loopback 0	4.4.4.4	255.0.0.0
Loopback 1	44.4.4.4	255.255.255.0
Loopback 10	10.4.4.4	255.255.255.255
E 0/0	192.168.24.4	255.255.255.0
E 0/1	192.168.34.4	255.255.255.0
E 0/2	192.1.46.4	255.255.255.0

**R5**

<b>Interface</b>	<b>IP Address</b>	<b>Subnet Mask</b>
Loopback 0	5.5.5.5	255.0.0.0
Loopback 1	55.5.5.5	255.255.255.0
E 0/0	192.1.35.5	255.255.255.0

**R6**

<b>Interface</b>	<b>IP Address</b>	<b>Subnet Mask</b>
Loopback 0	6.6.6.6	255.0.0.0
Loopback 1	66.6.6.6	255.255.255.0
E 0/0	192.1.46.6	255.255.255.0

### Task 1

Configure a BGP neighbor relationship between R3 and R5. R3 should be in AS 200 and R5 should be in AS 500. Advertise the loopback networks in BGP. Hard-code the Router ID for the BGP routers as 33.33.33.33 for R3 and 55.55.55.55 for R5.

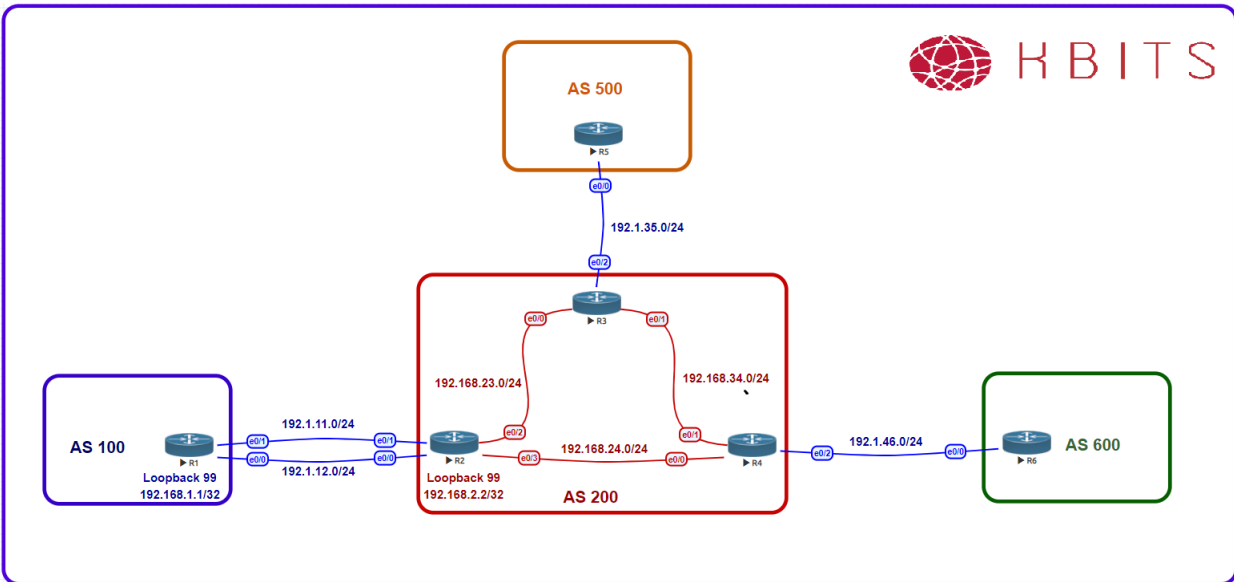
<b>R3</b>	<b>R5</b>
Router BGP 200 bgp router-id 33.33.33.33 Network 3.0.0.0 Network 33.3.3.0 mask 255.255.255.0 Neighbor 192.1.35.5 remote-as 500	Router BGP 500 bgp router-id 55.55.55.55 Network 5.0.0.0 Network 55.5.5.0 mask 255.255.255.0 Neighbor 192.1.35.3 remote-as 200

### Task 2

Configure a BGP neighbor relationship between R4 and R6. R4 should be in AS 200 and R6 should be in AS 600. Advertise the loopback networks in BGP. Hard-code the Router ID for the BGP routers as 44.44.44.44 for R4 and 66.66.66.66 for R6.

<b>R4</b>	<b>R6</b>
Router BGP 200 bgp router-id 44.44.44.44 Network 4.0.0.0 Network 44.4.4.0 mask 255.255.255.0 Neighbor 192.1.46.6 remote-as 600	Router BGP 600 bgp router-id 66.66.66.66 Network 6.0.0.0 Network 66.6.6.0 mask 255.255.255.0 Neighbor 192.1.46.4 remote-as 200

# Lab 2 – Configuring eBGP Multi-Hop



## Task 1

Configure a Static route on R1 & R2 to reach each others Loopback 99 via the 2 directed connected links.

### R1

```
Ip route 192.168.2.2 255.255.255.255 192.1.11.2  
Ip route 192.168.2.2 255.255.255.255 192.1.12.2
```

### R2

```
Ip route 192.168.1.1 255.255.255.255 192.1.11.1  
Ip route 192.168.1.1 255.255.255.255 192.1.12.1
```

## Task 2

Configure a BGP neighbor relationship between R1 & R2 in AS 100 & AS 200 respectively. Use Loopback99 address for the peering.

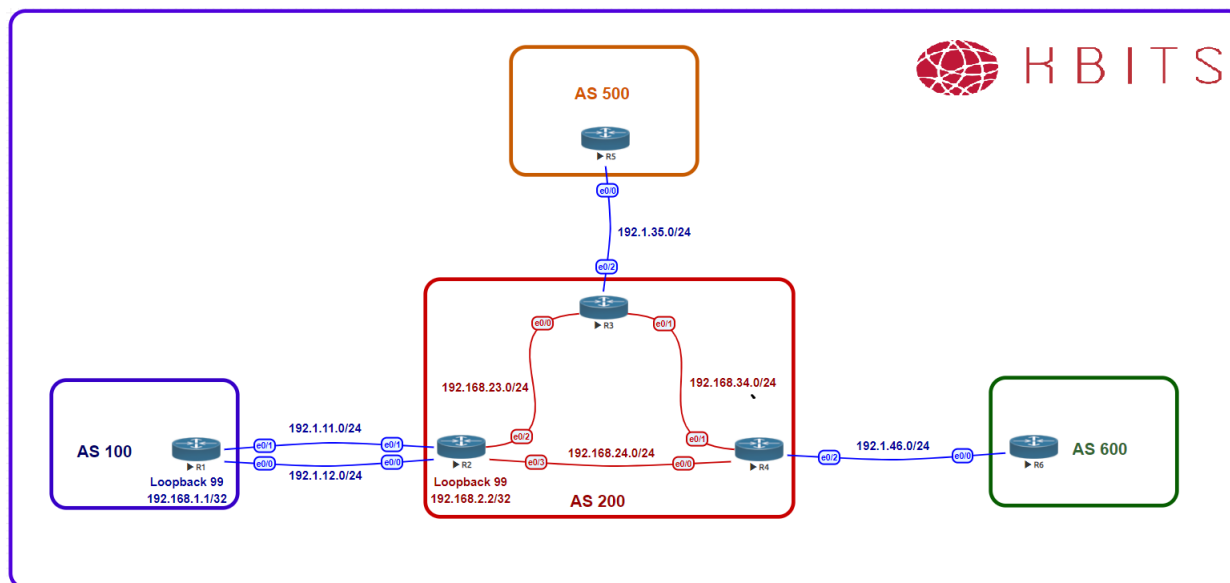
### R1

```
Router BGP 100  
Neighbor 192.168.2.2 remote-as 200  
Neighbor 192.168.2.2 ebgp-multihop
```

### R2

```
Router BGP 200  
Neighbor 192.168.1.1 remote-as 100  
Neighbor 192.168.1.1 ebgp-multihop
```

# Lab 3 – Redistributing Networks into BGP



## Task 1

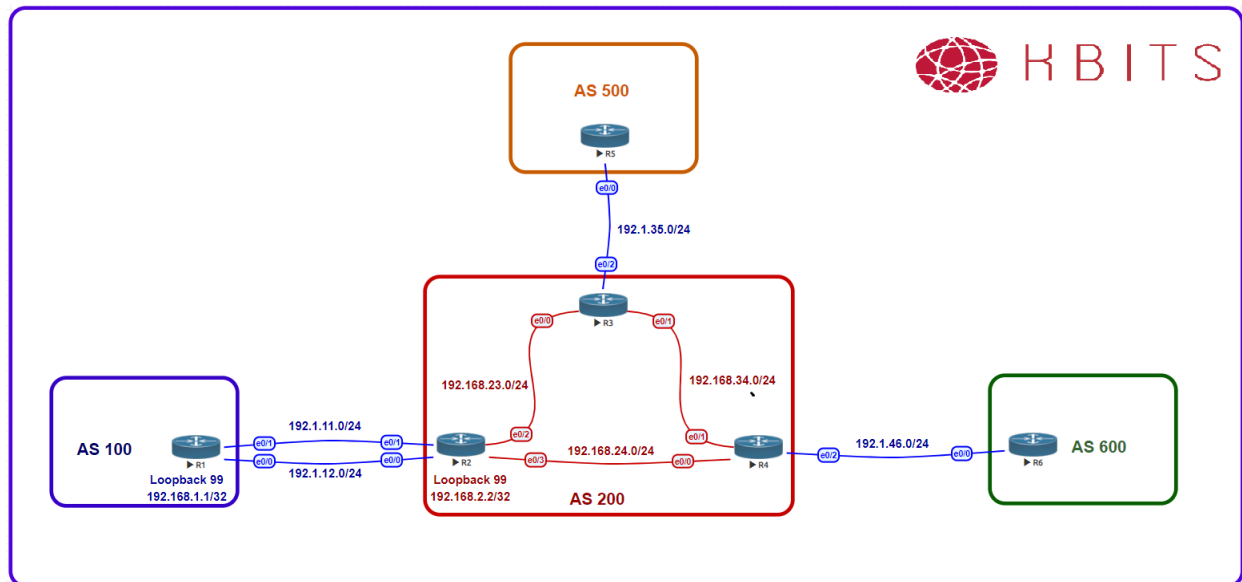
Inject Loopback0 & Loopback1 networks on R1 into BGP. Make sure that the routes appear with an origin code of “i” in the BGP table.

### R1

```
Ip prefix-list RC permit 1.0.0.0/8
Ip prefix-list RC permit 11.1.1.0/24
!
Route-map RC
Match ip address prefix RC
Set origin igp
!
Router bgp 100
Redistribute connected route-map RC
```



# Lab 4 – Configuring BGP Authentication

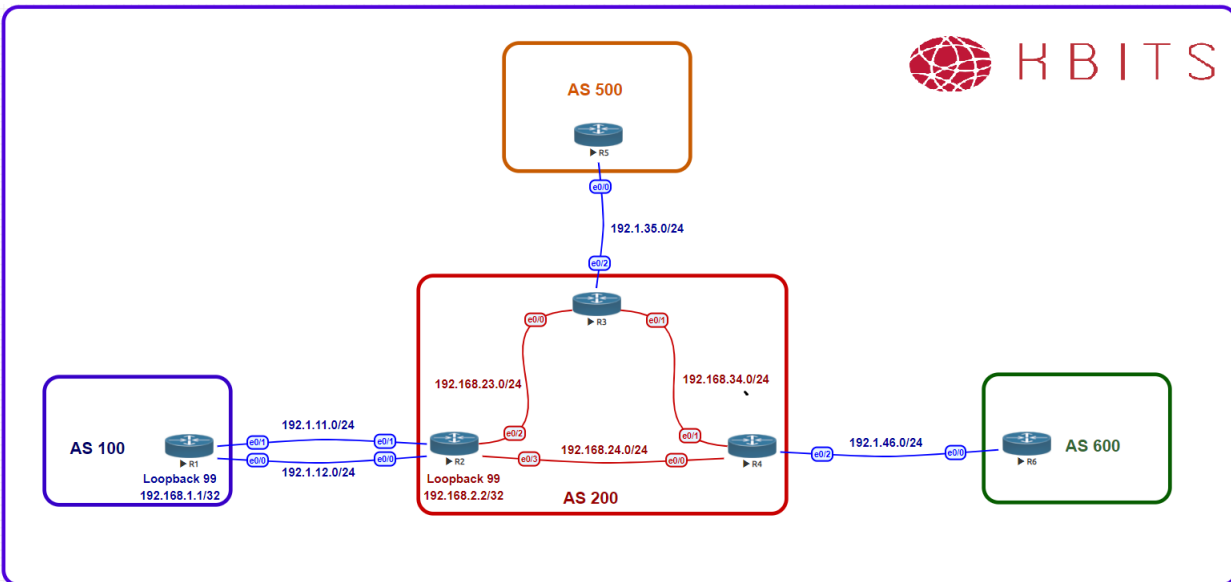


## Task 1

Configure MD5 Authentication between all eBGP peers using a password of **ccie123**.

<p><b>R1</b></p> <p>Router BGP 100 Neighbor 192.168.2.2 password ccie123</p>	<p><b>R2</b></p> <p>Router BGP 200 Neighbor 192.168.2.2 password ccie123</p>
<p><b>R3</b></p> <p>Router BGP 200 Neighbor 192.1.35.5 password ccie123</p>	<p><b>R5</b></p> <p>Router BGP 500 Neighbor 192.1.35.3 password ccie123</p>
<p><b>R4</b></p> <p>Router BGP 200 Neighbor 192.1.46.6 password ccie123</p>	<p><b>R6</b></p> <p>Router BGP 600 Neighbor 192.1.46.4 password ccie123</p>

# Lab 5 – Configuring iBGP with Route Reflectors



## Task 1

Configure IS-IS as the IGP to route the Loopback10 networks within AS 200. Configure IS-IS with a 24-bit metric. The IS-IS neighbors should maintain a Level-2 database only. Use the following for the NET address:

R2 – 49.0000.2222.2222.2222.00

R3 – 49.0000.3333.3333.3333.00

R4 – 49.0000.4444.4444.4444.00

### R2

```
Router isis
  Net 49.0000.2222.2222.2222.00
  Is-type level-2
  Metric-style wide
  !
Interface loopback10
  Ip router isis
  !
Interface E 0/2
  Ip router isis
  !
Interface E 0/3
  Ip router isis
```

### R3

```
Router isis
  Net 49.0000.3333.3333.3333.00
  Is-type level-2
  Metric-style wide
  !
Interface loopback10
  Ip router isis
  !
Interface E 0/0
  Ip router isis
  !
Interface E 0/1
  Ip router isis
```

**R4**

```

Router isis
  Net 49.0000.4444.4444.4444.00
  Is-type level-2
  Metric-style wide
!
Interface loopback10
  Ip router isis
!
Interface E 0/0
  Ip router isis
!
Interface E 0/1
  Ip router isis

```

**Task 2**

Configure an iBGP neighbor relationship between R2 & R3. The neighbor relationship should be configured with redundancy in mind. Make sure that the eBGP routes are propagated and injected into the BGP table.

**R2**

```

Router BGP 200
  Neighbor 10.3.3.3 remote-as 200
  Neighbor 10.3.3.3 update-source loop10
  Neighbor 10.3.3.3 next-hop-self

```

**R2**

```

Router BGP 200
  Neighbor 10.2.2.2 remote-as 200
  Neighbor 10.2.2.2 update-source loop10
  Neighbor 10.2.2.2 next-hop-self

```

**Task 3**

Configure an iBGP neighbor relationship between R3 & R4. The neighbor relationship should be configured with redundancy in mind. Make sure that the eBGP routes are propagated and injected into the BGP table.

**R3**

```

Router BGP 200
  Neighbor 10.4.4.4 remote-as 200
  Neighbor 10.4.4.4 update-source loop10
  Neighbor 10.4.4.4 next-hop-self

```

**R4**

```

Router BGP 200
  Neighbor 10.3.3.3 remote-as 200
  Neighbor 10.3.3.3 update-source loop10
  Neighbor 10.3.3.3 next-hop-self

```

### Verification:

- Make sure that AS 500 Loopbacks can reach the Loopback interfaces in AS 100, AS 200 & AS 600.
- Try the reachability between the AS 100 & AS 600 loopbacks? Are they reachable?

### Task 4

Re-configure R3 such that it propagates the routes from R2 towards R4 and vice versa. Use Peer-group to accomplish this task.

#### R3

```
Router BGP 200
No neighbor 10.2.2.2
No neighbor 10.4.4.4
Neighbor IBGP peer-group
Neighbor IBGP remote-as 200
Neighbor IBGP update-source Loopback10
Neighbor IBGP next-hop-self
Neighbor IBGP route-reflector-client
Neighbor 10.2.2.2 peer-group IBGP
Neighbor 10.4.4.4 peer-group IBGP
```

### Verification:

- Try the reachability between the AS 100 & AS 600 loopbacks? Are they reachable?
- Trace a packet from R1 to R6 (1.1.1.1 to 6.6.6.6). What path does it take?

### Task 5

You would like the RR to be an “inline RR”. This is for the purpose of future Data Filtering. Configure R3 to accomplish this.

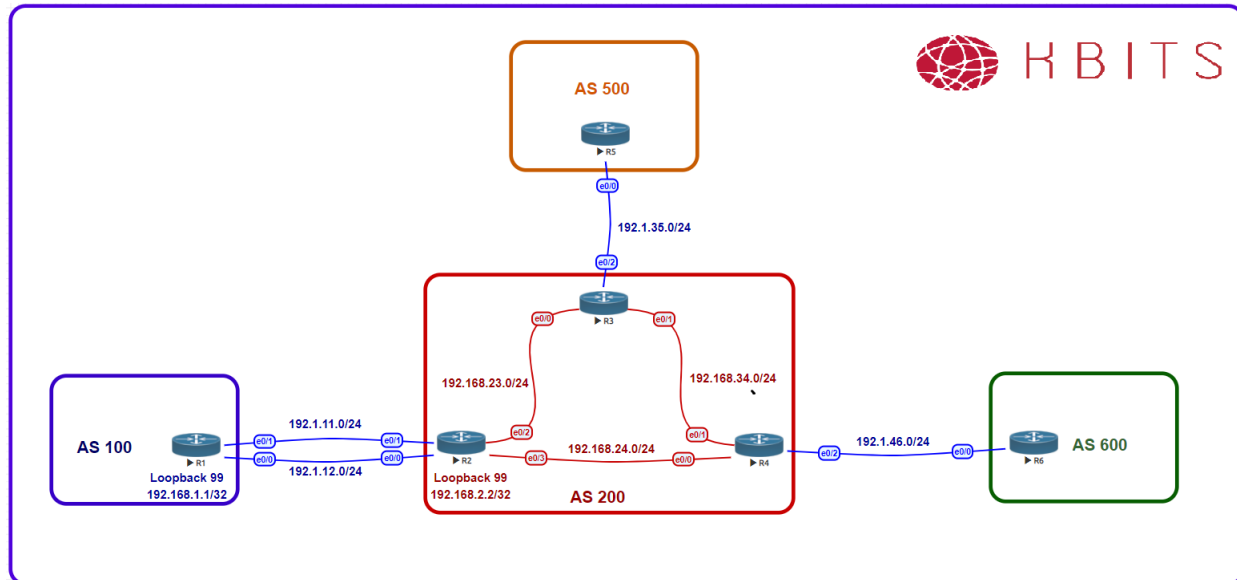
#### R3

```
Router BGP 200
Neighbor IBGP next-hop-self all
```

**Verification:**

- Trace a packet from R1 to R6 (1.1.1.1 to 6.6.6.6). What path does it take?

# Lab 6 – Route Filtering using ACLs



## Task 1

Create the following Loopbacks on R2

- Loopback 1 – 192.2.1.1/24
- Loopback 2 – 192.2.2.1/24
- Loopback 3 – 192.2.3.1/24
- Loopback 4 – 192.2.4.1/24
- Loopback 5 – 192.2.5.1/24
- Loopback 6 – 192.2.6.1/24
- Loopback 7 – 192.2.7.1/24
- Loopback 8 – 192.2.8.1/24

## R2

```
interface Loopback1
ip address 192.2.1.1 255.255.255.0
!
interface Loopback2
ip address 192.2.2.1 255.255.255.0
!
interface Loopback3
ip address 192.2.3.1 255.255.255.0
!
interface Loopback4
ip address 192.2.4.1 255.255.255.0
!
```

```
interface Loopback5
 ip address 192.2.5.1 255.255.255.0
!
interface Loopback6
 ip address 192.2.6.1 255.255.255.0
!
interface Loopback7
 ip address 192.2.7.1 255.255.255.0
!
interface Loopback8
 ip address 192.2.8.1 255.255.255.0
```

## **Task 2**

Advertise the newly created routes in BGP. Do not use the network command to accomplish this. These routes should have an origin code of “igp”.

## **R2**

```
Access-list 1 permit 192.2.1.1 0.0.0.255
Access-list 1 permit 192.2.2.1 0.0.0.255
Access-list 1 permit 192.2.3.1 0.0.0.255
Access-list 1 permit 192.2.4.1 0.0.0.255
Access-list 1 permit 192.2.5.1 0.0.0.255
Access-list 1 permit 192.2.6.1 0.0.0.255
Access-list 1 permit 192.2.7.1 0.0.0.255
Access-list 1 permit 192.2.8.1 0.0.0.255
!
Route-map RC permit 10
 Match address 1
 Set origin igp
!
Router bgp 200
 Redistribute connected route-map RC
```

### Task 3

Configure R2 such that it blocks all the 192.2.X.0 routes that have an odd number in the third octet from propagating outside the local AS. Use the distribute-list command to accomplish this task.

#### R2

```
Access-list 2 deny 192.2.1.0 0.0.254.255
Access-list 2 permit any
!
Router bgp 200
Neighbor 192.168.1.1 distribute-list 2 out
```

### Task 4

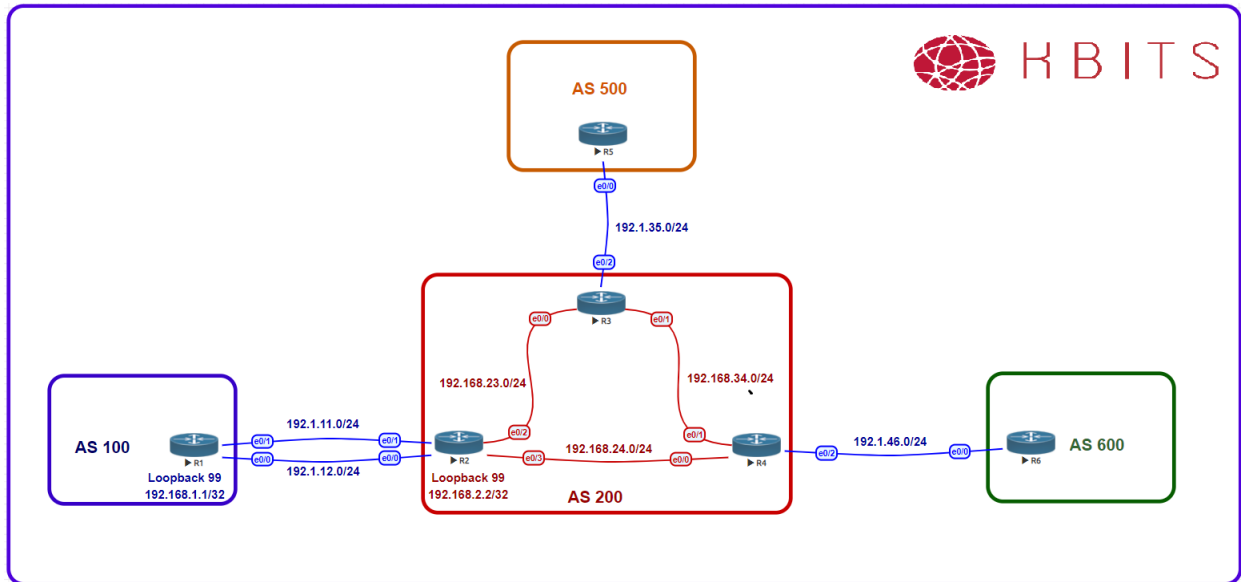
Configure R3 such that it blocks all the 192.2.X.0 routes that have an even number in the third octet from propagating from R2. Use the distribute-list command to accomplish this task.

#### R3

```
Access-list 1 deny 192.2.0.0 0.0.254.255
Access-list 1 permit any
!
Router bgp 200
Neighbor 10.2.2.2 distribute-list 1 in
```



# Lab 7 – Route Filtering using Prefix-Lists



## Task 1

Create the following Loopbacks on R3

- Loopback 1 – 150.3.16.1/20
- Loopback 2 – 150.3.36.1/22
- Loopback 3 – 150.3.40.1/22
- Loopback 4 – 150.3.50.1/23
- Loopback 5 – 150.3.65.1/24
- Loopback 6 – 150.13.0.1/16
- Loopback 7 – 150.14.64.1/18

## R3

```
interface Loopback1
 ip address 150.3.16.1 255.255.240.0
!
interface Loopback2
 ip address 150.3.36.1 255.255.252.0
!
interface Loopback3
 ip address 150.3.40.1 255.255.252.0
!
interface Loopback4
 ip address 150.3.50.1 255.255.254.0
!
interface Loopback5
```

```
ip address 150.3.65.1 255.255.255.0
!  
interface Loopback6  
ip address 150.13.0.1 255.255.0.0  
!  
interface Loopback7  
ip address 150.14.64.1 255.255.192.0
```

### **Task 2**

Advertise the newly created routes in BGP using the Network command.

### **R3**

```
Router bgp 200  
Network 150.3.16.0 mask 255.255.240.0  
Network 150.3.36.0 mask 255.255.252.0  
Network 150.3.40.0 mask 255.255.252.0  
Network 150.3.50.0 mask 255.255.254.0  
Network 150.3.65.0 mask 255.255.255.0  
Network 150.13.0.0  
Network 150.14.64.0 mask 255.255.192.0
```

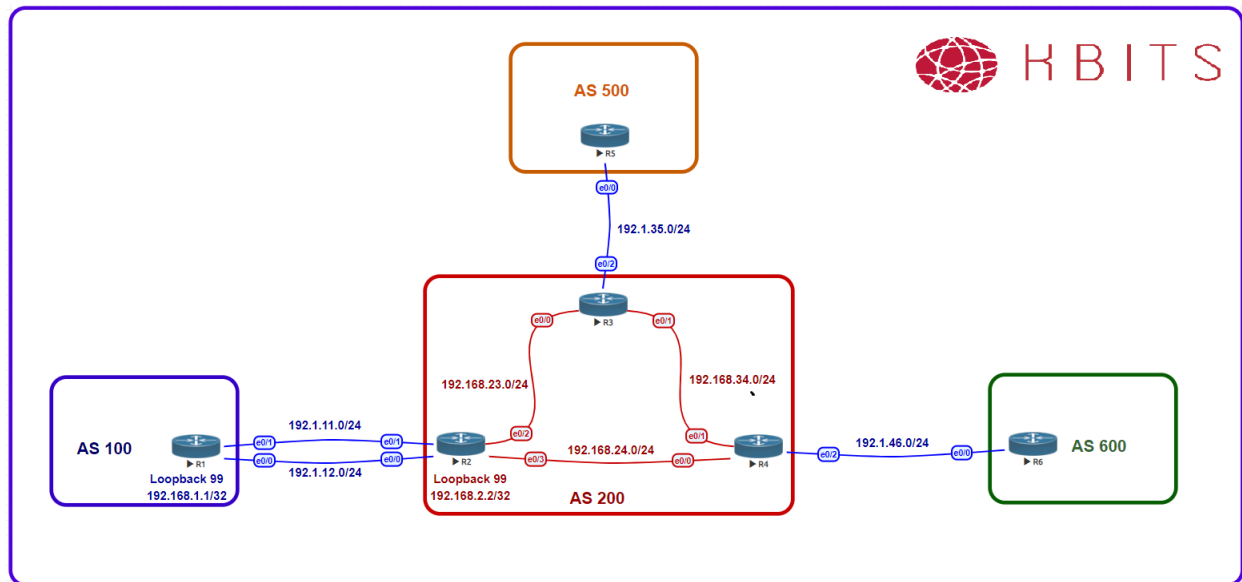
### **Task 3**

Configure R2 such that it blocks all the 150.X.X.0 routes that have a subnet mask between 17 and 23 bits coming in from R3.

### **R2**

```
IP Prefix-list PLIST1 deny 150.0.0.0/8 ge 17 le 23  
IP Prefix-list PLIST1 permit 0.0.0.0/0 le 32  
!  
Router bgp 200  
Neighbor 10.3.3.3 prefix-list PLIST1 in
```

# Lab 8 – Route Filtering using AS Path-Filter



## Task 1

Configure BGP such that AS 500 does not use AS 200 as a transit AS. Configuration should be done in AS 500.

### R5

```
Ip as-path access-list 1 permit ^200$
!
Router BGP 500
Neighbor 192.1.35.3 filter-list 1 in
```

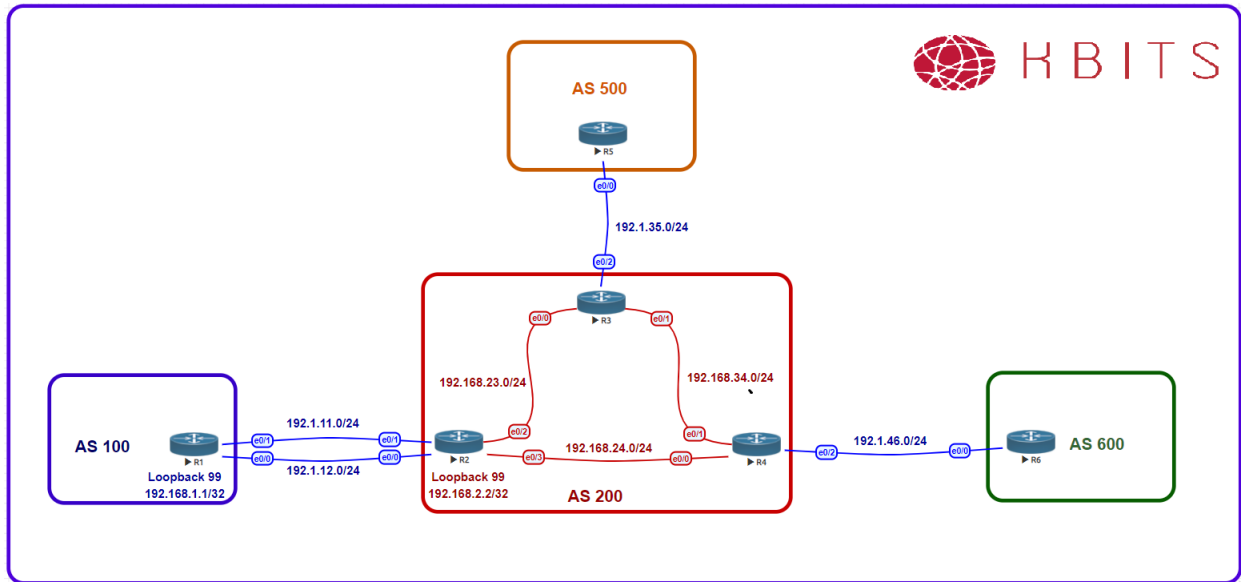
## Task 2

Configure BGP such that AS 100 does not use AS 200 to get AS 500 routes. Configuration should be done in AS 200. You are only allowed a single line in the AS-path filter.

### R2

```
Ip as-path access-list 1 permit ^$
!
Router BGP 200
Neighbor 192.168.1.1 filter-list 1 out
```

# Lab 9 – Configuring Route Aggregation – Summary Only



## Task 1

Create the following Loopbacks on R3 and advertise them under BGP:

- Loopback 1 – 203.1.4.1/24
- Loopback 2 – 203.1.5.1/24
- Loopback 3 – 203.1.6.1/24
- Loopback 4 – 203.1.7.1/24

## R3

```
interface Loopback1
 ip address 203.1.4.1 255.255.255.0
!
interface Loopback2
 ip address 203.1.5.1 255.255.255.0
!
interface Loopback3
 ip address 203.1.6.1 255.255.255.0
!
interface Loopback4
 ip address 203.1.7.1 255.255.255.0
!
Router BGP 200
 Network 203.1.4.0
 Network 203.1.5.0
 Network 203.1.6.0
```

```
Network 203.1.7.0
```

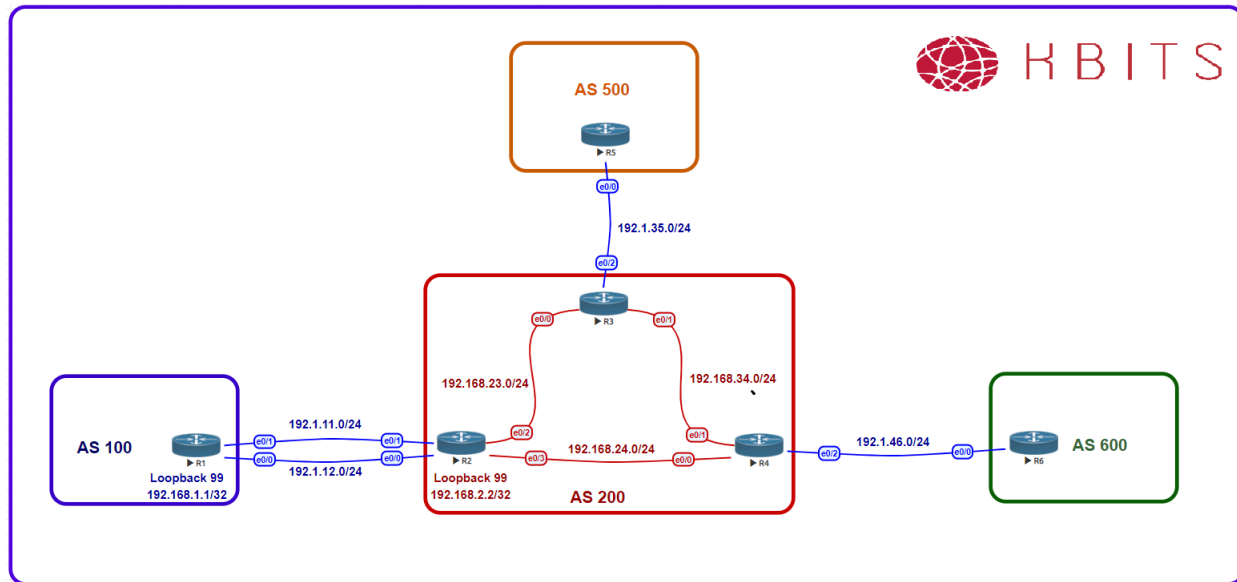
### **Task 2**

Configure Route Aggregation on R3 such that these routes are summarized as a single route. Only the Summary route should be send to R3's neighbors.

### **R3**

```
Router bgp 200  
Aggregate-address 203.1.4.0 255.255.252.0 summary-only
```

# Lab 10 – Configuring Route Aggregation – Manual Filtering



## Task 1

Create the following Loopbacks on R4 and advertise them under BGP:

- Loopback 1 – 204.1.4.1/24
- Loopback 2 – 204.1.5.1/24
- Loopback 3 – 204.1.6.1/24
- Loopback 4 – 204.1.7.1/24

## R4

```
interface Loopback1
ip address 204.1.4.1 255.255.255.0
!
interface Loopback2
ip address 204.1.5.1 255.255.255.0
!
interface Loopback3
ip address 204.1.6.1 255.255.255.0
!
interface Loopback4
ip address 204.1.7.1 255.255.255.0
!
Router BGP 200
```

```
Network 204.1.4.0
Network 204.1.5.0
Network 204.1.6.0
Network 204.1.7.0
```

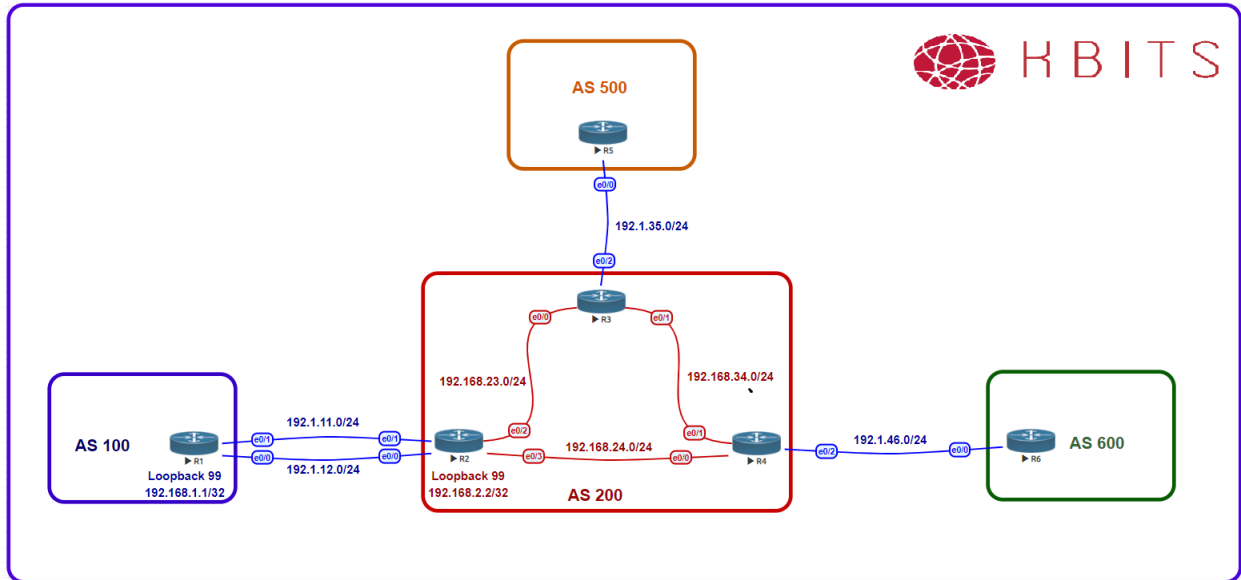
## **Task 2**

Configure Route Aggregation on R4 such that these routes are summarized as a single route. Only the Summary Route should be sent towards the eBGP neighbor (R6). Only the Specific Routes should be sent towards the iBGP neighbor (R3). The routes should not be seen as suppressed on R4.

## **R4**

```
IP Prefix-list PLIST-R6 deny 204.1.4.0/22 ge 24
IP Prefix-list PLIST-R6 permit 0.0.0.0/0 le 32
!
IP Prefix-list PLIST-R3 deny 204.1.4.0/22
IP Prefix-list PLIST-R3 permit 0.0.0.0/0 le 32
!
Router bgp 200
  Aggregate-address 204.1.4.0 255.255.252.0
  Neighbor 192.1.46.6 prefix-list PLIST-R6 out
  Neighbor 10.3.3.3 prefix-list PLIST-R3 out
```

# Lab 11 – Configuring Route Aggregation – Suppress Maps



## Task 1

Create the following Loopbacks on R2 and advertise them under BGP:

- Loopback 1 – 202.1.4.1/24
- Loopback 2 – 202.1.5.1/24
- Loopback 3 – 202.1.6.1/24
- Loopback 4 – 202.1.7.1/24

## R2

```
interface Loopback1
ip address 202.1.4.1 255.255.255.0
!
interface Loopback2
ip address 202.1.5.1 255.255.255.0
!
interface Loopback3
ip address 202.1.6.1 255.255.255.0
!
interface Loopback4
ip address 202.1.7.1 255.255.255.0
!
Router BGP 234
```



```
Network 202.1.4.0
Network 202.1.5.0
Network 202.1.6.0
Network 202.1.7.0
```

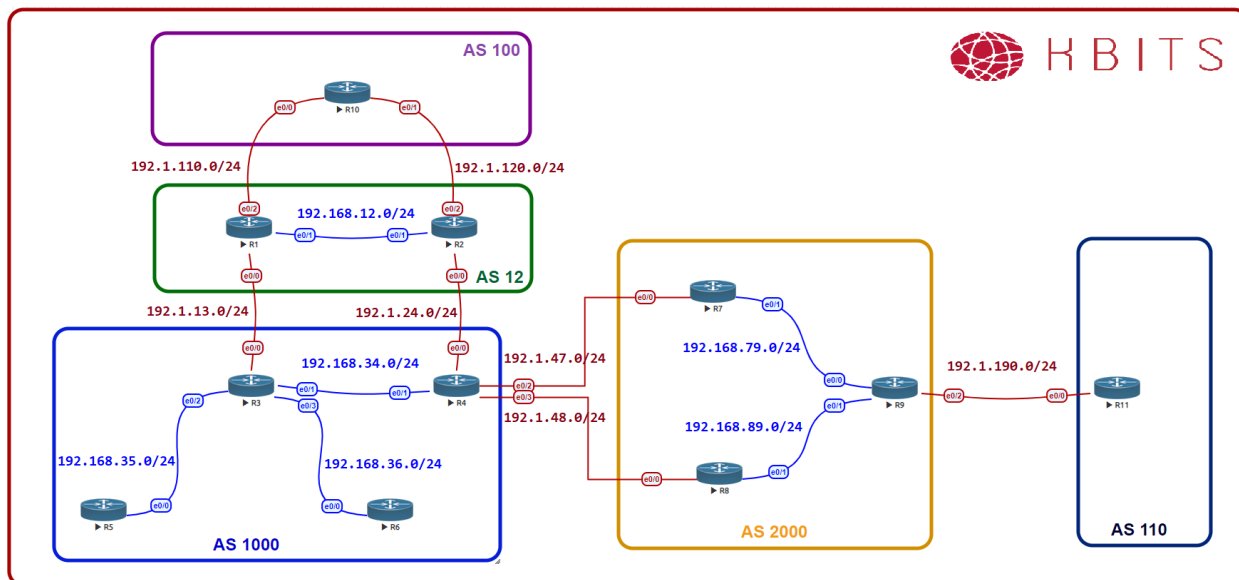
### **Task 2**

Configure Route Aggregation on R2 such that these routes are summarized as a single route. Only the Summary route and the 202.1.5.0 route should be sent to R2's neighbors. The other specific routes should be seen as suppressed on R2.

### **R2**

```
Access-list 5 permit 202.1.5.0 0.0.0.255
!
Route-map SUPMAP deny 10
  Match address 5
Route-map SUPMAP permit 20
!
Router bgp 200
Aggregate-address 202.1.4.0 255.255.252.0 supress-map SUPMAP
```

# Lab 12 – Configuring Base BGP Topology – eBGP & iBGP



## Interface IP Address Configuration

### R1

Interface	IP Address	Subnet Mask
Loopback 0	1.1.1.1	255.0.0.0
Loopback 1	11.1.1.1	255.255.255.0
Loopback 10	10.1.1.1	255.255.255.255
E 0/0	192.1.13.1	255.255.255.0
E 0/1	192.168.12.1	255.255.255.0
E 0/2	192.1.110.1	255.255.255.0

### R2

Interface	IP Address	Subnet Mask
Loopback 0	2.2.2.2	255.0.0.0
Loopback 1	22.2.2.2	255.255.255.0
Loopback 10	10.2.2.2	255.255.255.255
E 0/0	192.1.24.2	255.255.255.0
E 0/1	192.168.12.2	255.255.255.0
E 0/2	192.1.120.2	255.255.255.0

### R3

Interface	IP Address	Subnet Mask
Loopback 0	3.3.3.3	255.0.0.0
Loopback 1	33.3.3.3	255.255.255.0
Loopback 10	10.3.3.3	255.255.255.255
E 0/0	192.1.13.3	255.255.255.0
E 0/1	192.168.34.3	255.255.255.0
E 0/2	192.168.35.3	255.255.255.0
E 0/3	192.168.36.3	255.255.255.0

### R4

Interface	IP Address	Subnet Mask
Loopback 0	4.4.4.4	255.0.0.0
Loopback 1	44.4.4.4	255.255.255.0
Loopback 10	10.4.4.4	255.255.255.255
E 0/0	192.1.24.4	255.255.255.0
E 0/1	192.168.34.4	255.255.255.0
E 0/2	192.1.47.4	255.255.255.0
E 0/3	192.1.48.4	255.255.255.0

### R5

Interface	IP Address	Subnet Mask
Loopback 0	5.5.5.5	255.0.0.0
Loopback 1	55.5.5.5	255.255.255.0
E 0/0	192.168.35.5	255.255.255.0

### R6

Interface	IP Address	Subnet Mask
Loopback 0	6.6.6.6	255.0.0.0
Loopback 1	66.6.6.6	255.255.255.0
E 0/0	192.168.36.6	255.255.255.0

## R7

Interface	IP Address	Subnet Mask
Loopback 0	7.7.7.7	255.0.0.0
Loopback 1	77.7.7.7	255.255.255.0
Loopback 10	10.7.7.7	255.255.255.255
E 0/0	192.1.47.7	255.255.255.0
E 0/1	192.168.79.7	255.255.255.0

## R8

Interface	IP Address	Subnet Mask
Loopback 0	8.8.8.8	255.0.0.0
Loopback 1	88.8.8.8	255.255.255.0
Loopback 10	10.8.8.8	255.255.255.255
E 0/0	192.1.48.8	255.255.255.0
E 0/1	192.168.89.8	255.255.255.0

## R9

Interface	IP Address	Subnet Mask
Loopback 0	9.9.9.9	255.0.0.0
Loopback 1	99.9.9.9	255.255.255.0
Loopback 10	10.9.9.9	255.255.255.255
E 0/0	192.168.79.9	255.255.255.0
E 0/1	192.168.89.9	255.255.255.0
E 0/2	192.1.190.9	255.255.255.0

## R10

Interface	IP Address	Subnet Mask
Loopback 0	100.100.100.10	255.0.0.0
Loopback 1	101.101.101.10	255.255.255.0
E 0/0	192.1.110.10	255.255.255.0
E 0/1	192.1.120.10	255.255.255.0

## R11

Interface	IP Address	Subnet Mask
Loopback 0	111.111.111.11	255.0.0.0
Loopback 1	112.112.112.11	255.255.255.0
E 0/0	192.1.190.11	255.255.255.0

### Task 1

Configure eBGP neighbor relationships between R10 in AS 100 with R1 & R2 in AS 12. Advertise all public loopback networks in BGP.

<b>R1</b> Router BGP 12 Network 1.0.0.0 Network 11.1.1.0 mask 255.255.255.0 Neighbor 192.1.110.10 remote-as 100	<b>R2</b> Router BGP 12 Network 2.0.0.0 Network 22.2.2.0 mask 255.255.255.0 Neighbor 192.1.120.10 remote-as 100
<b>R10</b> Router BGP 100 Network 100.0.0.0 Network 101.101.101.0 mask 255.255.255.0 Neighbor 192.1.110.1 remote-as 12 Neighbor 192.1.120.1 remote-as 12	

### Task 2

Configure iBGP neighbor relationships between R1 & R2 in AS 12.  
Configure the neighbor relationship based on a private loopback address.  
Use EIGRP 12 as the underlay IGP.

<b>R1</b> Router eigrp 12 Network 192.168.12.0 Network 10.0.0.0 ! Router BGP 12 Neighbor 10.2.2.2 remote-as 12 Neighbor 10.2.2.2 update-source Lo10 Neighbor 10.2.2.2 next-hop-self	<b>R2</b> Router eigrp 12 Network 192.168.12.0 Network 10.0.0.0 ! Router BGP 12 Neighbor 10.1.1.1 remote-as 12 Neighbor 10.1.1.1 update-source Lo10 Neighbor 10.1.1.1 next-hop-self
---	---

### Task 3

Configure eBGP neighbor relationships between R1 in AS 12 and R3 in AS 1000. Advertise all public loopback networks on R3 in BGP.

<b>R1</b>  Router BGP 12 Neighbor 192.1.13.3 remote-as 1000	<b>R3</b>  Router BGP 1000 Network 3.0.0.0 Network 33.3.3.0 mask 255.255.255.0 Neighbor 192.1.13.1 remote-as 12
--	--

### Task 4

Configure eBGP neighbor relationships between R2 in AS 12 and R4 in AS 1000. Advertise all public loopback networks on R4 in BGP.

<b>R2</b>  Router BGP 12 Neighbor 192.1.24.4 remote-as 1000	<b>R4</b>  Router BGP 1000 Network 4.0.0.0 Network 44.4.4.0 mask 255.255.255.0 Neighbor 192.1.24.2 remote-as 12
--	--

### Task 5

Configure iBGP neighbor relationships between R3 & R4 in AS 1000. Configure the neighbor relationship based on the physical link.

<b>R3</b>  Router BGP 1000 Neighbor 192.168.34.4 remote-as 1000 Neighbor 192.168.34.4 next-hop-self	<b>R4</b>  Router BGP 1000 Neighbor 192.168.34.3 remote-as 1000 Neighbor 192.168.34.3 next-hop-self
---	---

### Task 6

Configure OSPF as the IGP to connect R3 to R6 in Area 0. Only enable OSPF on the R3-R6 physical link on R3. Enable OSPF on all interfaces on R6 in area 0. Configure Mutual Redistribution on R3 between OSPF and BGP

<b>R3</b> Router ospf 1 Network 192.168.36.0 0.0.0.255 area 0 Redistribute bgp 1000 ! Router bgp 1000 Redistribute ospf 1	<b>R6</b> Router ospf 1 Network 6.0.0.0 0.255.255.255 area 0 Network 66.0.0.0 0.255.255.255 area 0 Network 192.168.36.0 0.0.0.255 area 0
---	--

### Task 7

Configure iBGP neighbor relationships between R3 & R5 in AS 1000. Configure the neighbor relationship based on the physical link.

<b>R3</b> Router BGP 1000 Neighbor 192.168.35.5 remote-as 1000 Neighbor 192.168.35.5 next-hop-self	<b>R5</b> Router BGP 1000 Neighbor 192.168.35.3 remote-as 1000 Neighbor 192.168.35.3 next-hop-self
---	---

### Task 8

Configure eBGP neighbor relationships between R4 in AS 1000 with R7 & R8 in AS 2000. Advertise all public loopback networks in BGP on R7 & R8.

<b>R7</b> Router BGP 2000 Network 7.0.0.0 Network 77.7.7.0 mask 255.255.255.0 Neighbor 192.1.47.4 remote-as 1000	<b>R8</b> Router BGP 2000 Network 8.0.0.0 Network 88.8.8.0 mask 255.255.255.0 Neighbor 192.1.48.4 remote-as 1000
<b>R4</b> Router BGP 1000 Neighbor 192.1.47.4 remote-as 2000 Neighbor 192.1.48.4 remote-as 2000	

### Task 9

Configure iBGP neighbor relationships between R7, R8 & R9 in AS 2000. Advertise the public loopback addresses of R9 in BGP. Configure the neighbor relationship based on a private loopback address. Use IS-IS in area 49.0000 as the underlay IGP. Use System-ID on your choice. Configure R9 as the Route Reflector for R7 & R8. Do not configure a direct BGP peering between R7 & R8.

<b>R7</b>  Router isis Net 49.0000.7777.7777.7777.00 Is-type level-2 ! Interface E 0/1 Ip router isis ! Interface Loopback10 Ip router isis ! Router BGP 2000 Neighbor 10.9.9.9 remote-as 2000 Neighbor 10.9.9.9 update-source Lo10 Neighbor 10.9.9.9next-hop-self	<b>R8</b>  Router isis Net 49.0000.8888.8888.8888.00 Is-type level-2 ! Interface E 0/1 Ip router isis ! Interface Loopback10 Ip router isis ! Router BGP 2000 Neighbor 10.9.9.9 remote-as 2000 Neighbor 10.9.9.9 update-source Lo10 Neighbor 10.9.9.9next-hop-self
<b>R9</b>  Router isis Net 49.0000.9999.9999.9999.00 Is-type level-2 ! Interface E 0/0 Ip router isis ! Interface E 0/1 Ip router isis ! Interface Loopback10 Ip router isis ! Router BGP 2000 Network 9.0.0.0 Network 99.9.9.0 mask 255.255.255.0 Neighbor IBGP peer-group Neighbor IBGP remote-as 2000	



```
Neighbor IBGP update-source Lo10
Neighbor IBGP next-hop-self
Neighbor IBGP route-reflector-client
Neighbor 10.7.7.7 peer-group IBGP
Neighbor 10.8.8.8 peer-group IBGP
```

### **Task 10**

Configure eBGP neighbor relationships between R9 in AS 2000 and R11 in AS 110. Advertise all public loopback networks on R11 in BGP.

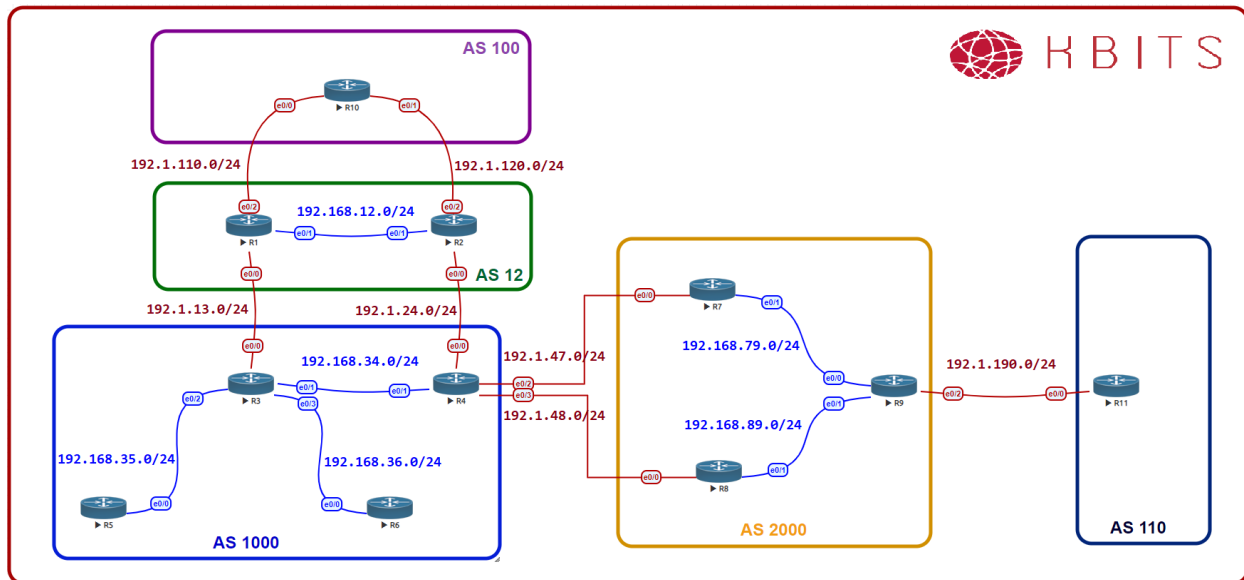
#### **R9**

```
Router BGP 2000
Neighbor 192.1.190.11 remote-as 110
```

#### **R11**

```
Router BGP 110
Network 111.0.0.0
Network 112.112.112.0 mask 255.255.255.0
Neighbor 192.1.190.9 remote-as 2000
```

# Lab 13 – Configuring BGP Attributes – Local Preference



## Task 1

Configure AS 2000 such that it prefers the Link between R4-R7 for traffic leaving AS 2000 towards AS 1000.

### R7

```
route-map SETATT permit 10
  set local-preference 111
!
router bgp 2000
  neighbor 192.1.47.4 route-map SETATT in
```

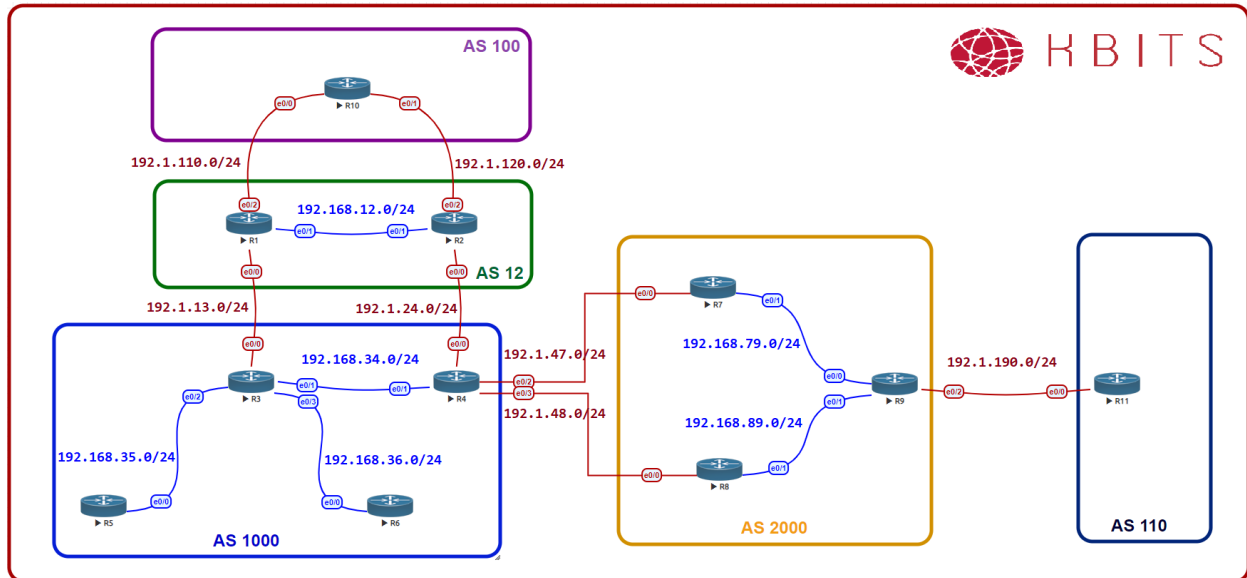
## Task 2

Configure AS 2000 such that it prefers the Link between R4-R8 for traffic destined towards 1.4.1.0/24 & 1.4.2.0/24 leaving AS 2000 towards AS 1000. R4-R8 should be preferred link only for 1.4.1.0/24 & 1.4.2.0/24. The rest should continue to use R4-R7.

### R8

```
ip prefix-list PL1 permit 1.4.1.0/24
ip prefix-list PL1 permit 1.4.2.0/24
!
route-map SETATT permit 10
  match ip address prefix PL1
  set local-preference 115
route-map SETATT permit 20
!
router bgp 2000
  neighbor 192.1.48.4 route-map SETATT in
```

# Lab 14 – Configuring BGP Attributes – MED



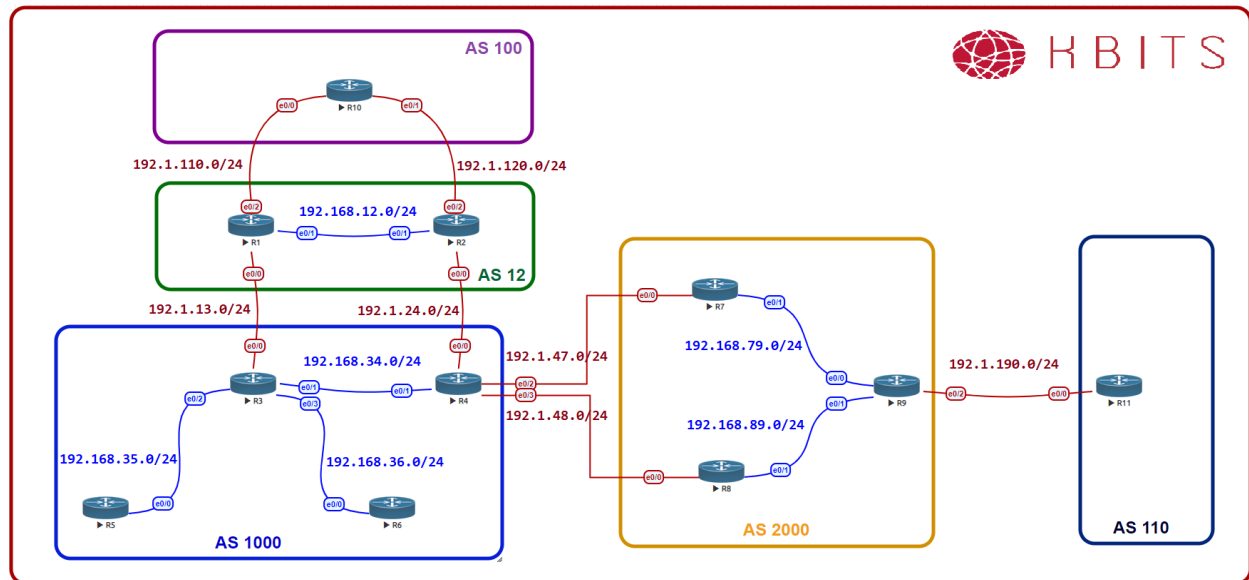
## Task 1

Configure AS 2000 such that it prefers the Link between R4-R8 for traffic entering AS 2000 from AS 1000.

### R7

```
route-map SETMED permit 10
  set metric 77
!
router bgp 2000
  neighbor 192.1.47.4 route-map SETMED out
```

# Lab 15 – Configuring BGP Attributes – Weight



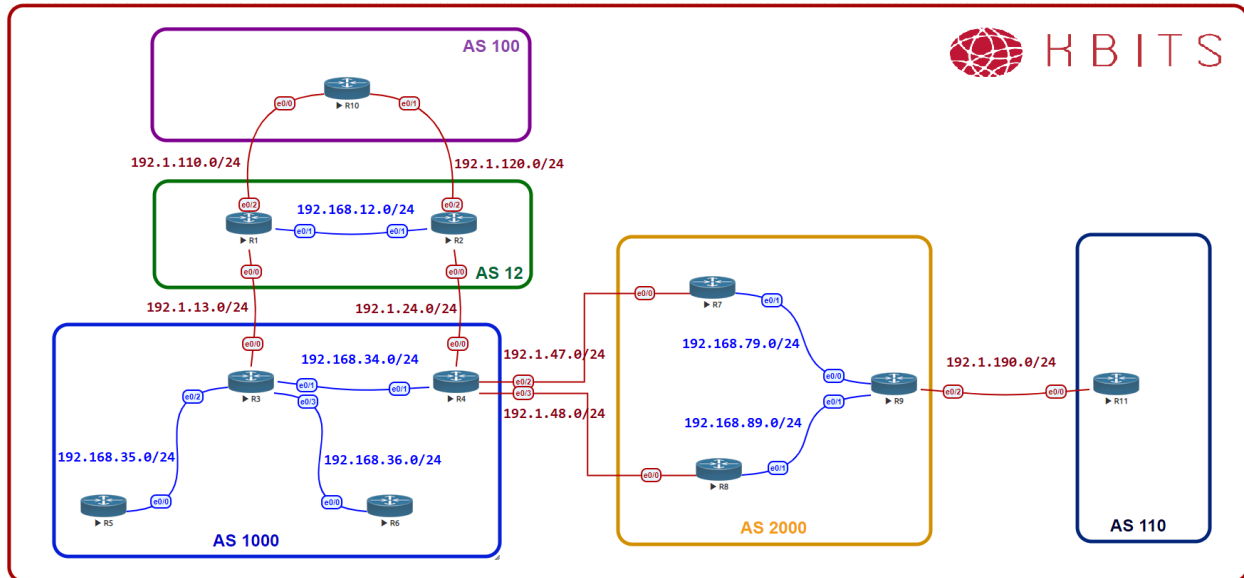
## Task 1

Configure R8 such that all traffic towards AS1000 should use the Link between R4 & R8 as the preferred link. This should only affect the local router and not the rest of the AS.

## R8

```
route-map SETWT
 set weight 88
!
router bgp 2000
 neighbor 192.1.48.4 route-map SETWT in
```

# Lab 16 – Configuring BGP Attributes – AS-Path



## Task 1

De-configure the Route-map from the previous 3 labs. This is done so that we can accomplish the same tasks using the AS-Path attribute

### R7

```
No route-map SETATT
No route-map SETMED
!
router bgp 2000
 no neighbor 192.1.47.4 route-map SETATT in
 no neighbor 192.1.47.4 route-map SETMED out
```

### R8

```
No route-map SETATT
No route-map SETWT
!
router bgp 2000
 no neighbor 192.1.48.4 route-map SETATT in
 no neighbor 192.1.48.4 route-map SETWT in
```

## Task 2

Configure AS 2000 such that it prefers the Link between R4-R7 for traffic leaving AS2000 towards AS1000. Use the AS-Path attribute to accomplish this task.

### R8

```
route-map SETAS permit 10
  set as-path prepend 1000
!
router bgp 2000
  neighbor 192.1.48.4 route-map SETAS in
```

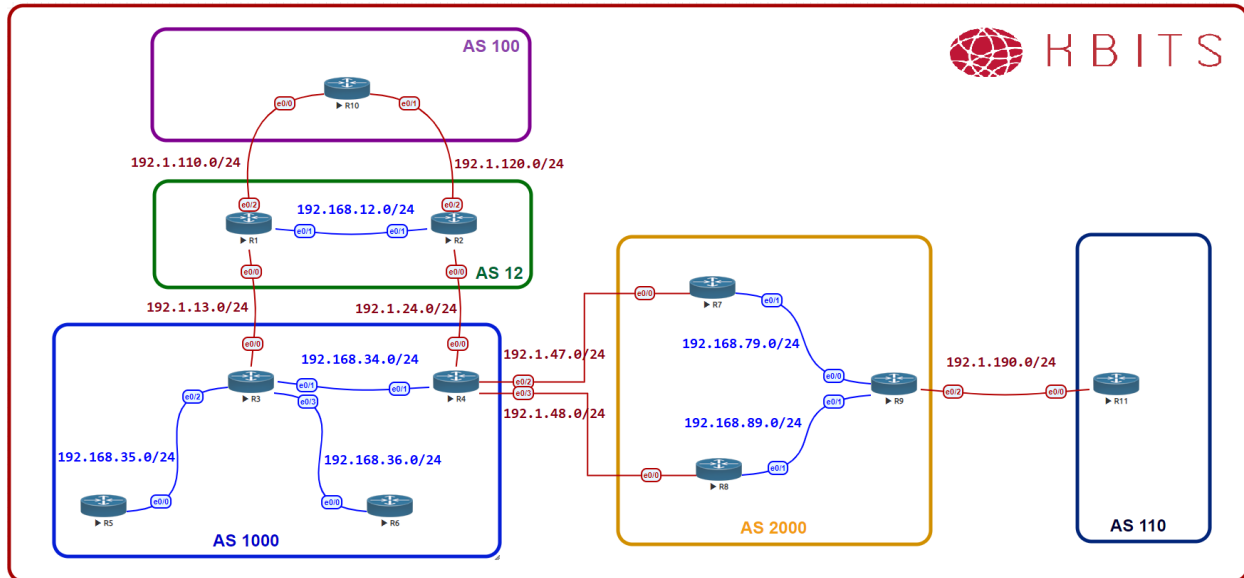
## Task 3

Configure AS 2000 such that it prefers the Link between R4-R8 for traffic entering AS2000 from AS1000. Use the AS-Path attribute to accomplish this task.

### R7

```
route-map SETAS permit 10
  set as-path prepend 2000
!
router bgp 2000
  neighbor 192.1.47.4 route-map SETAS out
```

# Lab 17 – Configuring BGP Attributes – No-Export Community Attribute



## Task 1

AS110 wants to limit the propagation of 111.0.0.0/8 network to AS2000 only. AS2000 should not export this route outside AS2000. Use the appropriate Community attribute to accomplish this.

### R11

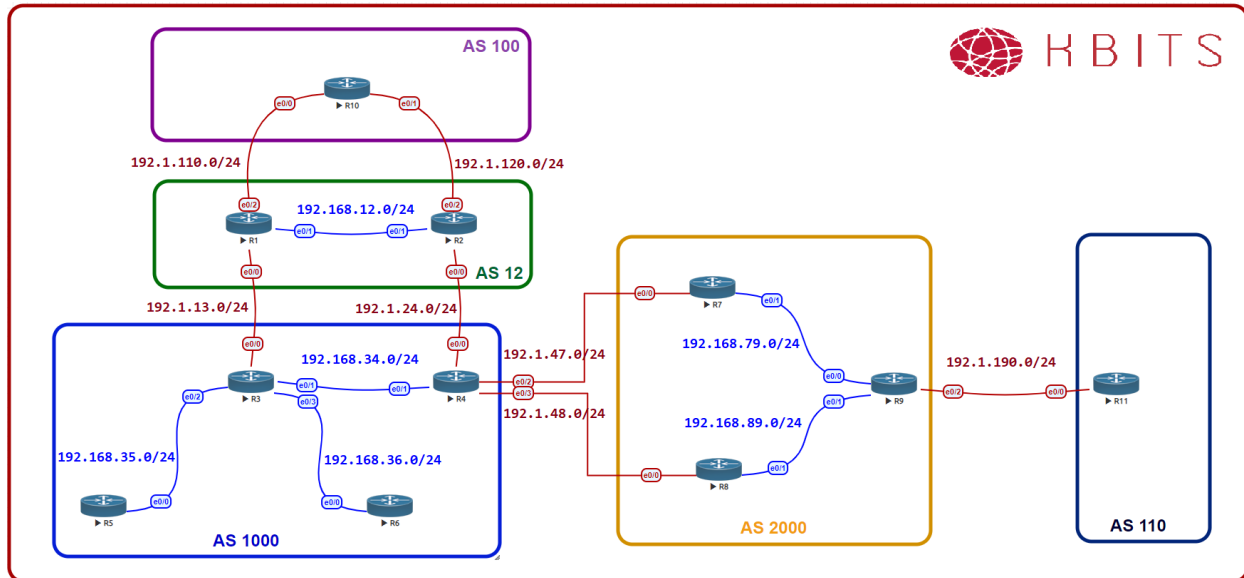
```
ip prefix-list PL1 permit 111.0.0.0/8
!
route-map SETCOMM permit 10
 match ip address prefix PL1
 set community no-export
route-map SETCOMM permit 20
!
router bgp 110
 neighbor 192.1.190.9 route-map SETCOMM out
 neighbor 192.1.190.9 send-community standard
```

### R9

```
router bgp 110
 neighbor 10.7.7.7 send-community standard
 neighbor 10.8.8.8 send-community standard
```



# Lab 18 – Configuring BGP Attributes – No-Advertise Community Attribute



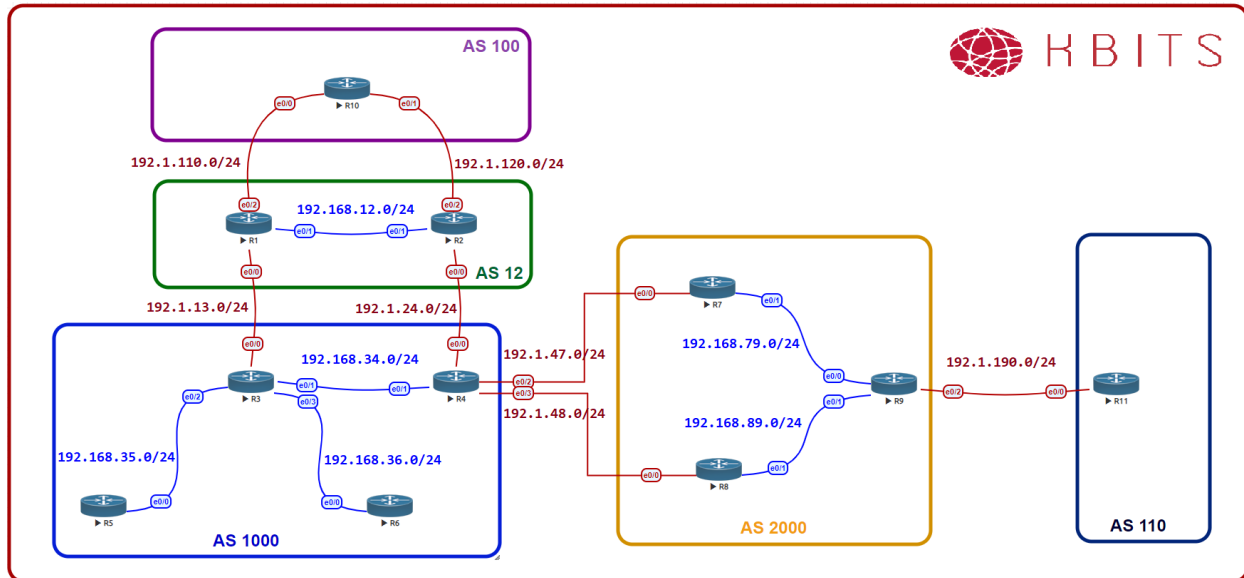
## Task 1

AS110 wants to limit the propagation of 112.112.112.0/24 network to R9 only. R9 should not forward this network to anyone including the iBGP Neighbors. Use the appropriate Community attribute to accomplish this.

## R11

```
ip prefix-list PL2 permit 112.112.112.0/24
!  
route-map SETCOMM permit 5  
  match ip address prefix PL2  
  set community no-advertise
```

# Lab 19 – Configuring BGP Conditional Advertisement



## Task 1

De-configure the Route-map from the previous 3 labs. This is done so that we have all the routes present for the next set of labs.

### R8

```
no route-map SETAS
!
router bgp 2000
no neighbor 192.1.48.4 route-map SETAS in
```

### R7

```
no route-map SETAS permit
!
router bgp 2000
no neighbor 192.1.47.4 route-map SETAS out
```

### R11

```
no route-map SETCOMM
!
router bgp 110
no neighbor 192.1.190.9 route-map SETCOMM out
```

## Task 2

Configure a loopback on R7 and advertise it thru BGP. This will be used to check the status of R7 in a later step.

### R7

```
Interface Loopback99
ip address 10.77.77.77 255.255.255.255
!
router bgp 2000
network 10.77.77.77 mask 255.255.255.255
```

## Task 3

Configure a route-map on R8 to classify the route that will be conditionally advertised.

### R8

```
ip prefix-list PL2 permit 111.0.0.0/8
ip prefix-list PL2 permit 112.112.112.0/24
!
route-map AMAP
match ip address prefix PL2
```

## Task 4

Configure a route-map on R8 to call an ACL that will indicate the absence of the 10.77.77.77/32 route.

### R8

```
ip prefix-list PL3 permit 10.77.77.77/32
!
route-map NEM
match ip address prefix PL3
```

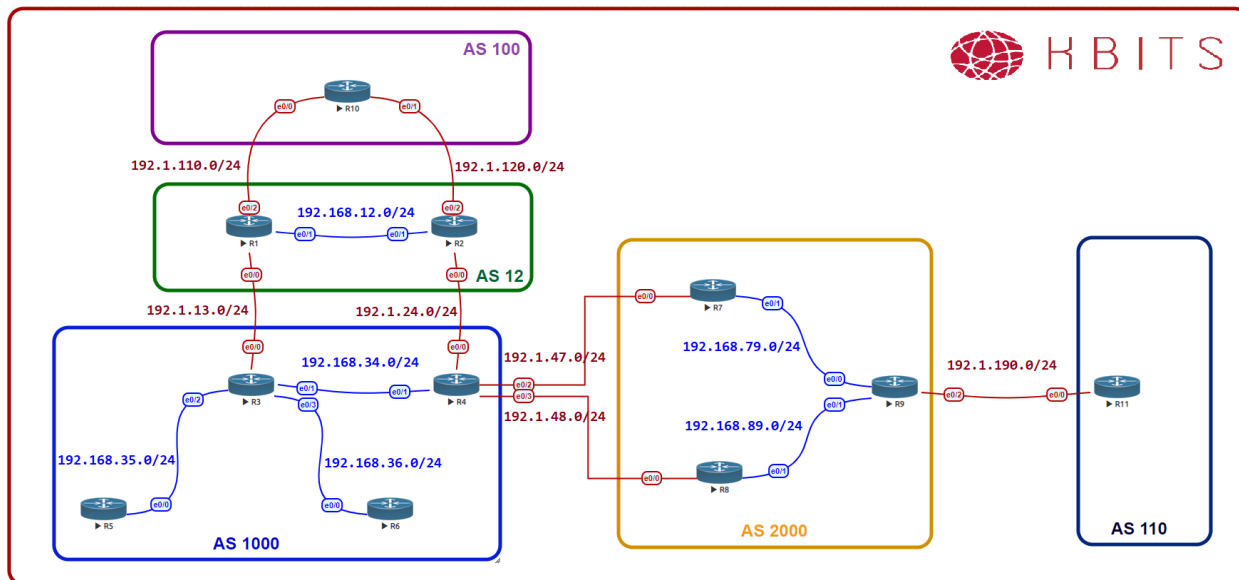
## Task 5

Configure the Conditional Advertisement of the 111.0.0.0/8 & 112.112.112.0/24 routes from R8 to R4 only if R7 is down.

### R8

```
router bgp 2000
neighbor 192.1.48.4 advertise-map AMAP non-exist-map NEM
```

# Lab 20 – Configuring BGP Multi-Path – eBGP – iBGP



## Task 1

Configure R10 to allow it to inject multiple routes on the Links between R10-R1 & R10-R2 (eBGP Neighbors).

### R10

```
Router bgp 100
maximum-paths 2
```

## Task 2

Configure R9 to allow it to inject multiple routes on the Links between R9-R7 & R9-R8 (iBGP Neighbors).

### R9

```
Router bgp 2000
maximum-paths ibgp 2
```

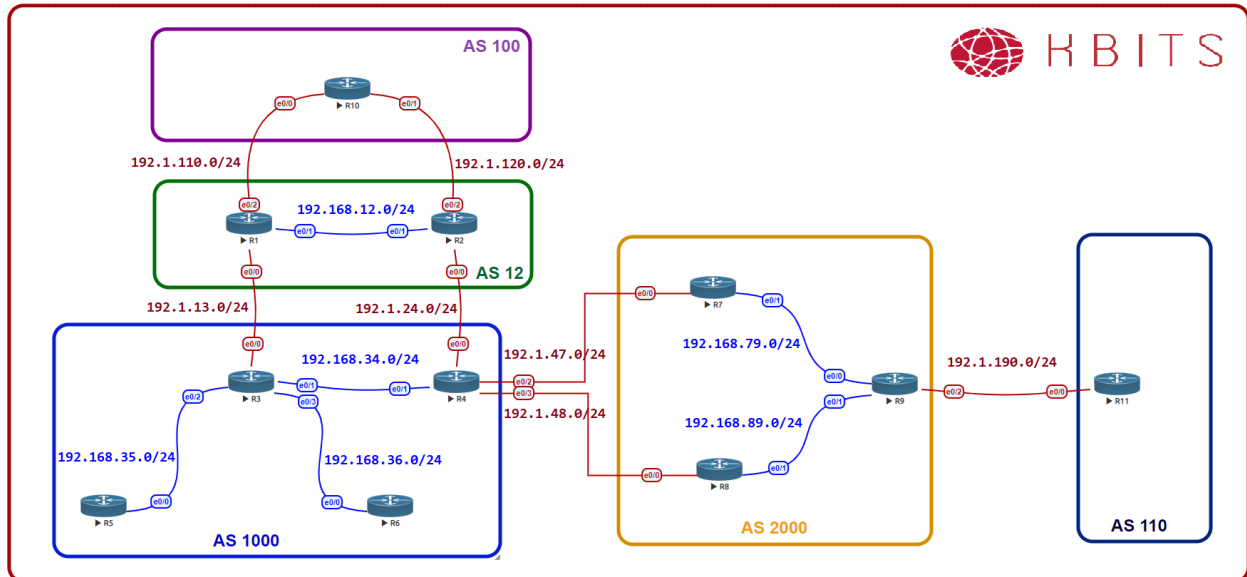
**Task 2**

Configure R3 to allow it to inject multiple routes on the Links between R1-R3(eBGP) & R3-R4 (iBGP Neighbors)

**R3**

```
Router bgp 1000
maximum-paths eibgp 2
```

# Lab 21 – Configuring to Redistribute iBGP Routes into IGP



## Task 1

Check the routing table of R6. Does it have all the routes from AS2000 & AS110?

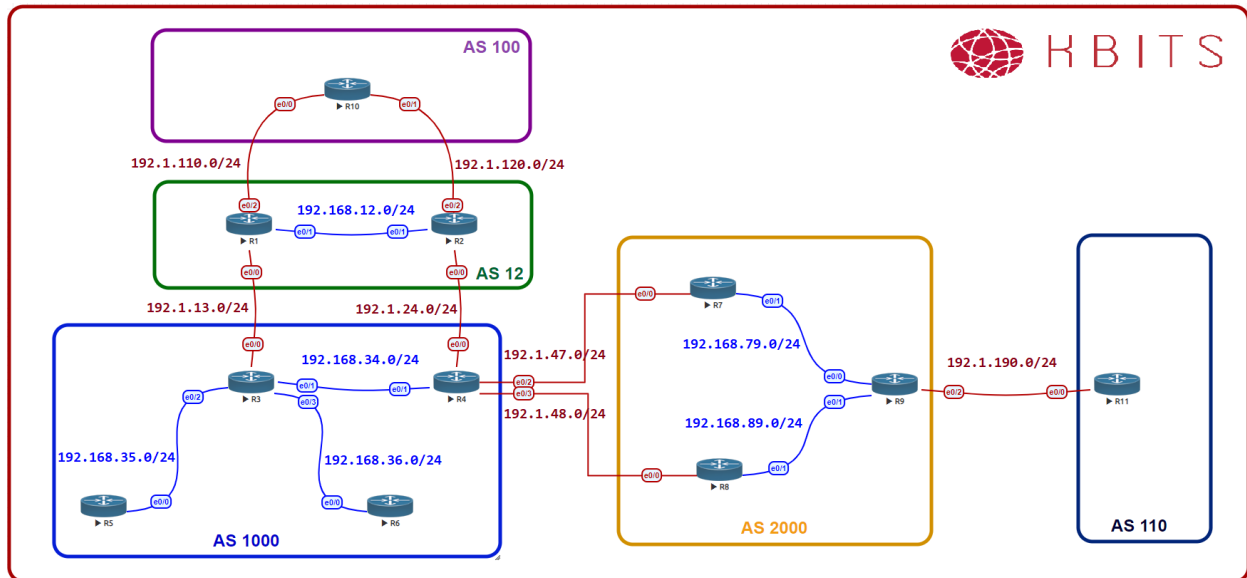
## Task 2

Configure R3 to redistribute iBGP routes into IGP.

### R3

```
router bgp 1000
  bgp redistribute-internal
```

# Lab 22 – Configuring BGP Route Reflector with Next-Hop Changed



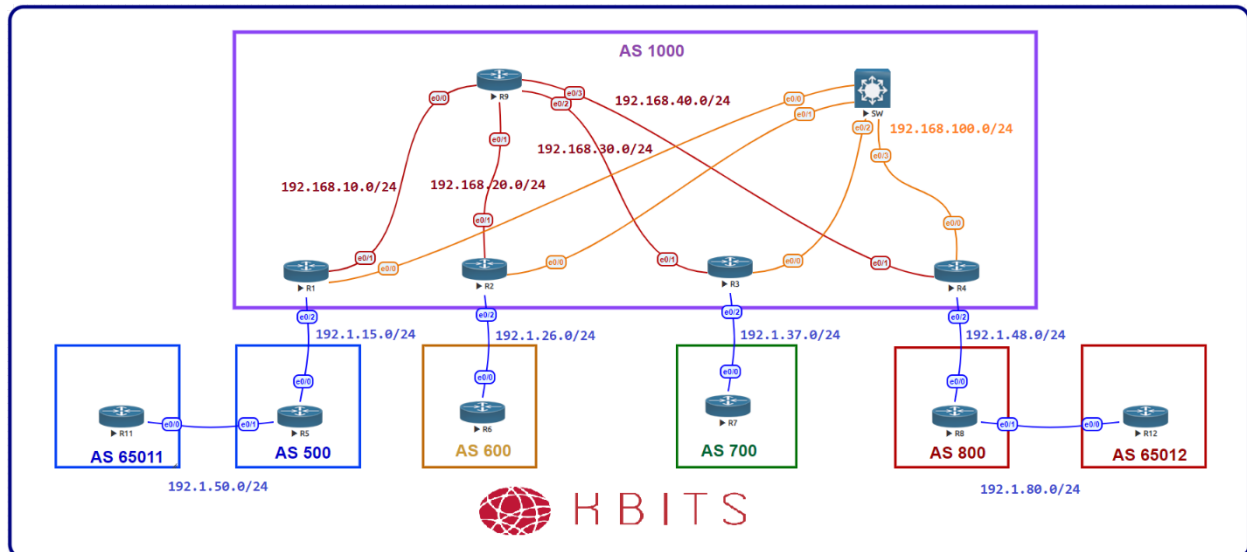
## Task 1

Configure R3 as the Route Reflector between R4 & R5. Make sure to change the next-hop to R3.

## R3

```
Router bgp 1000
Neighbor 192.168.35.5 route-reflector-client
Neighbor 192.168.35.5 next-hop-self all
Neighbor 192.168.34.4 route-reflector-client
Neighbor 192.168.34.4 next-hop-self all
```

# Lab 23 – Configuring BGP Route Reflection based on Dynamic Neighbors



## Interface IP Address Configuration

### R1

Interface	IP Address	Subnet Mask
Loopback 0	1.1.1.1	255.0.0.0
Loopback 10	172.16.1.1	255.255.255.255
E 0/0	192.168.100.1	255.255.255.0
E 0/1	192.168.10.1	255.255.255.0
E 0/2	192.1.15.1	255.255.255.0

### R2

Interface	IP Address	Subnet Mask
Loopback 0	2.2.2.2	255.0.0.0
Loopback 10	172.16.1.2	255.255.255.255
E 0/0	192.168.100.2	255.255.255.0
E 0/1	192.168.20.2	255.255.255.0
E 0/2	192.1.26.2	255.255.255.0



### R3

Interface	IP Address	Subnet Mask
Loopback 0	3.3.3.3	255.0.0.0
Loopback 10	172.16.1.3	255.255.255.255
E 0/0	192.168.100.3	255.255.255.0
E 0/1	192.168.30.3	255.255.255.0
E 0/2	192.1.37.3	255.255.255.0

### R4

Interface	IP Address	Subnet Mask
Loopback 0	4.4.4.4	255.0.0.0
Loopback 10	172.16.1.4	255.255.255.255
E 0/0	192.168.100.4	255.255.255.0
E 0/1	192.168.40.4	255.255.255.0
E 0/2	192.1.48.4	255.255.255.0

### R5

Interface	IP Address	Subnet Mask
Loopback 0	5.5.5.5	255.0.0.0
E 0/0	192.1.15.5	255.255.255.0
E 0/1	192.1.50.5	255.255.255.0

### R6

Interface	IP Address	Subnet Mask
Loopback 0	6.6.6.6	255.0.0.0
E 0/0	192.1.26.6	255.255.255.0

### R7

Interface	IP Address	Subnet Mask
Loopback 0	7.7.7.7	255.0.0.0
E 0/0	192.1.37.7	255.255.255.0

**R8**

Interface	IP Address	Subnet Mask
Loopback 0	8.8.8.8	255.0.0.0
E 0/0	192.1.48.8	255.255.255.0
E 0/2	192.1.80.8	255.255.255.0

**R9**

Interface	IP Address	Subnet Mask
Loopback 10	172.16.1.9	255.255.255.255
E 0/0	192.168.10.9	255.255.255.0
E 0/1	192.168.20.9	255.255.255.0
E 0/2	192.168.30.9	255.255.255.0
E 0/3	192.168.40.9	255.255.255.0

**R11**

Interface	IP Address	Subnet Mask
Loopback 0	11.11.11.11	255.0.0.0
E 0/0	192.1.50.11	255.255.255.0

**R12**

Interface	IP Address	Subnet Mask
Loopback 0	12.12.12.12	255.0.0.0
E 0/0	192.1.80.12	255.255.255.0

**Task 1**

Configure EIGRP 111 as the underlay IGP to route the Loopback 10 networks on the underlay networks.

<b>R1</b> Router eigrp 1000 network 192.168.100.0 network 192.168.10.0 network 172.16.1.0 0.0.0.255	<b>R2</b> Router eigrp 1000 network 192.168.100.0 network 192.168.20.0 network 172.16.1.0 0.0.0.0
<b>R3</b> Router eigrp 1000 network 192.168.100.0	<b>R4</b> Router eigrp 1000 network 192.168.100.0

network 192.168.30.0 network 172.16.1.0 0.0.0.0	network 192.168.40.0 network 172.16.1.0 0.0.0.0
<b>R9</b>  Router eigrp 1000 network 192.168.10.0 network 192.168.20.0 network 192.168.30.0 network 192.168.40.0 network 172.16.1.0 0.0.0.0	

## Task 2

Configuring iBGP between the ASBR (R1,R2,R3 & R4) and the RR (R9) based on Loopbacks. Configure R9 such that it accepts neighbor requests from any router from the 172.16.1.0/24 subnet. Authenticate the neighbor relationship with a password of **ccie12353**. Advertise the Loopback 0 networks on ASBRs in BGP.

<b>R1</b>  Router BGP 1000 Network 1.0.0.0 Neighbor 172.16.1.9 remote-as 1000 Neighbor 172.16.1.9 update-source Lo10 Neighbor 172.16.1.9 next-hop-self Neighbor 172.16.1.9 password ccie12353	<b>R2</b>  Router BGP 1000 Network 2.0.0.0 Neighbor 172.16.1.9 remote-as 1000 Neighbor 172.16.1.9 update-source Lo10 Neighbor 172.16.1.9 next-hop-self Neighbor 172.16.1.9 password ccie12353
<b>R3</b>  Router BGP 1000 Network 3.0.0.0 Neighbor 172.16.1.9 remote-as 1000 Neighbor 172.16.1.9 update-source Lo10 Neighbor 172.16.1.9 next-hop-self Neighbor 172.16.1.9 password ccie12353	<b>R4</b>  Router BGP 1000 Network 4.0.0.0 Neighbor 172.16.1.9 remote-as 1000 Neighbor 172.16.1.9 update-source Lo10 Neighbor 172.16.1.9 next-hop-self Neighbor 172.16.1.9 password ccie12353
<b>R9</b>  router bgp 1000 neighbor IBGP peer-group neighbor IBGP remote-as 1000 neighbor IBGP update-source loopback10 neighbor IBGP route-reflector-client neighbor IBGP password ccie12353 bgp listen range 172.16.1.0/24 peer-group IBGP	

### Task 3

Configuring eBGP neighbor relationship between AS 1000 and the connected ASs on the appropriate ASBRs. Advertise Loopback0 networks on R5, R6, R7 & R8.

#### R1

```
router bgp 1000
neighbor 192.1.15.5 remote-as 500
```

#### R2

```
router bgp 1000
neighbor 192.1.26.6 remote-as 600
```

#### R3

```
router bgp 1000
neighbor 192.1.37.7 remote-as 700
```

#### R4

```
router bgp 1000
neighbor 192.1.48.8 remote-as 800
```

#### R5

```
router bgp 500
network 5.0.0.0
neighbor 192.1.15.1 remote-as 1000
```

#### R6

```
router bgp 600
network 6.0.0.0
neighbor 192.1.26.2 remote-as 1000
```

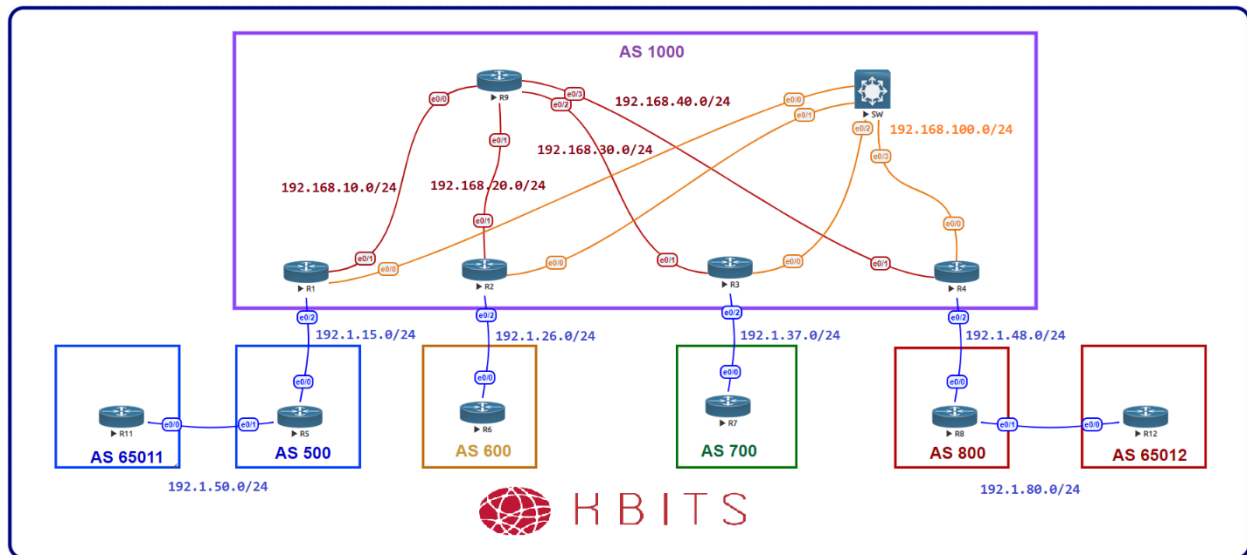
#### R7

```
router bgp 700
network 7.0.0.0
neighbor 192.1.37.3 remote-as 1000
```

#### R8

```
router bgp 800
network 8.0.0.0
neighbor 192.1.48.4 remote-as 1000
```

# Lab 24 – Working with Private AS Numbers



## Task 1

Configure a relationship between the Customer (R12) and AS 800. The Customer should use AS 65012 as the AS #. Advertise the Loopback 0 network on R12.

### R8

```
router bgp 800
neighbor 192.1.80.12 remote-as 65012
```

### R12

```
router bgp 65012
neighbor 192.1.80.8 remote-as 800
network 12.0.0.0
```

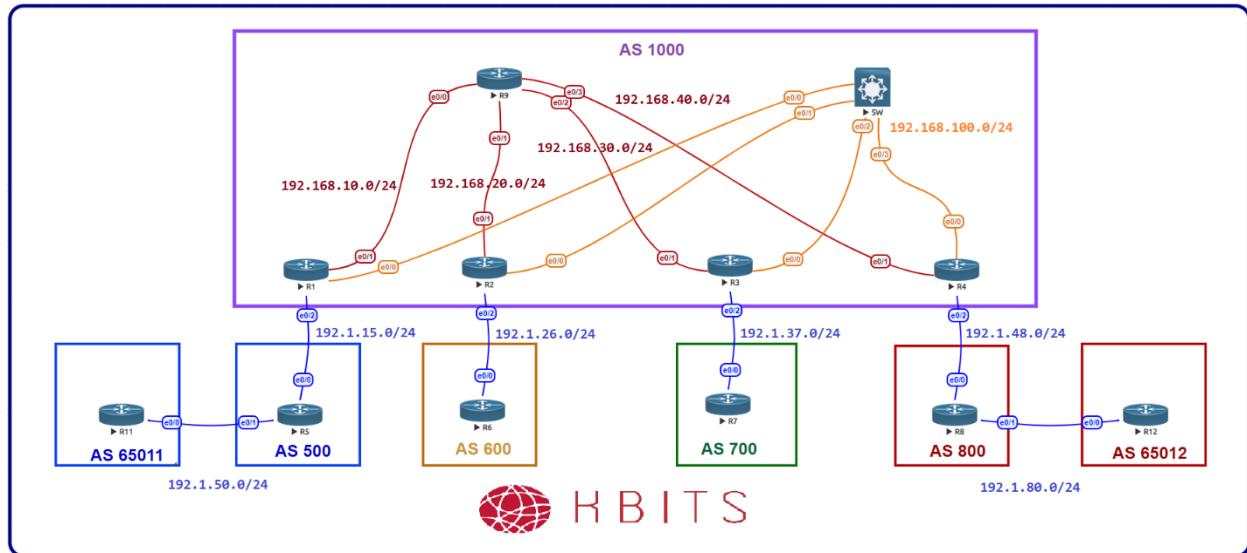
## Task 2

Configure R8 such that it removes the Private AS # from the AS Path before propagating the route towards AS 1000

### R8

```
router bgp 800
neighbor 192.1.48.4 remove-private-as
```

# Lab 25 – Configuring the Local-AS Command



## Task 1

Configure a relationship between the Customer (R11) and AS 500. The Customer should use AS 65011 as the AS #. Advertise the Loopback 0 network on R11.

### R5

```
router bgp 500
neighbor 192.1.50.11 remote-as 65011
```

### R11

```
router bgp 65011
neighbor 192.1.50.5 remote-as 500
network 11.0.0.0
```

## Task 2

Configure R5 such that it removes the Private AS # from the AS Path before propagating the route towards AS 1000

### R5

```
router bgp 500
neighbor 192.1.15.1 remove-private-as
```

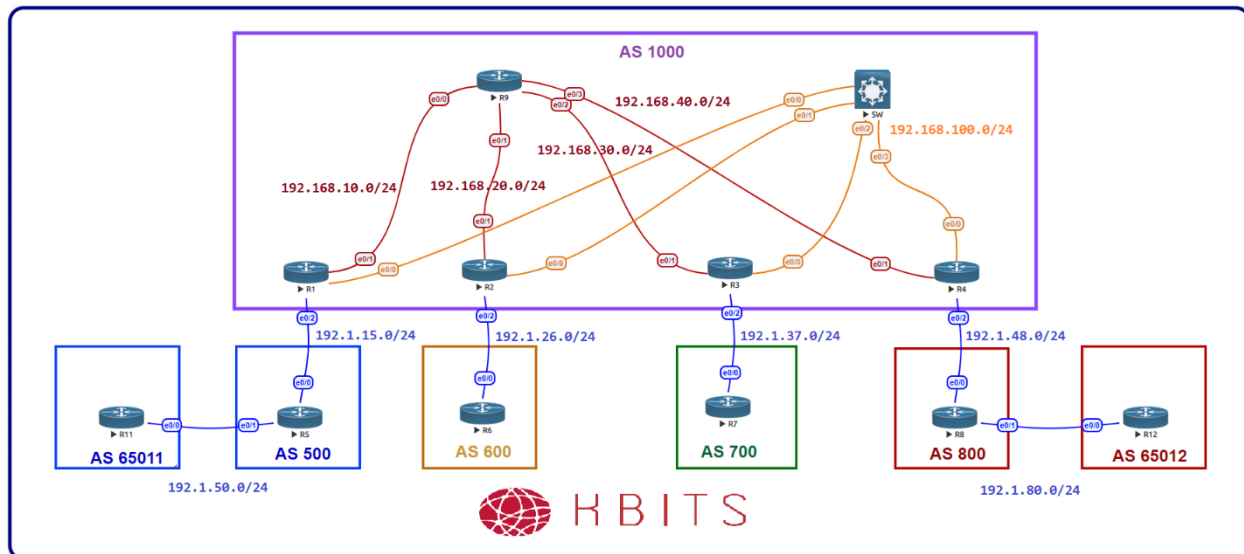
### **Task 3**

R11 acquires and configures a new AS #. It is a Public AS# 110. R5 will change the neighbor relationship in a maintenance window after 5 days. In the meanwhile R11 needs to change the AS # to the new to establish a new neighbor relationship with a new SP. Allow R11 to establish both neighbor relationships.

#### **R11**

```
no router bgp 65011
router bgp 110
network 1.11.1.0 mask 255.255.255.0
neighbor 192.1.50.5 remote-as 500
neighbor 192.1.50.5 local-as 65011
```

# Lab 26 – Configuring BFD for BGP



## Task 1

Configure BFD using a send and receive interval of 350 ms. A neighbor is deemed dead if 3 hellos are missed. Configure it for the following eBGP neighbor relationships:

- R1 - R5
- R2 - R6
- R3 - R7
- R4 - R8

### R1

```
Interface E 0/2
bfd interval 350 min_rx 350 multiplier 3
!
router bgp 1000
neighbor 192.1.15.5 fall-over bfd
```

### R5

```
Interface E 0/0
bfd interval 350 min_rx 350 multiplier 3
!
router bgp 500
neighbor 192.1.15.1 fall-over bfd
```

### R2



```
Interface E 0/2
bfd interval 350 min_rx 350 multiplier 3
!
router bgp 1000
neighbor 192.1.26.6 fall-over bfd
```

### **R6**

```
Interface E 0/0
bfd interval 350 min_rx 350 multiplier 3
!
router bgp 600
neighbor 192.1.26.2 fall-over bfd
```

### **R3**

```
Interface E 0/2
bfd interval 350 min_rx 350 multiplier 3
!
router bgp 1000
neighbor 192.1.37.7 fall-over bfd
```

### **R7**

```
Interface E 0/0
bfd interval 350 min_rx 350 multiplier 3
!
router bgp 700
neighbor 192.1.37.3 fall-over bfd
```

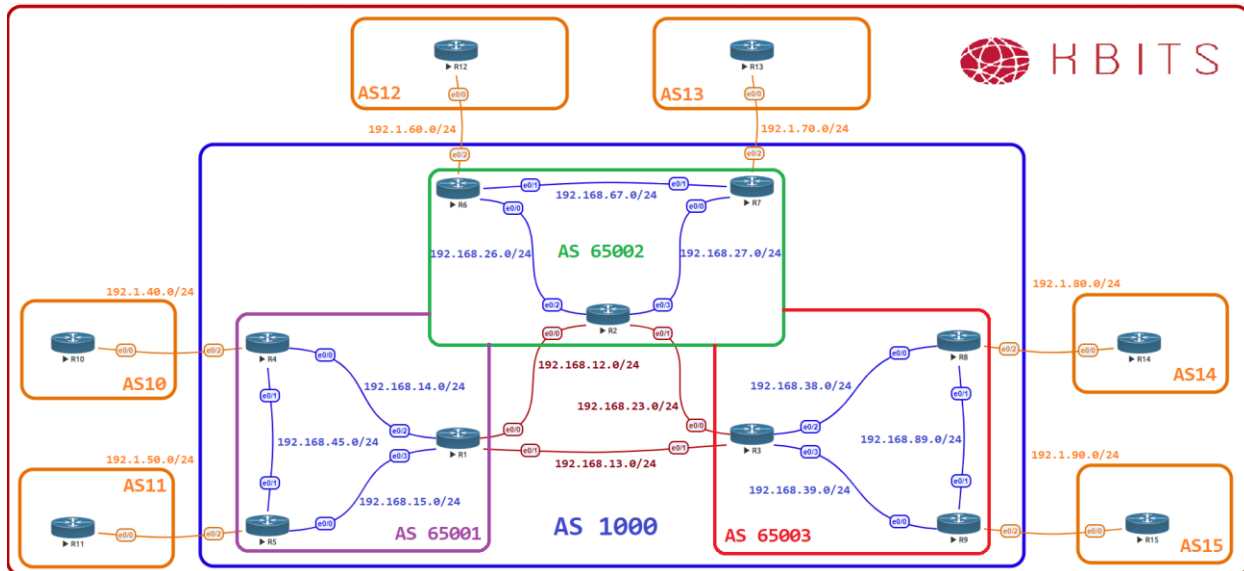
### **R4**

```
Interface E 0/2
bfd interval 350 min_rx 350 multiplier 3
!
router bgp 1000
neighbor 192.1.48.8 fall-over bfd
```

### **R8**

```
Interface E 0/0
bfd interval 350 min_rx 350 multiplier 3
!
router bgp 800
neighbor 192.1.48.4 fall-over bfd
```

# Lab 27 – Configuring BGP Confederations



## Interface IP Address Configuration

### R1

Interface	IP Address	Subnet Mask
Loopback 0	1.1.1.1	255.0.0.0
Loopback 10	172.16.1.1	255.255.255.255
E 0/0	192.168.12.1	255.255.255.0
E 0/1	192.168.13.1	255.255.255.0
E 0/2	192.168.14.1	255.255.255.0
E 0/3	192.168.15.1	255.255.255.0

### R2

Interface	IP Address	Subnet Mask
Loopback 0	2.2.2.2	255.0.0.0
Loopback 10	172.16.1.2	255.255.255.255
E 0/0	192.168.12.2	255.255.255.0
E 0/1	192.168.23.2	255.255.255.0
E 0/2	192.168.26.2	255.255.255.0
E 0/3	192.168.27.2	255.255.255.0

### R3

Interface	IP Address	Subnet Mask
Loopback 0	3.3.3.3	255.0.0.0
Loopback 10	172.16.1.3	255.255.255.255
E 0/0	192.168.23.3	255.255.255.0
E 0/1	192.168.13.3	255.255.255.0
E 0/2	192.168.38.3	255.255.255.0
E 0/3	192.168.39.3	255.255.255.0

### R4

Interface	IP Address	Subnet Mask
Loopback 0	4.4.4.4	255.0.0.0
Loopback 10	172.16.1.4	255.255.255.255
E 0/0	192.168.14.4	255.255.255.0
E 0/1	192.168.45.4	255.255.255.0
E 0/2	192.1.40.4	255.255.255.0

### R5

Interface	IP Address	Subnet Mask
Loopback 0	5.5.5.5	255.0.0.0
Loopback 10	172.16.1.5	255.255.255.255
E 0/0	192.168.15.5	255.255.255.0
E 0/1	192.168.45.5	255.255.255.0
E 0/2	192.1.50.5	255.255.255.0

### R6

Interface	IP Address	Subnet Mask
Loopback 0	6.6.6.6	255.0.0.0
Loopback 10	172.16.1.6	255.255.255.255
E 0/0	192.168.26.6	255.255.255.0
E 0/1	192.168.67.6	255.255.255.0
E 0/2	192.1.60.6	255.255.255.0

**R7**

<b>Interface</b>	<b>IP Address</b>	<b>Subnet Mask</b>
Loopback 0	7.7.7.7	255.0.0.0
Loopback 10	172.16.1.7	255.255.255.255
E 0/0	192.168.27.7	255.255.255.0
E 0/1	192.168.67.7	255.255.255.0
E 0/2	192.1.70.7	255.255.255.0

**R8**

<b>Interface</b>	<b>IP Address</b>	<b>Subnet Mask</b>
Loopback 0	8.8.8.8	255.0.0.0
Loopback 10	172.16.1.8	255.255.255.255
E 0/0	192.168.38.8	255.255.255.0
E 0/1	192.168.89.8	255.255.255.0
E 0/2	192.1.80.8	255.255.255.0

**R9**

<b>Interface</b>	<b>IP Address</b>	<b>Subnet Mask</b>
Loopback 0	9.9.9.9	255.0.0.0
Loopback 10	172.16.1.9	255.255.255.255
E 0/0	192.168.39.9	255.255.255.0
E 0/1	192.168.89.9	255.255.255.0
E 0/2	192.1.90.9	255.255.255.0

**R10**

<b>Interface</b>	<b>IP Address</b>	<b>Subnet Mask</b>
Loopback 0	100.100.100.100	255.0.0.0
E 0/0	192.1.40.10	255.255.255.0

**R11**

<b>Interface</b>	<b>IP Address</b>	<b>Subnet Mask</b>
Loopback 0	111.111.111.111	255.0.0.0
E 0/0	192.1.50.11	255.255.255.0

## R12

Interface	IP Address	Subnet Mask
Loopback 0	112.112.112.112	255.0.0.0
E 0/0	192.1.60.12	255.255.255.0

## R13

Interface	IP Address	Subnet Mask
Loopback 0	113.113.113.113	255.0.0.0
E 0/0	192.1.70.13	255.255.255.0

## R14

Interface	IP Address	Subnet Mask
Loopback 0	114.114.114.114	255.0.0.0
E 0/0	192.1.80.14	255.255.255.0

## R15

Interface	IP Address	Subnet Mask
Loopback 0	115.115.115.115	255.0.0.0
E 0/0	192.1.90.15	255.255.255.0

### Task 1

Configure the underlay IGP as EIGRP in AS 1000 between R1, R2 & R3. These routers represent their respective Sub-AS's.

<b>R1</b>  router eigrp 1000 network 192.168.12.0 network 192.168.13.0 network 172.16.1.0 0.0.0.255	<b>R2</b>  router eigrp 1000 network 192.168.12.0 network 192.168.23.0 network 172.16.1.0 0.0.0.255
<b>R3</b>  router eigrp 1000 network 192.168.13.0 network 192.168.23.0 network 172.16.1.0 0.0.0.255	

### Task 2

Configure the underlay IGP as EIGRP in Sub-AS 65001 between R1, R4 & R5. Advertise the links with the Sub-AS and the Loopback 10 networks in EIGRP.

<b>R1</b>  router eigrp 65001 network 192.168.14.0 network 192.168.15.0 network 172.16.1.0 0.0.0.255	<b>R4</b>  router eigrp 65001 network 192.168.14.0 network 192.168.45.0 network 172.16.1.0 0.0.0.255
<b>R5</b>  router eigrp 65001 network 192.168.15.0 network 192.168.45.0 network 172.16.1.0 0.0.0.255	

### Task 3

Configure the underlay IGP as EIGRP in Sub-AS 65002 between R2, R6 & R7. Advertise the links with the Sub-AS and the Loopback 10 networks in EIGRP.

<b>R1</b>  router eigrp 65002 network 192.168.26.0 network 192.168.27.0 network 172.16.1.0 0.0.0.255	<b>R6</b>  router eigrp 65002 network 192.168.26.0 network 192.168.67.0 network 172.16.1.0 0.0.0.255
<b>R7</b>  router eigrp 65002 network 192.168.27.0 network 192.168.67.0 network 172.16.1.0 0.0.0.255	

#### Task 4

Configure the underlay IGP as EIGRP in Sub-AS 65003 between R3, R8 & R9. Advertise the links with the Sub-AS and the Loopback 10 networks in EIGRP.

<b>R3</b>  router eigrp 65003 network 192.168.38.0 network 192.168.39.0 network 172.16.1.0 0.0.0.255	<b>R8</b>  router eigrp 65003 network 192.168.38.0 network 192.168.89.0 network 172.16.1.0 0.0.0.255
<b>R9</b>  router eigrp 65003 network 192.168.39.0 network 192.168.89.0 network 172.16.1.0 0.0.0.255	

#### Task 5

Configure AS 65001 with iBGP. Configure R1 as the RR. Set the relationship based on Loopback10. The Confederation Identifier is 1000. R1 is peering up only with 65002 in its confederation.

<b>R1</b>  router bgp 65001 bgp confederation identifier 1000 bgp confederation peer 65002 network 1.0.0.0 neighbor IBGP peer-group neighbor IBGP remote-as 65001 neighbor IBGP update-source Loopback10 neighbor IBGP next-hop-self neighbor IBGP route-reflector-client neighbor 172.16.1.4 peer-group IBGP neighbor 172.16.1.5 peer-group IBGP	
<b>R4</b>  router bgp 65001 bgp confederation identifier 1000 network 4.0.0.0 neighbor 172.16.1.1 remote-as 65001 neighbor 172.16.1.1 update-source Lo10 neighbor 172.16.1.1 next-hop-self	<b>R5</b>  router bgp 65001 bgp confederation identifier 1000 network 5.0.0.0 neighbor 172.16.1.1 remote-as 65001 neighbor 172.16.1.1 update-source Lo10 neighbor 172.16.1.1 next-hop-self

### Task 6

Configure AS 65002 with iBGP. Configure R2 as the RR. Set the relationship based on Loopback10. The Confederation Identifier is 1000. R2 is peering up with 65001 & 65003 in its confederation.

#### R2

```
router bgp 65002
  bgp confederation identifier 1000
  bgp confederation peer 65001 65003
  network 2.0.0.0
  neighbor IBGP peer-group
  neighbor IBGP remote-as 65002
  neighbor IBGP update-source Loopback10
  neighbor IBGP next-hop-self
  neighbor IBGP route-reflector-client
  neighbor 172.16.1.6 peer-group IBGP
  neighbor 172.16.1.7 peer-group IBGP
```

#### R6

```
router bgp 65002
  bgp confederation identifier 1000
  network 6.0.0.0
  neighbor 172.16.1.2 remote-as 65002
  neighbor 172.16.1.2 update-source Lo10
  neighbor 172.16.1.2 next-hop-self
```

#### R7

```
router bgp 65002
  bgp confederation identifier 1000
  network 7.0.0.0
  neighbor 172.16.1.2 remote-as 65002
  neighbor 172.16.1.2 update-source Lo10
  neighbor 172.16.1.2 next-hop-self
```



### Task 7

Configure AS 65003 with iBGP. Configure R3 as the RR. Set the relationship based on Loopback10. The Confederation Identifier is 1000. R3 is peering up only with 65002 in its confederation.

#### R3

```
router bgp 65003
  bgp confederation identifier 1000
  bgp confederation peer 65002
  network 3.0.0.0
  neighbor IBGP peer-group
  neighbor IBGP remote-as 65003
  neighbor IBGP update-source Loopback10
  neighbor IBGP next-hop-self
  neighbor IBGP route-reflector-client
  neighbor 172.16.1.8 peer-group IBGP
  neighbor 172.16.1.9 peer-group IBGP
```

#### R8

```
router bgp 65003
  bgp confederation identifier 1000
  network 8.0.0.0
  neighbor 172.16.1.3 remote-as 65003
  neighbor 172.16.1.3 update-source Lo10
  neighbor 172.16.1.3 next-hop-self
```

#### R9

```
router bgp 65003
  bgp confederation identifier 1000
  network 9.0.0.0
  neighbor 172.16.1.3 remote-as 65003
  neighbor 172.16.1.3 update-source Lo10
  neighbor 172.16.1.3 next-hop-self
```

**Task 8**

Configure eBGP neighbor relationships with Remote-AS's (10,11,12,13 & 14). Use the appropriate ASBR to configure the relationship. Have the Remote AS's advertise the Loopback0 interface network.

<b>R4</b> router bgp 65001 neighbor 192.1.40.10 remote-as 10	<b>R10</b> router bgp 10 network 100.0.0.0 neighbor 192.1.40.4 remote-as 1000
<b>R5</b> router bgp 65001 neighbor 192.1.50.11 remote-as 11	<b>R11</b> router bgp 11 network 111.0.0.0 neighbor 192.1.50.5 remote-as 1000
<b>R6</b> router bgp 65002 neighbor 192.1.60.12 remote-as 12	<b>R12</b> router bgp 12 network 112.0.0.0 neighbor 192.1.60.6 remote-as 1000
<b>R7</b> router bgp 65002 neighbor 192.1.70.13 remote-as 13	<b>R13</b> router bgp 13 network 113.0.0.0 neighbor 192.1.70.7 remote-as 1000
<b>R8</b> router bgp 65003 neighbor 192.1.80.14 remote-as 14	<b>R14</b> router bgp 14 network 114.0.0.0 neighbor 192.1.80.8 remote-as 1000
<b>R9</b> router bgp 65003 neighbor 192.1.90.15 remote-as 15	<b>R15</b> router bgp 15 network 115.0.0.0 neighbor 192.1.90.9 remote-as 1000

### Task 9

Configure eBGP neighbor relationships between the Confederation Peers. (R1-R2) & (R2-R3). These are eBGP neighbor relationships that are on Loopbacks. Make sure to allow the ebgp-multihop.

#### R1

```
router bgp 65001
neighbor 172.16.1.2 remote-as 65002
neighbor 172.16.1.2 update-source Loopback10
neighbor 172.16.1.2 next-hop-self
neighbor 172.16.1.2 ebgp-multihop
```

#### R2

```
router bgp 65002
neighbor 172.16.1.1 remote-as 65001
neighbor 172.16.1.1 update-source Loopback10
neighbor 172.16.1.1 next-hop-self
neighbor 172.16.1.1 ebgp-multihop
neighbor 172.16.1.3 remote-as 65003
neighbor 172.16.1.3 update-source Loopback10
neighbor 172.16.1.3 next-hop-self
neighbor 172.16.1.3 ebgp-multihop
```

#### R3

```
router bgp 65003
neighbor 172.16.1.2 remote-as 65002
neighbor 172.16.1.2 update-source Loopback10
neighbor 172.16.1.2 next-hop-self
neighbor 172.16.1.2 ebgp-multihop
```

### Verification:

Use Ping to verify end-to-end reachability between AS's 10,11,12,13 & 14 via AS 1000.

# Configuring IPv6

**Authored By:**

**Khawar Butt**

CCIE # 12353

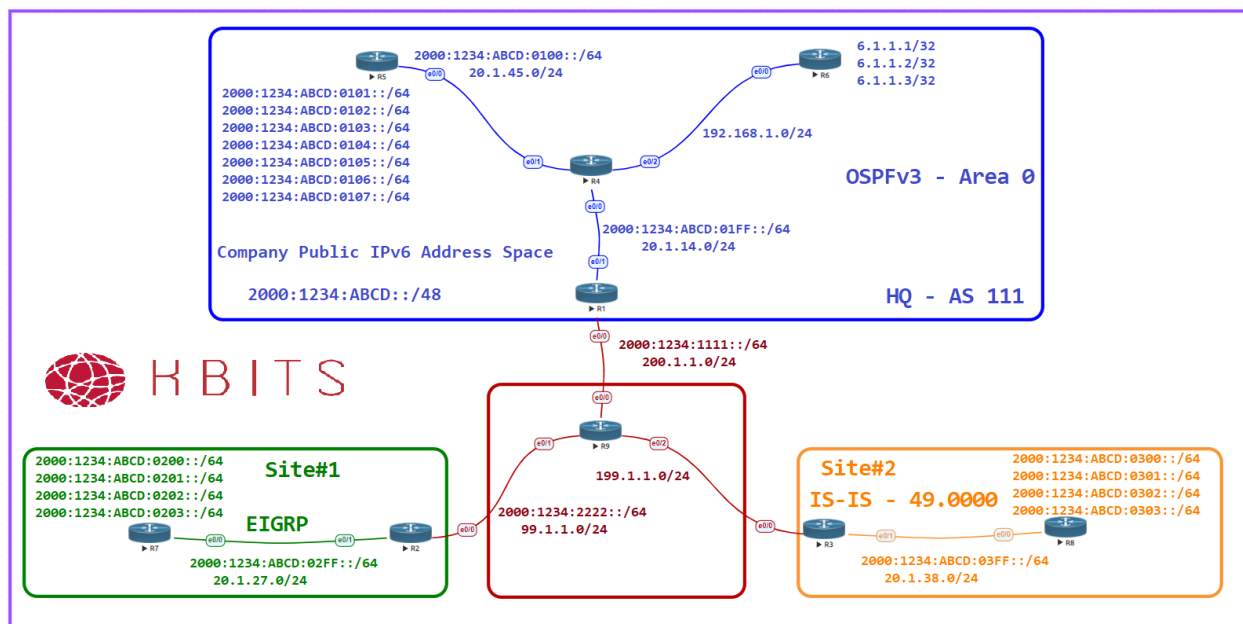
Hepta CCIE#12353

CCDE # 20110020

**Configuring IPv6**



# Lab 1 – Configuring IPv6 Addressing



## Task 1

Configure Headquarters with IPv6 addressing based on the Network Diagram. The Network between R4 & R6 will remain as IPv4 only. Configure the rest of the routers with IPv6 addressing based on the Network Diagram. Configure a default route on the Edge Router (R1) towards the ISP.

### R1

```

ipv6 unicast-routing
!
Interface E 0/0
ipv6 address 2000:1234:1111::1/64
no shut
!
Interface E 0/1
ipv6 address 2000:1234:ABCD:01FF::1/64
no shut
!
ipv6 route ::/0 2000:1234:1111::9

```

### R4

```

ipv6 unicast-routing
!

```

```
Interface E 0/0
ipv6 address 2000:1234:ABCD:01FF::4/64
no shut
!
Interface E 0/1
ipv6 address 2000:1234:ABCD:0100::4/64
no shut
```

## **R5**

```
ipv6 unicast-routing
!
Interface E 0/0
ipv6 address 2000:1234:ABCD:0100::5/64
no shut
!
Interface Loopback1
ipv6 address 2000:1234:ABCD:0101::5/64
!
Interface Loopback2
ipv6 address 2000:1234:ABCD:0102::5/64
!
Interface Loopback3
ipv6 address 2000:1234:ABCD:0103::5/64
!
Interface Loopback4
ipv6 address 2000:1234:ABCD:0104::5/64
!
Interface Loopback5
ipv6 address 2000:1234:ABCD:0105::5/64
!
Interface Loopback6
ipv6 address 2000:1234:ABCD:0106::5/64
!
Interface Loopback7
ipv6 address 2000:1234:ABCD:0107::5/64
```

## **Task 2**

Configure **Site#1** with IPv6 addressing based on the Network Diagram.  
Configure a default route on the Edge Router (R2) towards the ISP.

## **R2**

```
Interface E 0/0
ipv6 address 2000:1234:2222::2/64
no shut
```

```
!  
Interface E 0/1  
  ipv6 address 2000:1234:ABCD:02FF::2/64  
  no shut  
!  
ipv6 route ::/0 2000:1234:2222::9
```

### **R7**

```
ipv6 unicast-routing  
!  
Interface E 0/0  
  ipv6 address 2000:1234:ABCD:02FF::7/64  
  no shut  
!  
Interface Loopback1  
  ipv6 address 2000:1234:ABCD:0200::7/64  
!  
Interface Loopback2  
  ipv6 address 2000:1234:ABCD:0201::7/64  
!  
Interface Loopback3  
  ipv6 address 2000:1234:ABCD:0202::7/64  
!  
Interface Loopback4  
  ipv6 address 2000:1234:ABCD:0203::7/64
```

### **Task 3**

Configure **Site#2** with IPv6 addressing based on the Network Diagram.

### **R3**

```
ipv6 unicast-routing  
!  
Interface E 0/1  
  ipv6 address 2000:1234:ABCD:03FF::3/64  
  no shut
```

### **R8**

```
ipv6 unicast-routing  
!  
Interface E 0/0  
  ipv6 address 2000:1234:ABCD:03FF::8/64  
  no shut  
!  
Interface Loopback1
```

```
ipv6 address 2000:1234:ABCD:0300::8/64
!  
Interface Loopback2  
ipv6 address 2000:1234:ABCD:0301::8/64
!  
Interface Loopback3  
ipv6 address 2000:1234:ABCD:0302::8/64
!  
Interface Loopback4  
ipv6 address 2000:1234:ABCD:0303::8/64
```

#### **Task 4**

Configure IPv4 IP Addresses based on the network diagram. Configure Static Routing to provide full reachability for IPv4 networks. You are allowed to use static routes.

#### **R1**

```
Interface E 0/0  
Ip address 200.1.1.1 255.255.255.0  
No shut  
!  
Interface E 0/1  
Ip address 20.1.14.1 255.255.255.0  
No shut  
!  
Ip route 0.0.0.0 0.0.0.0 200.1.1.9  
Ip route 20.1.45.0 255.255.255.0 20.1.14.4  
Ip route 6.1.1.0 255.255.255.0 20.1.14.4
```

#### **R2**

```
Interface E 0/0  
Ip address 99.1.1.2 255.255.255.0  
No shut  
!  
Interface E 0/1  
Ip address 20.1.27.2 255.255.255.0  
No shut  
!  
Ip route 0.0.0.0 0.0.0.0 99.1.1.9
```

#### **R3**

```
Interface E 0/0  
Ip address 199.1.1.3 255.255.255.0  
No shut
```



```
!  
Interface E 0/1  
Ip address 20.1.38.3 255.255.255.0  
No shut  
!  
Ip route 0.0.0.0 0.0.0.0 199.1.1.9
```

#### **R4**

```
Interface E 0/0  
Ip address 20.1.14.4 255.255.255.0  
No shut  
!  
Interface E 0/1  
Ip address 20.1.45.4 255.255.255.0  
No shut  
!  
Interface E 0/2  
Ip address 192.168.1.4 255.255.255.0  
No shut  
!  
Ip route 0.0.0.0 0.0.0.0 20.1.14.1  
Ip route 6.1.1.0 255.255.255.0 192.168.1.6
```

#### **R5**

```
Interface E 0/0  
Ip address 20.1.45.5 255.255.255.0  
No shut  
!  
Ip route 0.0.0.0 0.0.0.0 20.1.45.4
```

#### **R6**

```
Interface E 0/0  
Ip address 192.168.1.6 255.255.255.0  
No shut  
!  
Interface Loo1  
Ip address 6.1.1.1 255.255.255.255  
!  
Interface Loo2  
Ip address 6.1.1.2 255.255.255.255  
!  
Interface Loo3  
Ip address 6.1.1.3 255.255.255.255  
!  
Ip route 0.0.0.0 0.0.0.0 192.168.1.4
```

```
!  
Line vty 0 4  
Password cisco  
Login  
Transport input all
```

### **R7**

```
Interface E 0/0  
Ip address 20.1.27.7 255.255.255.0  
No shut  
!  
Ip route 0.0.0.0 0.0.0.0 20.1.27.2
```

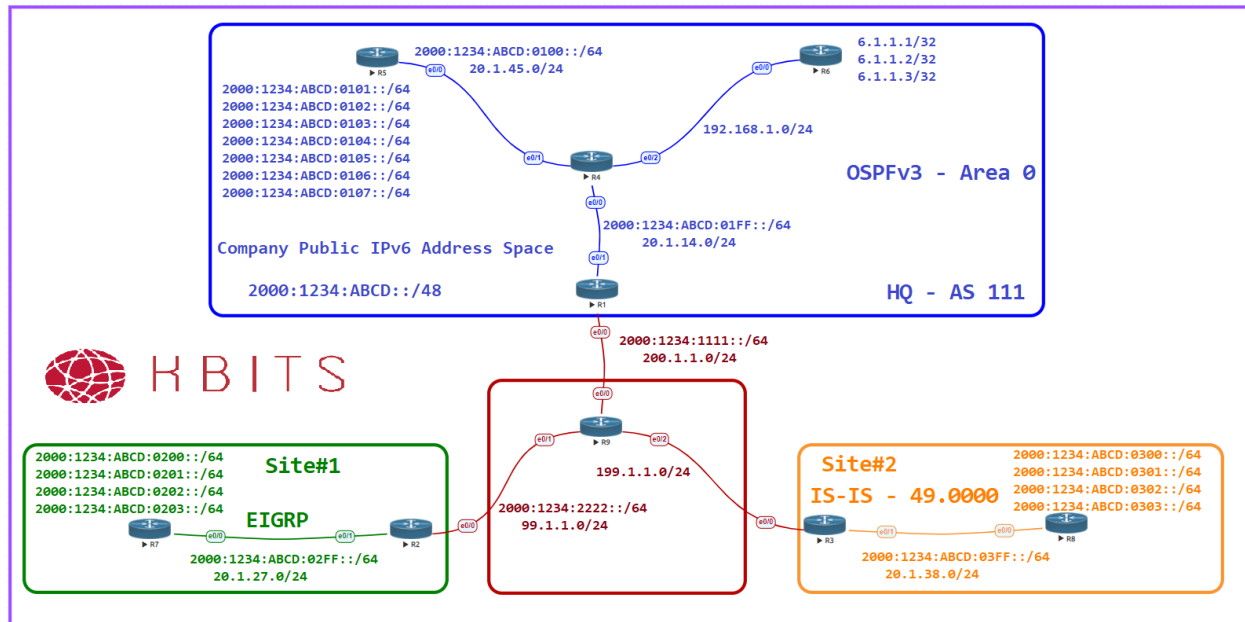
### **R8**

```
Interface E 0/0  
Ip address 20.1.38.8 255.255.255.0  
No shut  
!  
Ip route 0.0.0.0 0.0.0.0 20.1.38.3
```

### **R9**

```
Interface E 0/0  
Ip address 200.1.1.9 255.255.255.0  
No shut  
!  
Interface E 0/1  
Ip address 99.1.1.9 255.255.255.0  
No shut  
!  
Interface E 0/2  
Ip address 199.1.1.9 255.255.255.0  
No shut  
!  
Ip route 20.1.14.0 255.255.255.0 200.1.1.1  
Ip route 20.1.45.0 255.255.255.0 200.1.1.1  
Ip route 6.1.1.0 255.255.255.0 200.1.1.1  
Ip route 20.1.27.0 255.255.255.0 99.1.1.2  
Ip route 20.1.38.0 255.255.255.0 199.1.1.3
```

# Lab 2 – Configuring OSPFv3



## Task 1

Configure Headquarters with OSPFv3 within the HQ Site. Use X.X.X.X. as the router-id. (X stands for the Router #). Enable all the IPv6 addresses within the HQ site in OSPF. Have R1 inject a default route towards R4. The loopback interfaces should appear in the routing table using the interface mask.

### R1

```

ipv6 router ospf 1
router-id 1.1.1.1
default-information originate always
!
Interface E 0/1
ipv6 ospf 1 area 0

```

### R4

```

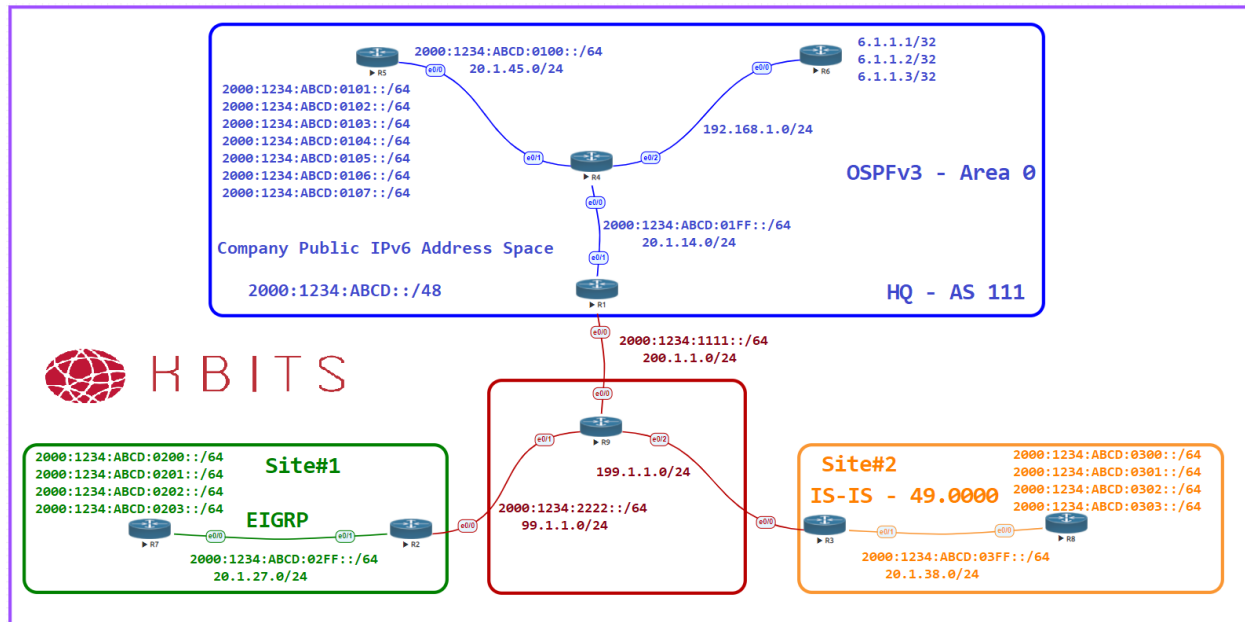
ipv6 router ospf 1
router-id 4.4.4.4
!
Interface E 0/0
ipv6 ospf 1 area 0
!
Interface E 0/1
ipv6 ospf 1 area 0

```

## R5

```
ipv6 router ospf 1
  router-id 5.5.5.5
!
Interface E 0/0
  ipv6 ospf 1 area 0
!
Interface Loopback 1
  ipv6 ospf 1 area 0
  ipv6 ospf network point-to-point
!
Interface Loopback 2
  ipv6 ospf 1 area 0
  ipv6 ospf network point-to-point
!
Interface Loopback 3
  ipv6 ospf 1 area 0
  ipv6 ospf network point-to-point
!
Interface Loopback 4
  ipv6 ospf 1 area 0
  ipv6 ospf network point-to-point
!
Interface Loopback 5
  ipv6 ospf 1 area 0
  ipv6 ospf network point-to-point
!
Interface Loopback 6
  ipv6 ospf 1 area 0
  ipv6 ospf network point-to-point
!
Interface Loopback 7
  ipv6 ospf 1 area 0
  ipv6 ospf network point-to-point
```

# Lab 3 – Configuring EIGRP for IPv6



## Task 1

Configure EIGRP 222 within Site#1. Use X.X.X.X. as the router-id. (X stands for the Router #). Enable all the IPv6 addresses within Site#1 in EIGRP. Configure a default route on R7 towards R2.

### R2

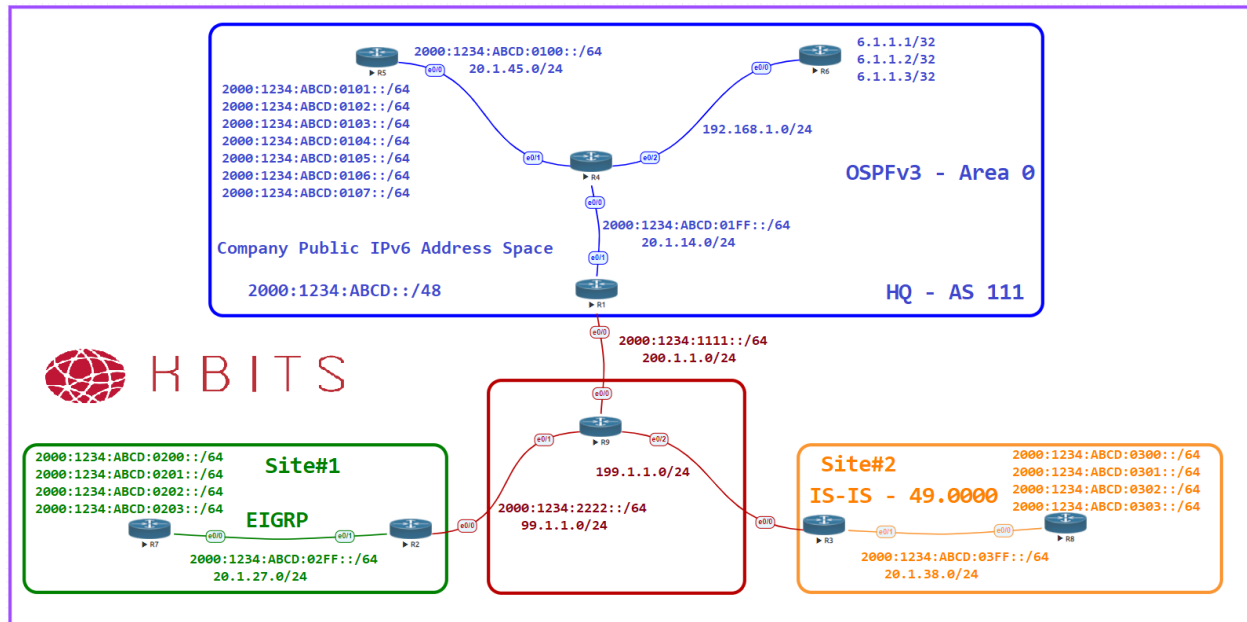
```
ipv6 router eigrp 222
router-id 2.2.2.2
!
Interface E 0/1
ipv6 eigrp 222
```

### R7

```
ipv6 router eigrp 222
router-id 7.7.7.7
!
Interface E 0/0
ipv6 eigrp 222
!
Interface Loopback 1
ipv6 eigrp 222
!
Interface Loopback 2
ipv6 eigrp 222
```

```
!  
Interface Loopback 3  
  ipv6 eigrp 222  
!  
Interface Loopback 4  
  ipv6 eigrp 222  
!  
Ipv6 route ::/0 2000:1234:ABCD:02FF::2
```

# Lab 4 – Configuring IS-IS for IPv6



## Task 1

Configure IS-IS within Site#1 based on the diagram. Use XXXX.XXXX.XXXX. as the System-id. (X stands for the Router #). Enable all the IPv6 addresses within Site#1 in IS-IS. Configure the Routers as Level-2 Routers with a metric-style of wide. Configure a default route on R7 towards R2.

### R3

```
router isis
net 49.0000.3333.3333.3333.00
is-type level-2-only
metric-style wide
!
address-family ipv6
multi-topology
!
Interface E 0/1
Ipv6 router isis
```

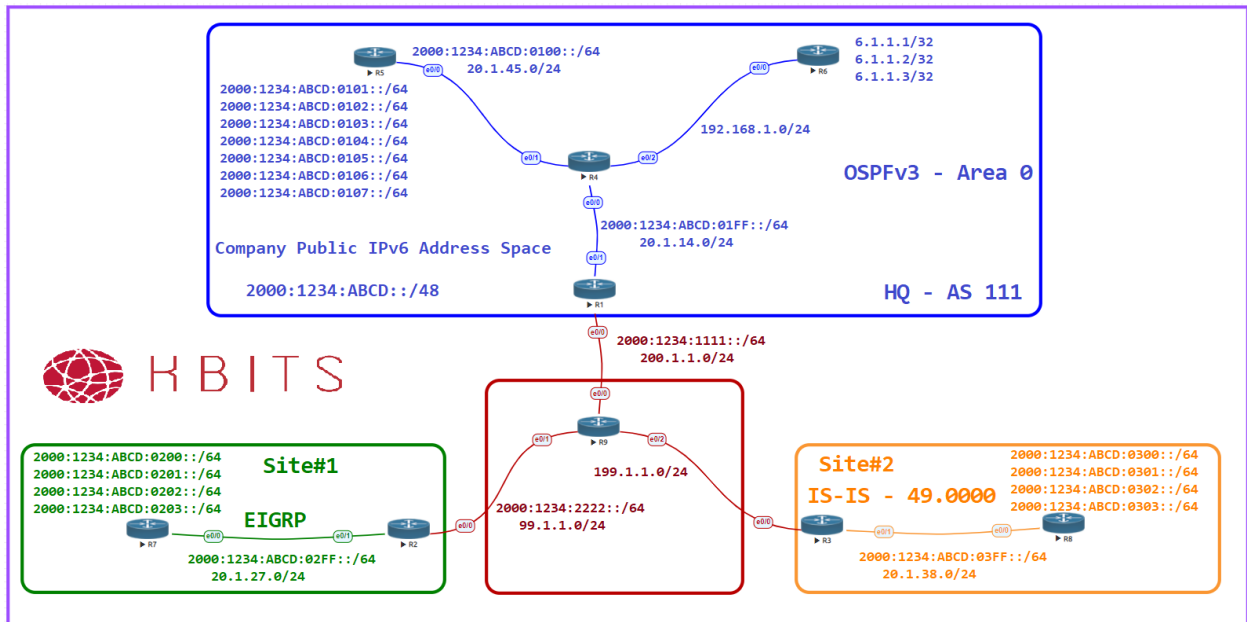
### R8

```
router isis
net 49.0000.8888.8888.8888.00
is-type level-2-only
metric-style wide
```

```
!  
address-family ipv6  
multi-topology  
!  
Interface E 0/0  
Ipv6 router isis  
!  
Interface Loopback 1  
Ipv6 router isis  
!  
Interface Loopback 2  
Ipv6 router isis  
!  
Interface Loopback 3  
Ipv6 router isis  
!  
Interface Loopback 4  
Ipv6 router isis  
!  
Ipv6 route ::/0 2000:1234:ABCD:03FF::3
```



# Lab 5 – Configuring BGP for IPv6



## Task 1

Configure BGP between R1 & R9. Configure R1 in AS 111. Redistribute the internal networks to BGP and vice versa.

### R1

```
router bgp 111
neighbor 2000:1234:1111::9 remote-as 1000
address-family ipv6
neighbor 2000:1234:1111::9 activate
redistribute ospf 1
!
ipv6 router ospf 1
redistribute bgp 111
```

### R9

```
router bgp 1000
neighbor 2000:1234:1111::1 remote-as 111
address-family ipv6
neighbor 2000:1234:1111::1 activate
```

## Task 2

Configure BGP between R2 & R9. Configure R2 in AS 222. Redistribute the internal networks to BGP and vice versa.

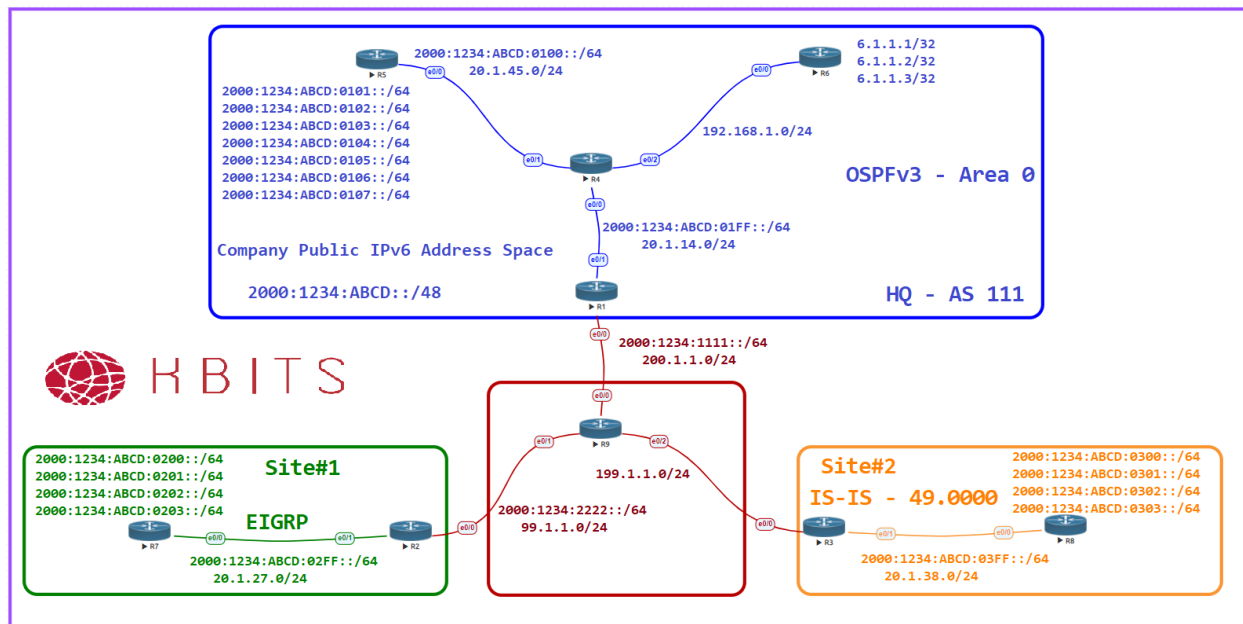
### R2

```
router bgp 111
neighbor 2000:1234:2222::9 remote-as 1000
address-family ipv6
neighbor 2000:1234:2222::9 activate
redistribute eigrp 222
!
ipv6 router eigrp 222
redistribute bgp 222 metric 10 10 10 10 10
```

### R9

```
router bgp 1000
neighbor 2000:1234:2222::2 remote-as 222
address-family ipv6
neighbor 2000:1234:2222::2 activate
```

# Lab 6 – Configuring IPv6IP Tunneling



## Task 1

Configure a IPv6IP tunnel to connect R1 to R3. Use the 2000:1234:ABCD:01FE::/64 as the Tunnel Network. Enable the Tunnel Interface in OSPF.

### R1

```
Interface tunnel 1
 tunnel source 200.1.1.1
 tunnel destination 199.1.1.3
 tunnel mode ipv6ip
 ipv6 address 2000:1234:ABCD:01FE::1/64
 ipv6 ospf 1 area 0
```

### R3

```
Interface tunnel 1
 tunnel source 199.1.1.3
 tunnel destination 200.1.1.1
 tunnel mode ipv6ip
 ipv6 address 2000:1234:ABCD:01FE::3/64
 ipv6 ospf 1 area 0
```

## **Task 2**

Configure route redistribution on R3 between OSPF and IS-IS.

### **R3**

```
Ipv6 router ospf 1
  Redistribute isis
!
Router isis
  Address-family ipv6 unicast
  Redistribute ospf 1
```

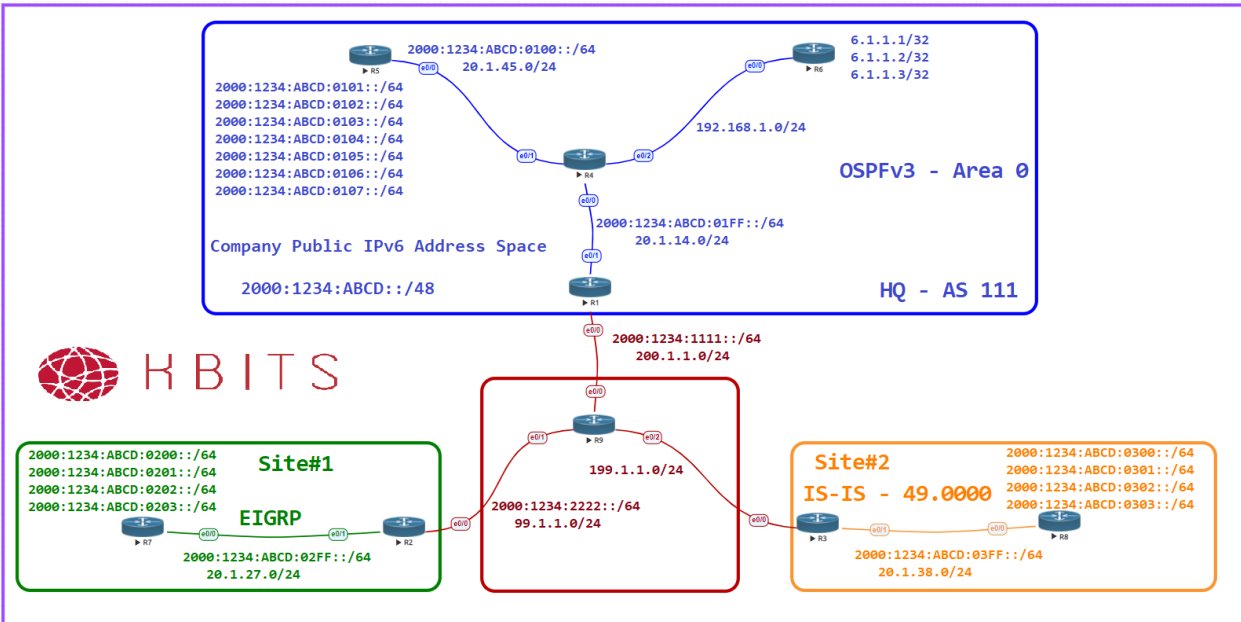
## **Task 3**

Configure route redistribution on R1 between OSPF and BGP for external OSPF routes as well.

### **R1**

```
Router bgp 111
  Address-family ipv6
  Redistribute ospf 1 match internal external
```

# Lab 7 – Configuring NAT64



## Task 1

Enable NAT64 on all Interfaces on R4.

### R4

```
Interface E0/0
 nat64 enable
!
Interface E0/1
 nat64 enable
!
Interface E0/2
 nat64 enable
```

## Task 2

Dedicate an IPv6 network prefix for NAT64

### R4

```
nat64 prefix stateful 2000:1234:ABCD:0400::/64
```

### Task 3

Inject the NAT64 into the IPv6 network by creating a Null 0 route for it and redistributing it into BGP. Allow this route to be redistributed into BGP on R1.

#### R4

```
ipv6 route 2000:1234:ABCD:400::/64 Null0
ipv6 router ospf 1
 redistribute static
```

#### R1

```
Router bgp 111
Address-family ipv6
Redistribute ospf 1 match internal external
```

### Task 4

Configure Static NAT for IPv4 Servers. Translate to the following:

- 6.1.1.1 – 2000:1234:ABCD:0400::1
- 6.1.1.2 – 2000:1234:ABCD:0400::2
- 6.1.1.3 – 2000:1234:ABCD:0400::3

#### R4

```
nat64 v4v6 static 6.1.1.1 2000:1234:ABCD:0400::1
nat64 v4v6 static 6.1.1.2 2000:1234:ABCD:0400::2
nat64 v4v6 static 6.1.1.3 2000:1234:ABCD:0400::3
```

### Task 5

Configure Dynamic PAT for your networks (2000:1234:ABCD::/64 to a pool of 10.10.10.1 & 10.10.10.2).

#### R4

```
ipv6 access-list IPV6LIST
 permit ip 2000:1234:ABCD::/48 any
!
nat64 v4 pool V4POOL 10.10.10.1 10.10.10.2
!
nat64 v6v4 list IPV6LIST pool V4POOL overload
```

# Configuring Virtual Private Networks (VPNs)

**Authored By:**

**Khawar Butt**

CCIE # 12353

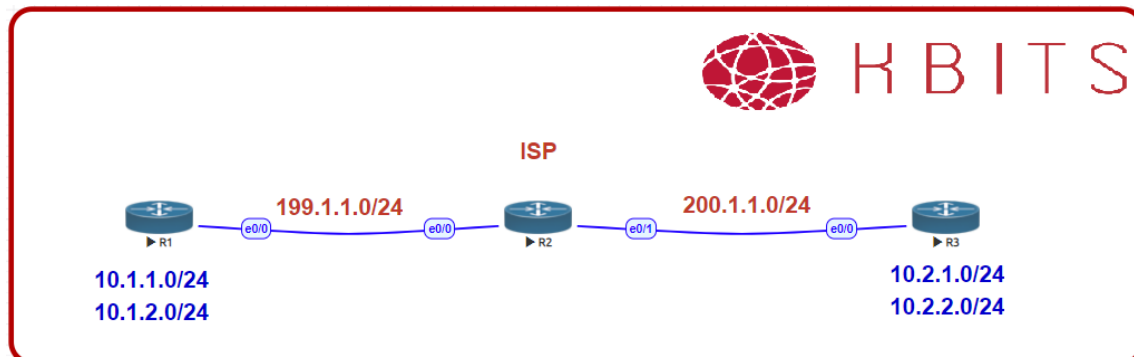
Hepta CCIE#12353

CCDE # 20110020

**Configuring Virtual Private Networks (VPNs)**



# Lab 1 – Point-to-Point GRE



## Interface IP Address Configuration

### R1

Interface	IP Address	Subnet Mask
Loopback 0	10.1.1.1	255.255.255.0
Loopback 1	10.1.2.1	255.255.255.0
E 0/0	199.1.1.1	255.255.255.0

### R2

Interface	IP Address	Subnet Mask
E 0/0	199.1.1.2	255.255.255.0
E 0/1	200.1.1.2	255.255.255.0

### R3

Interface	IP Address	Subnet Mask
Loopback 0	10.2.1.1	255.255.255.0
Loopback 1	10.2.2.1	255.255.255.0
E 0/0	200.1.1.3	255.255.255.0



### Task 1

Configure Default routes on R1 & R2 pointing towards R2 (ISP).

<b>R1</b> Ip route 0.0.0.0 0.0.0.0 199.1.1.2	<b>R3</b> Ip route 0.0.0.0 0.0.0.0 200.1.1.2
---	---

### Task 2

Configure a Point-to-Point GRE tunnel between R1 and R3. Use 192.168.13.0/24 as the Tunnel Network IP.

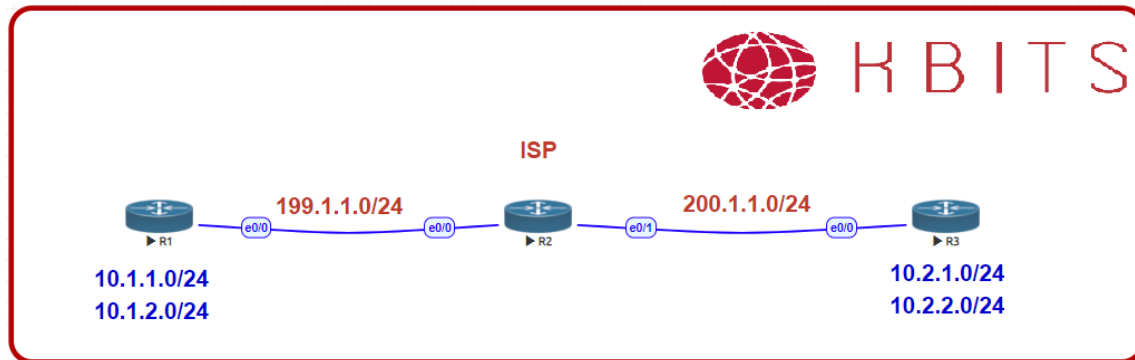
<b>R1</b> Interface Tunnel 1 Ip add 192.168.13.1 255.255.255.0 Tunnel source 199.1.1.1 Tunnel destination 200.1.1.3	<b>R3</b> Interface Tunnel 1 Ip add 192.168.13.3 255.255.255.0 Tunnel source 200.1.1.3 Tunnel destination 199.1.1.1
---	---

### Task 3

Configure EIGRP in AS 13 to route the internal networks (Loopbacks) on the GRE Tunnel between R1 and R3.

<b>R1</b> Router EIGRP 13 No auto-summary Network 192.168.13.0 Network 10.0.0.0	<b>R3</b> Router EIGRP 13 No auto-summary Network 192.168.13.0 Network 10.0.0.0
---	---

# Lab 2 – Encrypting GRE Tunnels Using IPsec



## Task 1

Configure IPsec to encrypt the traffic passing thru the GRE tunnel. Make sure the packet does not duplicate the IP addresses in the Header. Use the following parameters for the IPsec Tunnel:

- ISAKMP Parameters
  - Authentication : Pre-shared
  - Encryption : 3DES
  - Group : 2
  - Hash : MD5
  - Pre-Shared Key : **cisco**
- IPsec Parameters
  - Encryption : ESP-3DES
  - Authentication : ESP-SHA-HMAC

## R1

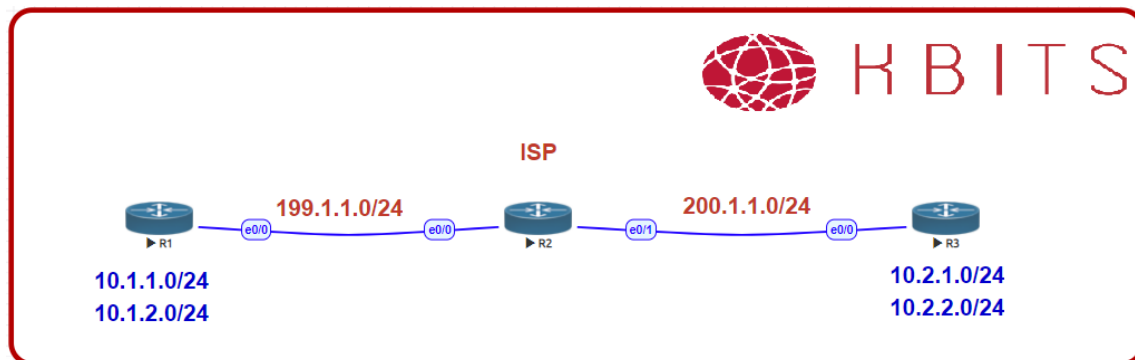
```
Crypto isakmp policy 10
Authentication pre-share
Hash md5
Group 2
Encryption 3des
!
Crypto isakmp key cisco address 200.1.1.3
!
crypto ipsec transform-set t-set esp-3des esp-sha-hmac
mode transport
!
crypto ipsec profile IPSEC
set transform-set t-set
!
Interface Tunnel 1
```

```
Tunnel protection ipsec profile IPSEC
```

```
R3
```

```
Crypto isakmp policy 10
  Authentication pre-share
  Hash md5
  Group 2
  Encryption 3des
!
Crypto isakmp key cisco address 199.1.1.1
!
crypto ipsec transform-set t-set esp-3des esp-sha-hmac
  mode transport
!
crypto ipsec profile IPSEC
  set transform-set t-set
!
Interface Tunnel 1
  Tunnel protection ipsec profile IPSEC
```

# Lab 3 – Configuring a Native IPsec Tunnel Interface



## Task 1

Convert the Existing GRE/IPsec tunnel into a Native IPsec tunnel by changing the Tunnel mode to IPsec.

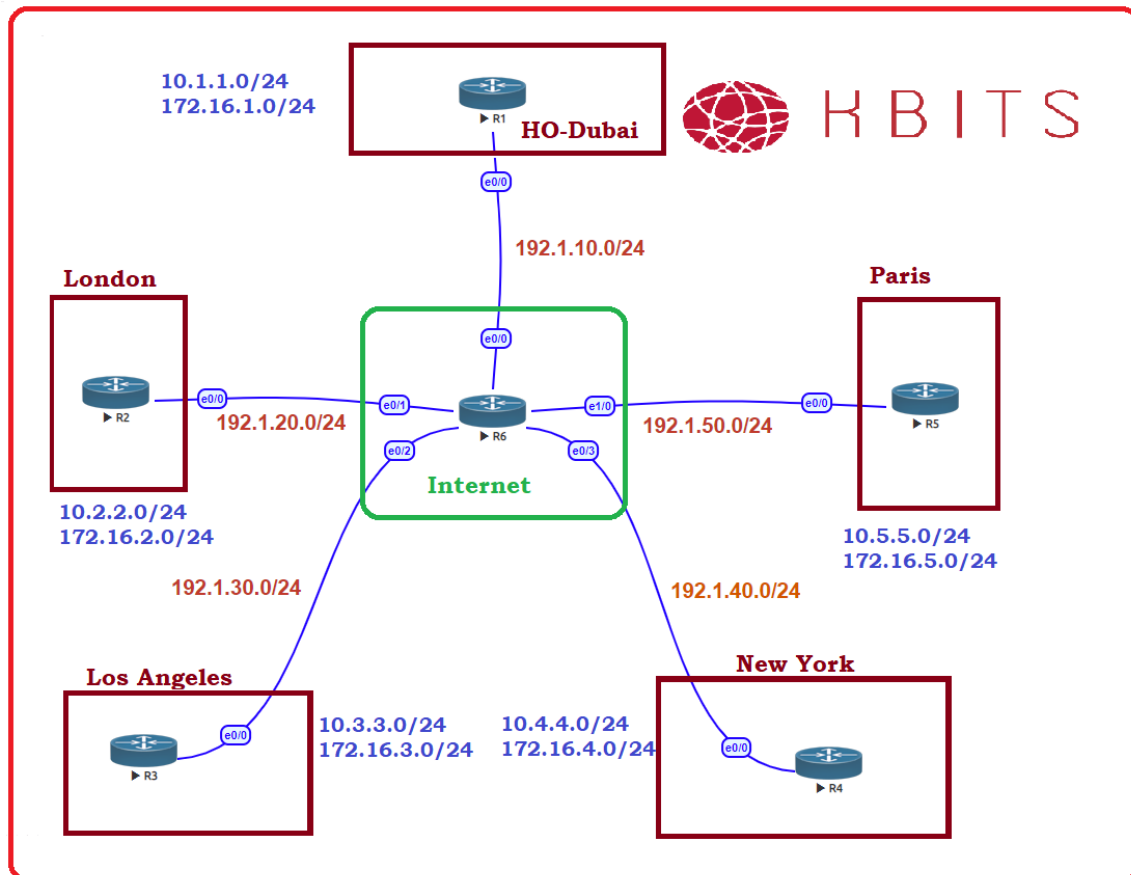
### R1

```
Interface Tunnel 1
Tunnel mode ipsec ipv4
```

### R3

```
Interface Tunnel 1
Tunnel mode ipsec ipv4
```

# Lab 4 – Configuring a mGRE VPN



## Interface IP Address Configuration

### R1

Interface	IP Address	Subnet Mask
Loopback 0	10.1.1.1	255.255.255.0
Loopback 1	172.16.1.1	255.255.255.0
E 0/0	192.1.10.1	255.255.255.0

### R2

Interface	IP Address	Subnet Mask
Loopback 0	10.2.2.2	255.255.255.0
Loopback 1	172.16.2.2	255.255.255.0
E 0/0	192.1.20.2	255.255.255.0

### R3

Interface	IP Address	Subnet Mask
Loopback 0	10.3.3.3	255.255.255.0
Loopback 1	172.16.3.3	255.255.255.0
E 0/0	192.1.30.3	255.255.255.0

### R4

Interface	IP Address	Subnet Mask
Loopback 0	10.4.4.4	255.255.255.0
Loopback 1	172.16.4.4	255.255.255.0
E 0/0	192.1.40.4	255.255.255.0

### R5

Interface	IP Address	Subnet Mask
Loopback 0	10.5.5.5	255.255.255.0
Loopback 1	172.16.5.5	255.255.255.0
E 0/0	192.1.50.5	255.255.255.0

### R6

Interface	IP Address	Subnet Mask
E 0/0	192.1.10.6	255.255.255.0
E 0/1	192.1.20.6	255.255.255.0
E 0/2	192.1.30.6	255.255.255.0
E 0/3	192.1.40.6	255.255.255.0
E 1/0	192.1.50.6	255.255.255.0

### Task 1

Configure Default routes on R1 – R5 pointing towards R6 (ISP).

<b>R1</b> Ip route 0.0.0.0 0.0.0.0 192.1.10.6	<b>R2</b> Ip route 0.0.0.0 0.0.0.0 192.1.20.6
<b>R3</b> Ip route 0.0.0.0 0.0.0.0 192.1.30.6	<b>R4</b> Ip route 0.0.0.0 0.0.0.0 192.1.40.6
<b>R5</b> Ip route 0.0.0.0 0.0.0.0 192.1.50.6	

## Task 2

Configure a MultiPoint GRE tunnel between R1, R2, R3, R4 & R5. Use 192.168.1.0/24 as the Tunnel Network IP. The NHRP mapping will be done in the next Task. Use the following parameters for your MGRE Tunnel:

- NHRP Parameters
  - NHRP ID – 100
  - NHRP Authentication key – cisco
- Tunnel Parameters
  - Tunnel Authentication Key : 100

### R1

```
Interface Tunnel 1
Ip address 192.168.1.1 255.255.255.0
Ip nhrp network-id 100
Ip nhrp authentication cisco
Tunnel source E 0/0
Tunnel mode gre multipoint
Tunnel key 1234
```

### R2

```
Interface Tunnel 1
Ip address 192.168.1.2 255.255.255.0
Ip nhrp network-id 100
Ip nhrp authentication cisco
Tunnel source E 0/0
Tunnel mode gre multipoint
Tunnel key 100
```

### R3

```
Interface Tunnel 1
Ip address 192.168.1.3 255.255.255.0
Ip nhrp network-id 100
Ip nhrp authentication cisco
Tunnel source E 0/0
Tunnel mode gre multipoint
Tunnel key 100
```

### R4

```
Interface Tunnel 1
Ip address 192.168.1.4 255.255.255.0
Ip nhrp network-id 100
Ip nhrp authentication cisco
```

```
Tunnel source E 0/0
Tunnel mode gre multipoint
Tunnel key 100
```

### **R5**

```
Interface Tunnel 1
Ip address 192.168.1.5 255.255.255.0
Ip nhrp network-id 100
Ip nhrp authentication cisco
Tunnel source E 0/0
Tunnel mode gre multipoint
Tunnel key 100
```

### **Task 3**

Configure NHRP Mapping allowing all devices to connect to each other directly for Unicast traffic. Configure Multicast mappings in such a way that all devices use R1 as the routing hub.

### **R1**

```
Interface Tunnel 1
Ip nhrp map 192.168.1.2 192.1.20.2
Ip nhrp map 192.168.1.3 192.1.30.3
Ip nhrp map 192.168.1.4 192.1.40.4
Ip nhrp map 192.168.1.5 192.1.50.5
Ip nhrp map Multicast 192.1.20.2
Ip nhrp map Multicast 192.1.30.3
Ip nhrp map Multicast 192.1.40.4
Ip nhrp map Multicast 192.1.50.5
```

### **R2**

```
Interface Tunnel 1
Ip nhrp map 192.168.1.1 192.1.10.1
Ip nhrp map 192.168.1.3 192.1.30.3
Ip nhrp map 192.168.1.4 192.1.40.4
Ip nhrp map 192.168.1.5 192.1.50.5
Ip nhrp map Multicast 192.1.10.1
```

### **R3**

```
Interface Tunnel 1
Ip nhrp map 192.168.1.1 192.1.10.1
Ip nhrp map 192.168.1.2 192.1.20.2
Ip nhrp map 192.168.1.4 192.1.40.4
Ip nhrp map 192.168.1.5 192.1.50.5
Ip nhrp map Multicast 192.1.10.1
```



**R4**

```
Interface Tunnel 1
Ip nhrp map 192.168.1.1 192.1.10.1
Ip nhrp map 192.168.1.2 192.1.20.2
Ip nhrp map 192.168.1.3 192.1.30.3
Ip nhrp map 192.168.1.5 192.1.50.5
Ip nhrp map Multicast 192.1.10.1
```

**R5**

```
Interface Tunnel 1
Ip nhrp map 192.168.1.1 192.1.10.1
Ip nhrp map 192.168.1.2 192.1.20.2
Ip nhrp map 192.168.1.3 192.1.30.3
Ip nhrp map 192.168.1.4 192.1.40.4
Ip nhrp map Multicast 192.1.10.1
```

**Task 4**

Configure EIGRP in AS 100 to route the internal networks (Loopbacks) on the GRE Tunnel on all the MGRE Routers. Disable Split horizon on R1 to allow it propagate routes from the Spoke routers to the other spoke routers.

Note: You might need to bounce the Tunnel interface to make the Routing work. Bring up the Hub router before the Spoke Routers.

**R1**

```
Router EIGRP 100
No auto-summary
Network 192.168.1.0
Network 10.0.0.0
!
Interface Tunnel 1
No ip split-horizon eigrp 100
Shut
No shut
```

**R2**

```
Router EIGRP 100
No auto-summary
Network 192.168.1.0
Network 10.0.0.0
```

**R3**

Router EIGRP 100  
No auto-summary  
Network 192.168.1.0  
Network 10.0.0.0

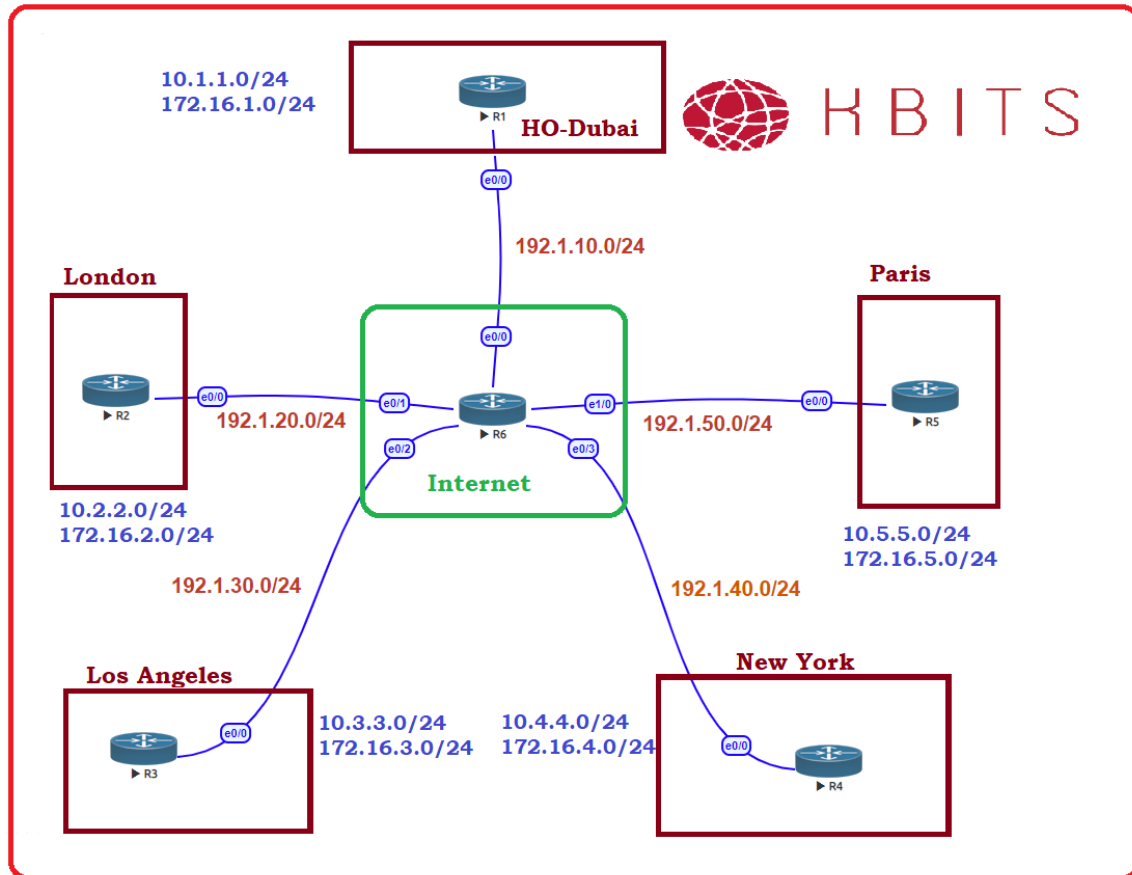
**R4**

Router EIGRP 100  
No auto-summary  
Network 192.168.1.0  
Network 10.0.0.0

**R5**

Router EIGRP 100  
No auto-summary  
Network 192.168.1.0  
Network 10.0.0.0

# Lab 5 – Configuring DMVPN – Phase I



## Task 1

De-Configure the Tunnels created in the previous Lab.

<b>R1</b> No Interface Tunnel 1	<b>R2</b> No Interface Tunnel 1
<b>R3</b> No Interface Tunnel 1	<b>R4</b> No Interface Tunnel 1
<b>R5</b> No Interface Tunnel 1	

## Task 2

Configure a MultiPoint GRE tunnel between R1, R2, R3 & R4. Use 192.168.1.0/24 as the Tunnel Network IP. Tunnel:

- NHRP Parameters
  - NHRP ID – 100
  - NHRP Authentication key – cisco
  - NHS : R1
  - Routing Hub: R1 [Configure the multicast mapping to accommodate routing protocols]
- Tunnel Parameters
  - Tunnel Authentication Key : 100

### R1

```
Interface Tunnel 1
Ip address 192.168.1.1 255.255.255.0
Ip nhrp network-id 100
Ip nhrp authentication cisco
Ip nhrp map multicast dynamic
Tunnel source E 0/0
Tunnel mode gre multipoint
Tunnel key 100
```

### R2

```
Interface Tunnel 1
Ip address 192.168.1.2 255.255.255.0
Ip nhrp network-id 100
Ip nhrp authentication cisco
Ip nhrp nhs 192.168.1.1
Ip nhrp map 192.168.1.1 192.1.10.1
Ip nhrp map multicast 192.1.10.1
Tunnel source E 0/0
Tunnel mode gre multipoint
Tunnel key 100
```

### R3

```
Interface Tunnel 1
Ip address 192.168.1.3 255.255.255.0
Ip nhrp network-id 100
Ip nhrp authentication cisco
Ip nhrp nhs 192.168.1.1
Ip nhrp map 192.168.1.1 192.1.10.1
Ip nhrp map multicast 192.1.10.1
Tunnel source E 0/0
```

```
Tunnel mode gre multipoint
Tunnel key 100
```

#### **R4**

```
Interface Tunnel 1
Ip address 192.168.1.4 255.255.255.0
Ip nhrp network-id 100
Ip nhrp authentication cisco
Ip nhrp nhs 192.168.1.1
Ip nhrp map 192.168.1.1 192.1.10.1
Ip nhrp map multicast 192.1.10.1
Tunnel source E 0/0
Tunnel mode gre multipoint
Tunnel key 100
```

#### **Task 3**

Configure EIGRP in AS 100 to route the internal networks (Loopbacks) on the GRE Tunnel on all the MGRE Routers. Disable Split horizon on R1 to allow it propagate routes from the Spoke routers to the other spoke routers.

#### **R1**

```
Interface Tunnel 1
No ip split-horizon eigrp 100
!
Router EIGRP 100
No auto-summary
Network 192.168.1.0
Network 10.0.0.0
```

#### **R2**

```
Router EIGRP 100
No auto-summary
Network 192.168.1.0
Network 10.0.0.0
```

#### **R3**

```
Router EIGRP 100
No auto-summary
Network 192.168.1.0
Network 10.0.0.0
```

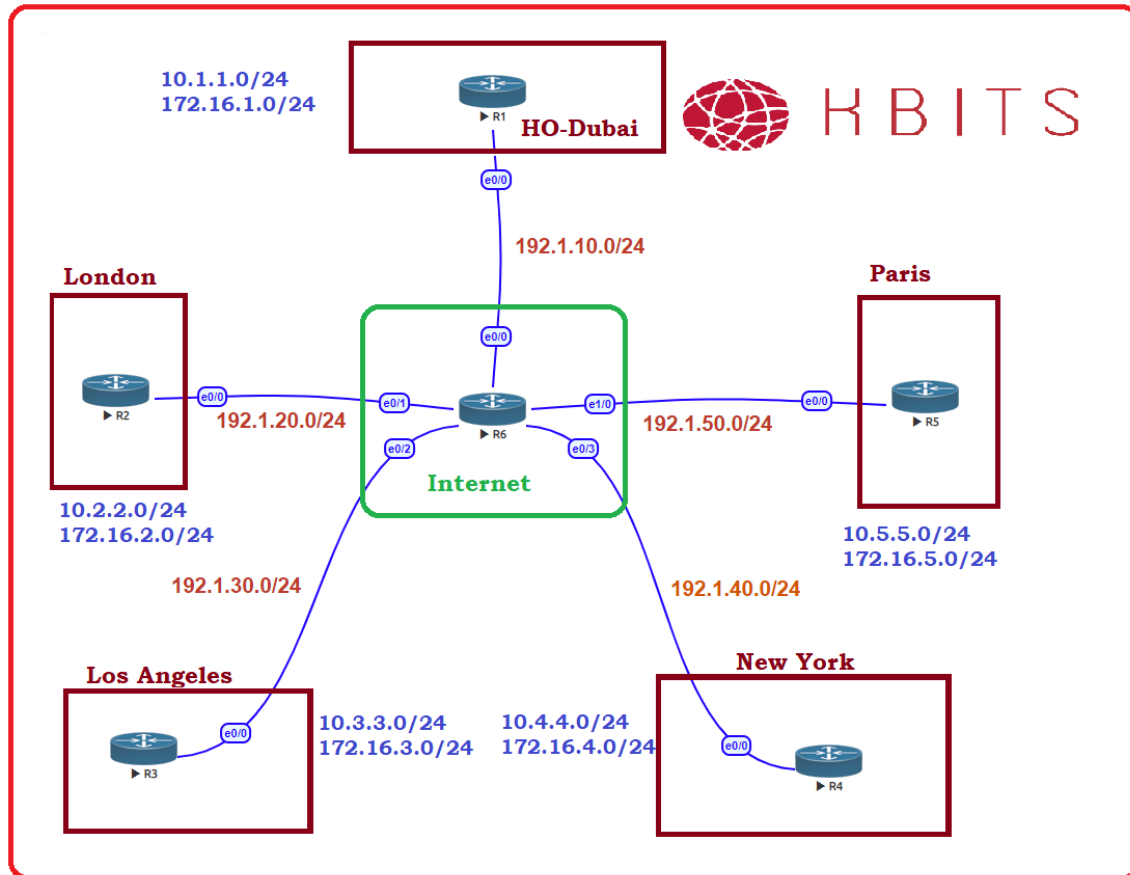
#### **R4**

```
Router EIGRP 100
```

```
No auto-summary
Network 192.168.1.0
Network 10.0.0.0
```

**Note:** The default behavior of EIGRP is to change the Next-hop to itself while propagating the spoke routes to other spokes. The result is that the spokes will use the hub route as the next hop for all spoke-to-spoke traffic. **This is DMVPN Phase I [Hub-n-Spoke forwarding]**

# Lab 6 – Configuring DMVPN – Phase II



## Task 1

Disable the Hub from changing the next-hop attribute on the hub.

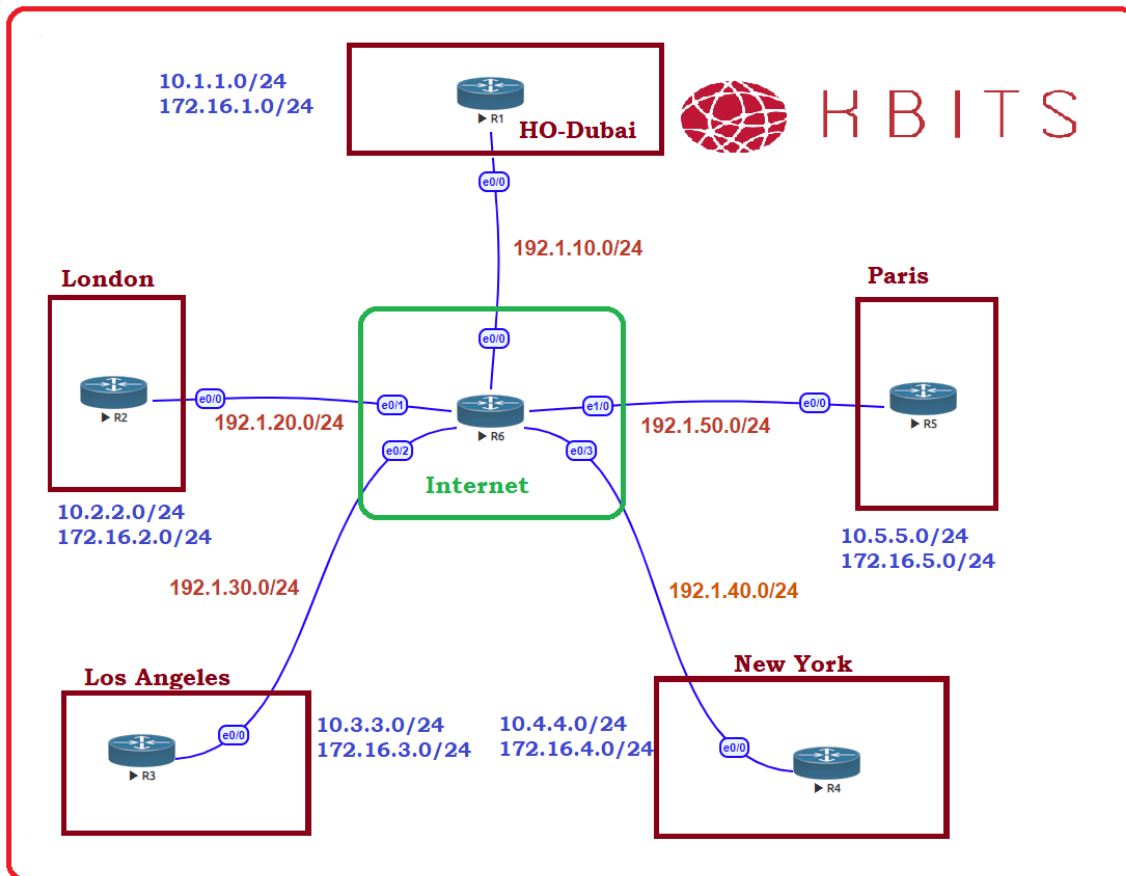
### R1

```
Interface Tunnel 1  
No ip next-hop-self eigrp 100
```

**Note:** Check the Routing table. The next-hop attribute for the Spoke-routes is unchanged by the hub and is directly pointing to the spoke Tunnel IP. This causes the spokes to do a NHRP resolution directly for the spoke. Although the resolution packet will go thru the hub, the actual packet will take the direct path. Use the traceroute command to verify this.

**This is DMVPN Phase II.** In this phase, the spoke-to-spoke traffic is forwarded directly between the spokes. Phase II is accomplished by tweaking the Routing protocol behavior.

# Lab 7 – Configuring DMVPN – Phase III



## Task 1

Change the Next-hop back to Self. All routes should again have a next-hop pointing to the Hub [ DMVPN Phase I]

### R1

```
Interface Tunnel 1  
ip next-hop-self eigrp 100
```

## Task 2

Configure NHRP Redirection on the Hub such that the Hub should push down a dynamic mapping to the spokes for the spoke internal routes. Configure the spokes to accept the mapping.

### R1

```
Interface Tunnel 1  
Ip nhrp redirect
```



**R2**

```
Interface Tunnel 1
Ip nhrp shortcut
```

**R3**

```
Interface Tunnel 1
Ip nhrp shortcut
```

**R4**

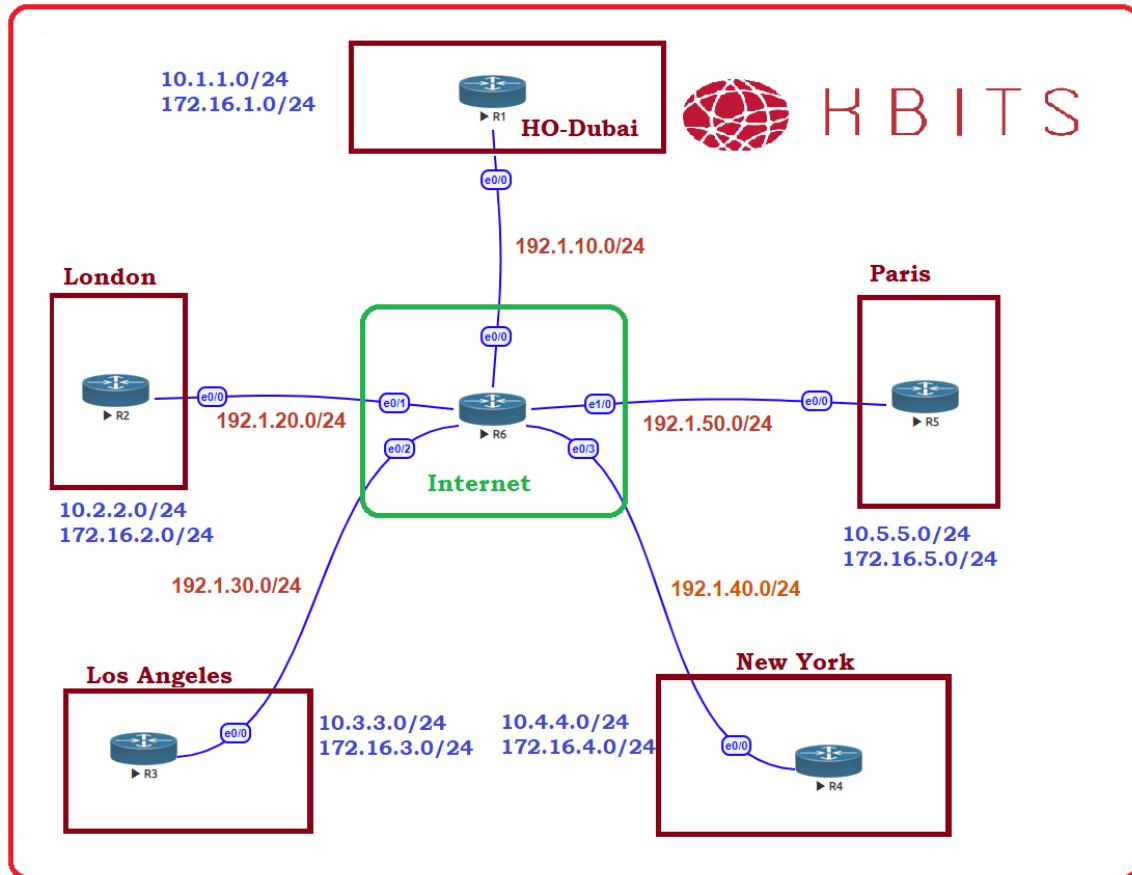
```
Interface Tunnel 1
Ip nhrp shortcut
```

**Note:** Check the Routing table. The next-hop attribute is pointing to the hub. Do a traceroute from the R2 to R4. You will notice the first trace goes thru the hub. This is due to the routing table pointing towards the Hub. The hub detects that the spokes are both connected on the same tunnel interface, hence sends a NHRP redirect message to both of them. The NHRP redirect message contains the mapping for the destination public IP for the internal networks.

The subsequent packets will be forwarded directly from spoke to spoke. If you check the routing table entry, it still points to the Hub. If you check the NHRP table, you will see an entry for the destination spoke network with the Spoke public IP.

**This is DMVPN Phase III.** In this phase, the spoke-to-spoke traffic is forwarded directly between the spokes. Phase III is accomplished by using NHRP redirect messages to override the routing table.

# Lab 8 – Configuring a Dual-Hub DMVPN



## Task 1

Configure a Static Tunnel between R1 and R5. R5 should be configured with a Tunnel IP address of 192.168.1.5/24 using the Tunnel parameters specified on the other routers (R1 – R4 – Lab 5). Enable EIGRP on R5.

## R5

```
Interface Tunnel 1
Ip address 192.168.1.5 255.255.255.0
Ip nhrp network-id 100
Ip nhrp authentication cisco
Ip nhrp map 192.168.1.1 192.1.10.1
Ip nhrp map multicast 192.1.10.1
Tunnel source E 0/0
Tunnel mode gre multipoint
Tunnel key 100
No ip split-horizon eigrp 100
!
```

```
router eigrp 100
 Network 192.168.1.0
 Network 10.0.0.0
```

### **R1**

```
Interface Tunnel1
 Ip nhrp map 192.168.1.5 192.1.50.5
 Ip nhrp map multicast 192.1.50.5
```

### **Task 2**

Configure R5 as another NHS in your network. Configure R2, R3 & R4 to use R5 as the NHS Server and a Routing hub as well.

### **R2**

```
Interface Tunnel1
 Ip nhrp nhs 192.168.1.5
 Ip nhrp map 192.168.1.5 192.1.50.5
 Ip nhrp map multicast 192.1.50.5
```

### **R3**

```
Interface Tunnel1
 Ip nhrp nhs 192.168.1.5
 Ip nhrp map 192.168.1.5 192.1.50.5
 Ip nhrp map multicast 192.1.50.5
```

### **R4**

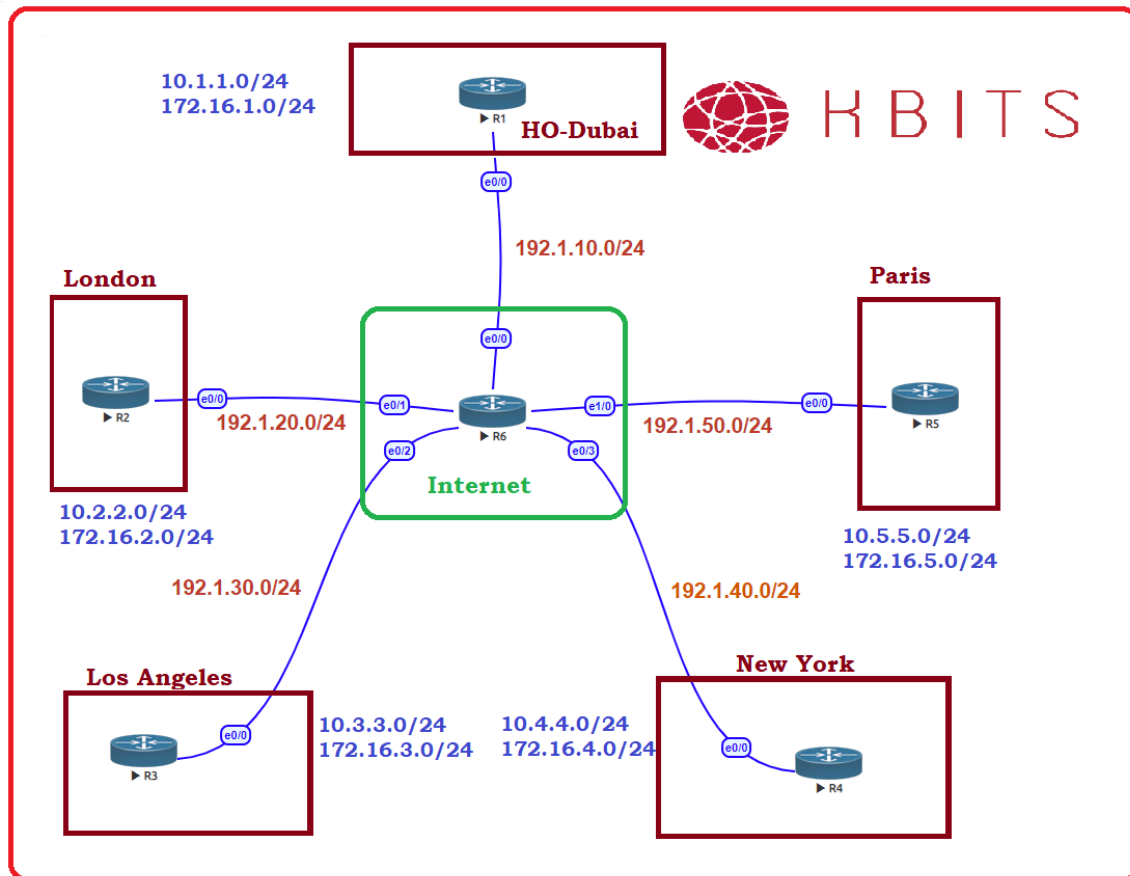
```
Interface Tunnel1
 Ip nhrp nhs 192.168.1.5
 Ip nhrp map 192.168.1.5 192.1.50.5
 Ip nhrp map multicast 192.1.50.5
```

### **R5**

```
Interface Tunnel 1
 Ip nhrp map multicast dynamic
 No ip split-horizon eigrp 100
 Ip nhrp redirect
```

Note: You should now see routes from both Routing hubs. Although, it sees 2 entries, the Data path will be direct due to Phase III.

# Lab 9 – Encrypting the DMVPN Traffic using IPsec



## Task 1

Configure IPsec to encrypt the traffic passing thru the tunnel. Make sure the packet does not duplicate the IP addresses in the Header. Use the following parameters for the IPsec Tunnel:

- ISAKMP Parameters
  - Authentication : Pre-shared
  - Encryption : 3DES
  - Group : 2
  - Hash : MD5
  - Pre-Shared Key : **cisco**
- IPsec Parameters
  - Encryption : ESP-3DES
  - Authentication : ESP-SHA-HMAC

**R1**

```
Crypto isakmp policy 10
  Authentication pre-share
  Hash md5
  Group 2
  Encryption 3des
!
Crypto isakmp key cisco address 0.0.0.0
!
crypto ipsec transform-set t-set esp-3des esp-sha-hmac
mode transport
!
crypto ipsec profile IPSEC
set transform-set t-set
!
Interface Tunnel 1
  Tunnel protection ipsec profile IPSEC
```

**R2**

```
Crypto isakmp policy 10
  Authentication pre-share
  Hash md5
  Group 2
  Encryption 3des
!
Crypto isakmp key cisco address 0.0.0.0
!
crypto ipsec transform-set t-set esp-3des esp-sha-hmac
mode transport
!
crypto ipsec profile IPSEC
set transform-set t-set
!
Interface Tunnel 1
  Tunnel protection ipsec profile IPSEC
```

**R3**

```
Crypto isakmp policy 10
  Authentication pre-share
  Hash md5
  Group 2
  Encryption 3des
!
```

```
Crypto isakmp key cisco address 0.0.0.0
!
crypto ipsec transform-set t-set esp-3des esp-sha-hmac
mode transport
!
crypto ipsec profile IPSEC
set transform-set t-set
!
Interface Tunnel 1
Tunnel protection ipsec profile IPSEC
```

#### **R4**

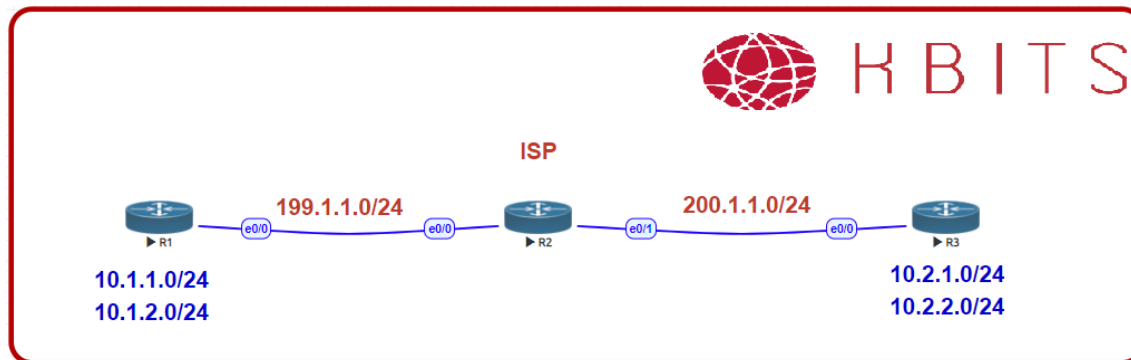
```
Crypto isakmp policy 10
Authentication pre-share
Hash md5
Group 2
Encryption 3des
!
Crypto isakmp key cisco address 0.0.0.0
!
crypto ipsec transform-set t-set esp-3des esp-sha-hmac
mode transport
!
crypto ipsec profile IPSEC
set transform-set t-set
!
Interface Tunnel 1
Tunnel protection ipsec profile IPSEC
```

#### **R5**

```
Crypto isakmp policy 10
Authentication pre-share
Hash md5
Group 2
Encryption 3des
!
Crypto isakmp key cisco address 0.0.0.0
!
crypto ipsec transform-set t-set esp-3des esp-sha-hmac
mode transport
!
crypto ipsec profile IPSEC
set transform-set t-set
!
Interface Tunnel 1
```

Tunnel protection ipsec profile IPSEC

# Lab 10 – Configuring Flex VPN – Point-to-Point



## Interface IP Address Configuration

### R1

Interface	IP Address	Subnet Mask
Loopback 0	10.1.1.1	255.255.255.0
Loopback 1	10.1.2.1	255.255.255.0
E 0/0	199.1.1.1	255.255.255.0

### R2

Interface	IP Address	Subnet Mask
E 0/0	199.1.1.2	255.255.255.0
E 0/1	200.1.1.2	255.255.255.0

### R3

Interface	IP Address	Subnet Mask
Loopback 0	10.2.1.1	255.255.255.0
Loopback 1	10.2.2.1	255.255.255.0
E 0/0	200.1.1.3	255.255.255.0



### Task 1

Configure Default routes on R1 & R2 pointing towards R2 (ISP).

<b>R1</b> Ip route 0.0.0.0 0.0.0.0 199.1.1.2	<b>R3</b> Ip route 0.0.0.0 0.0.0.0 200.1.1.2
---	---

### Task 2

Configure a Site-to-Site Flex VPN to encrypt traffic from 10.1.X.0/24 networks on R1 (Loopback 0 & Loopback 1) to the 10.3.X.0/24 on R3 (Loopback 0 & Loopback 1). Do not create a Static VTI on R1. It should be created dynamically based on an incoming connection from R3.

### Task 3

Use the following Parameters for the Tunnel between R1 and R3:

- IKEv2 Proposal Parameters
  - Integrity: SHA1
  - Encryption: 3DES
  - Group: 2
  - Authentication: Pre-share
  - Pre-Shared Key: **cisco**
- IPsec Parameters
  - Encryption: ESP-3DES
  - Authentication: ESP-MD5-HMAC
- Tunnel IP Address:
  - IP Address for Virtual-template on R1: 192.168.1.1/24
  - Tunnel Interface on R3: 192.168.1.3/24

### R1

```
int Loopback11
 ip add 192.168.1.1 255.255.255.0
!
int virtual-template 1 type tunnel
 ip unnumbered Loopback11
 tunnel source 199.1.1.1
 tunnel mode ipsec ipv4
!
crypto ikev2 proposal PROP_1
 integrity sha1
 group 2
 encryption 3des
!
```

```
crypto ikev2 policy POL_1
 proposal PROP_1
!
crypto ikev2 keyring KR_R3
 peer R3
  address 0.0.0.0
  pre-shared local cisco
  pre-shared remote cisco
!
crypto ikev2 profile PROF_1
 match identity remote address 0.0.0.0
 authentication remote pre-share
 authentication local pre-share
 keyring local KR_R3
!
crypto ipsec transform-set ABC esp-3des esp-md5-hmac
!
crypto ipsec profile ABC
 set transform-set ABC
 set ikev2-profile PROF_1
!
int virtual-template 1 type tunnel
 tunnel protection ipsec profile ABC
!
crypto ikev2 profile PROF_1
 virtual-template 1
!
router eigrp 100
 network 10.0.0.0
 network 192.168.1.0
```

### **R3**

```
crypto ikev2 proposal PROP_1
 integrity sha1
 group 2
 encryption 3des
!
crypto ikev2 policy POL_1
 proposal PROP_1
!
crypto ikev2 keyring KR_R1
 peer R1
  address 199.1.1.1
  pre-shared local cisco
  pre-shared remote cisco
```

```
!  
crypto ikev2 profile PROF_1  
  match identity remote address 199.1.1.1 255.255.255.255  
  authentication remote pre-share  
  authentication local pre-share  
  keyring local KR_R1  
!  
crypto ipsec transform-set ABC esp-3des esp-md5-hmac  
!  
crypto ipsec profile ABC  
  set transform-set ABC  
  set ikev2-profile PROF_1  
!  
int tunnel 1  
  ip add 192.168.1.3 255.255.255.0  
  tunnel source E 0/0  
  tunnel destination 199.1.1.1  
  tunnel mode ipsec ipv4  
  tunnel protection ipsec profile ABC  
!  
router eigrp 100  
  network 10.0.0.0  
  network 192.168.1.0
```

# Configuring MPLS-based Networking

**Authored By:**

**Khawar Butt**

CCIE # 12353

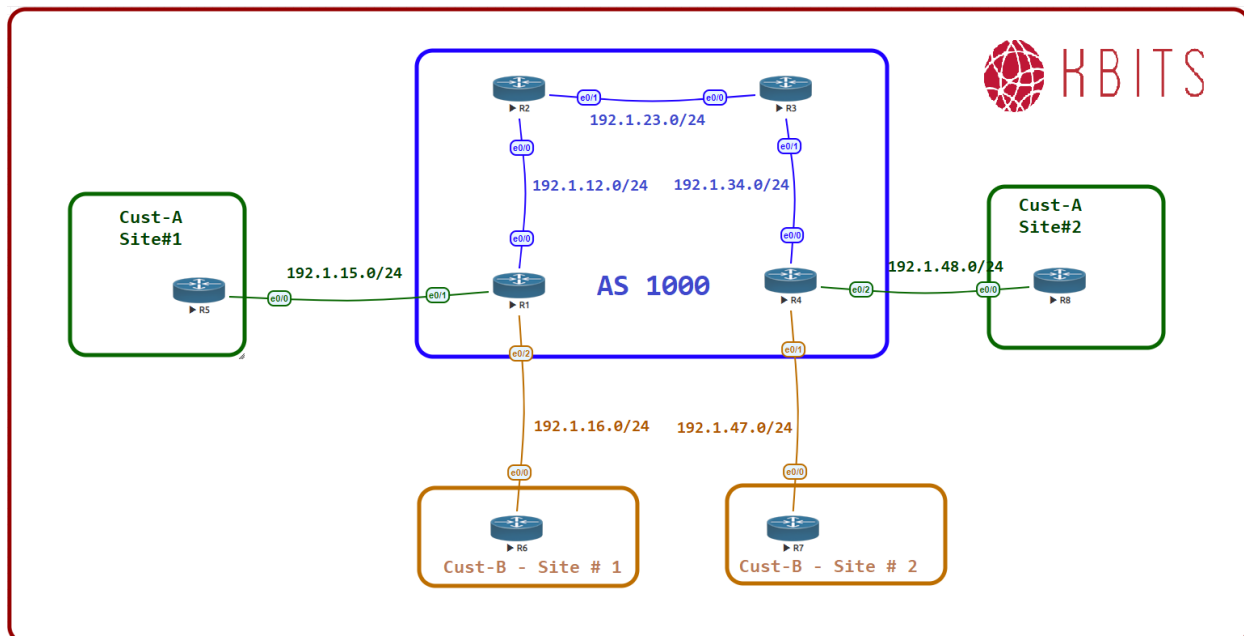
Hepta CCIE#12353

CCDE # 20110020

**Configuring MPLS-based  
Networking**



# Lab 1 – Configuring MPLS Unicast Routing



## Interface IP Address Configuration

### R1

Interface	IP Address	Subnet Mask
Loopback 0	1.1.1.1	255.255.255.255
E 0/0	192.1.12.1	255.255.255.0
E 0/1	192.1.15.1	255.255.255.0
E 0/2	192.1.16.1	255.255.255.0

### R2

Interface	IP Address	Subnet Mask
Loopback 0	2.2.2.2	255.255.255.255
E 0/0	192.1.12.2	255.255.255.0
E 0/1	192.1.23.2	255.255.255.0

### R3

Interface	IP Address	Subnet Mask
Loopback 0	3.3.3.3	255.255.255.255
E 0/0	192.1.23.3	255.255.255.0
E 0/1	192.1.34.3	255.255.255.0

### R4

Interface	IP Address	Subnet Mask
Loopback 0	4.4.4.4	255.255.255.255
E 0/0	192.1.34.4	255.255.255.0
E 0/1	192.1.47.4	255.255.255.0
E 0/2	192.1.48.4	255.255.255.0

### R5

Interface	IP Address	Subnet Mask
Loopback 0	10.5.5.5	255.255.255.0
E 0/0	192.1.15.5	255.255.255.0

### R6

Interface	IP Address	Subnet Mask
Loopback 0	10.6.6.6	255.255.255.0
E 0/0	192.1.16.6	255.255.255.0

### R7

Interface	IP Address	Subnet Mask
Loopback 0	10.7.7.7	255.255.255.0
E 0/0	192.1.47.7	255.255.255.0

### R8

Interface	IP Address	Subnet Mask
Loopback 0	10.8.8.8	255.255.255.0
S 0/0	192.1.48.8	255.255.255.0

### Task 1

Configure OSPF between all the SP routers (R1, R2, R3, R4). Use x.x.x.x as the router-id, where x is the Router number. Advertise all Internal links in OSPF in area 0.

<b>R1</b> Router ospf 1 Router-id 1.1.1.1 Network 1.1.1.1 0.0.0.0 area 0 Network 192.1.12.0 0.0.0.255 area 0	<b>R2</b> Router ospf 1 Router-id 2.2.2.2 Network 2.2.2.2 0.0.0.0 area 0 Network 192.1.12.0 0.0.0.255 area 0 Network 192.1.23.0 0.0.0.255 area 0
<b>R3</b> Router ospf 1 Router-id 3.3.3.3 Network 3.3.3.3 0.0.0.0 area 0 Network 192.1.23.0 0.0.0.255 area 0 Network 192.1.34.0 0.0.0.255 area 0	<b>R4</b> Router ospf 1 Router-id 4.4.4.4 Network 4.4.4.4 0.0.0.0 area 0 Network 192.1.34.0 0.0.0.255 area 0

### Task 2

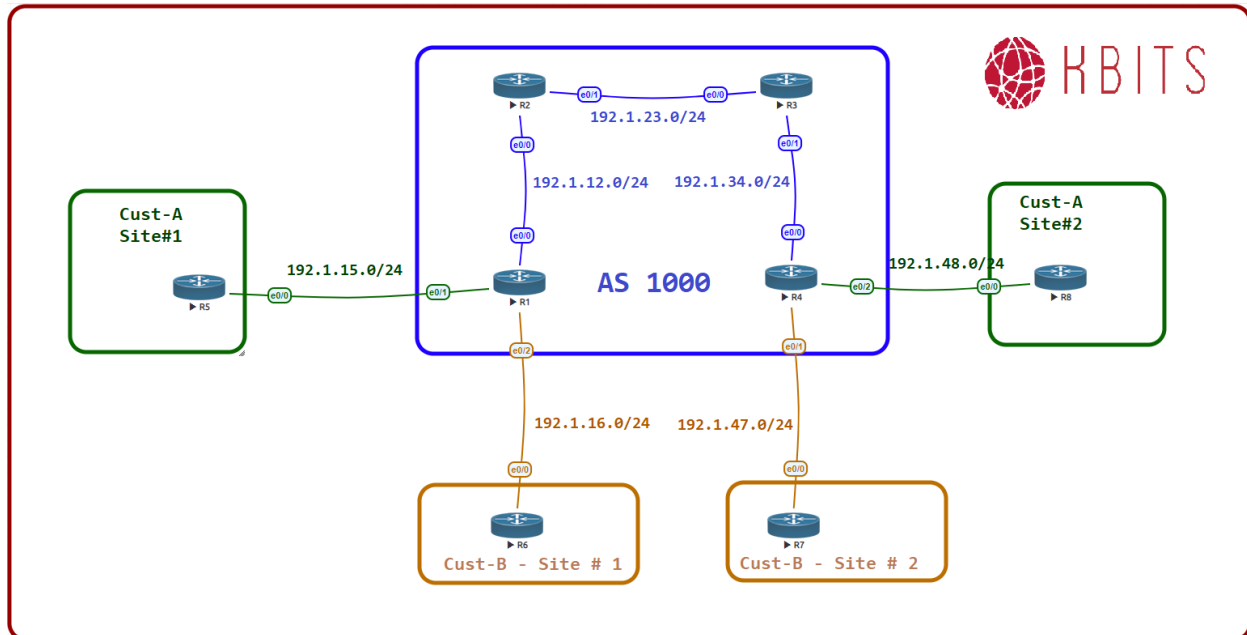
Configure MPLS on all the physical links in the SP Network. Use LDP to distribute labels. The LDP neighbour relationships should be formed based on the most reliable interface. The Labels should be assigned from the range X00 – X99, where X is the router number.

<b>R1</b> Mpls ldp router-id Loopback0 ! Mpls label range 100 199 ! Interface E 0/0 Mpls ip	<b>R2</b> Mpls ldp router-id Loopback0 ! Mpls label range 200 299 ! Interface E 0/0 Mpls ip ! Interface E 0/1 Mpls ip
<b>R3</b> Mpls ldp router-id Loopback0 ! Mpls label range 300 399 ! Interface E 0/0 Mpls ip	<b>R4</b> Mpls ldp router-id Loopback0 ! Mpls label range 400 499 ! Interface E 0/0 Mpls ip

! Interface E 0/1 Mpls ip	
---------------------------------	--



# Lab 2 – Authenticating LDP Peers



## Task 1

All LDP neighbor relationships should be authenticated using a password of **ccie12353**.

### R1

```
Mpls ldp neighbor 2.2.2.2 password ccie12353
```

### R2

```
Mpls ldp neighbor 1.1.1.1 password ccie12353  
Mpls ldp neighbor 3.3.3.3 password ccie12353
```

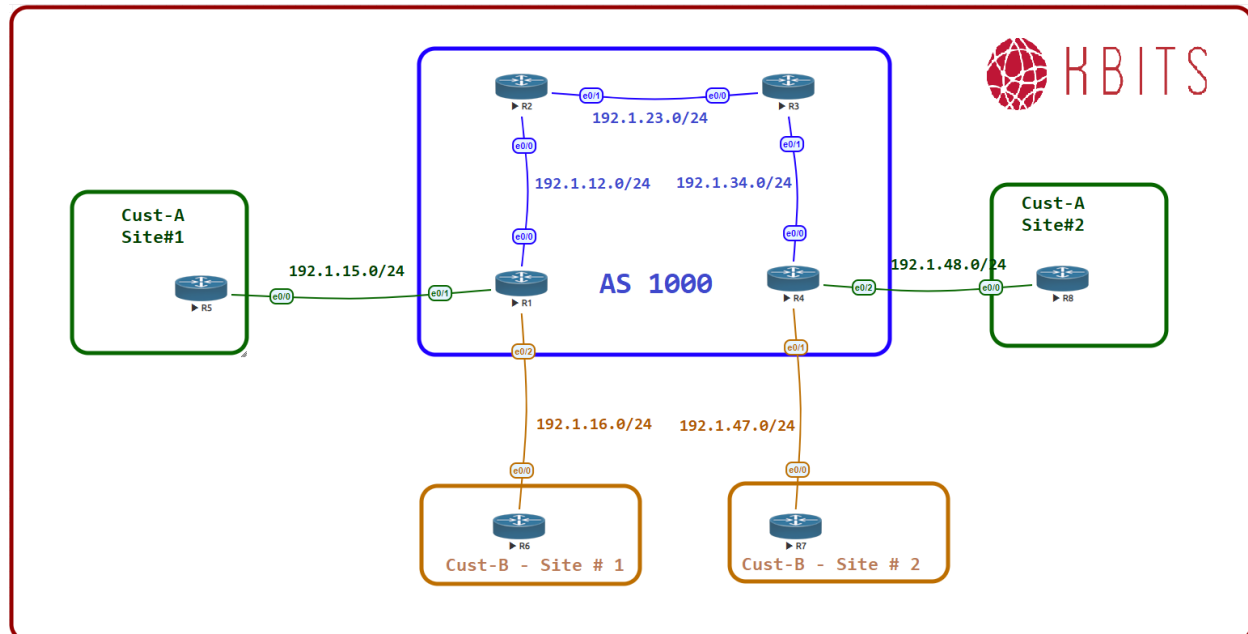
### R3

```
Mpls ldp neighbor 2.2.2.2 password ccie12353  
Mpls ldp neighbor 4.4.4.4 password ccie12353
```

### R4

```
Mpls ldp neighbor 3.3.3.3 password ccie12353
```

# Lab 3 – Configuring MPLS VPN – PE-CE using Static Routing



**Note:**

Save the Configs on all the routers. **Do not save the configs during the labs.** At the completion of this lab, **reload the routers without saving.** This will allow you to do the next lab based on the same topology.

### Task 1

Configure a VPNv4 (MP-iBGP) neighbor relationship between R1 and R4.

#### R1

```
Router BGP 1000
Neighbor 4.4.4.4 remote-as 1000
Neighbor 4.4.4.4 update-source loopback0
!
Address-family vpnv4
Neighbor 4.4.4.4 activate
```

#### R4

```
Router BGP 1000
Neighbor 1.1.1.1 remote-as 1000
Neighbor 1.1.1.1 update-source loopback0
!
Address-family vpnv4
Neighbor 1.1.1.1 activate
```

### Task 2

Configure a VRF **Cust-A** with a RD value of 1000:1 on R1 and R4. Use the same extended community for your Route-target import and export. Assign this VRF to the links that connect to Cust-A sites on R1 and R4.

#### R1

```
Vrf definition Cust-A
rd 1000:1
address-family ipv4
route-target both 1000:1
!
Interface E 0/1
vrf forwarding Cust-A
Ip address 192.1.15.1 255.255.255.0
No shut
```

#### R4

```
Vrf definition Cust-A
rd 1000:1
address-family ipv4
route-target both 1000:1
!
Interface E 0/2
vrf forwarding Cust-A
Ip address 192.1.48.4 255.255.255.0
No shut
```

### Task 3

Configure a static route on R1 in the Cust-A vrf to reach the 10.5.5.0 on R5. Inject this route into BGP such that it should be reachable from Cust-A VRF on R4. Configure a default Route on R5 towards R1.

#### R1

```
ip route vrf Cust-A 10.5.5.0 255.255.255.0 192.1.15.5
!  
Router BGP 1000  
!  
Address-family ipv4 vrf Cust-A  
  Redistribute static
```

#### **R5**

```
ip route 0.0.0.0 0.0.0.0 192.1.15.1
```

#### **Task 4**

Configure a static route on R4 in the Cust-A vrf to reach the 10.8.8.0 on R8. Inject this route into BGP such that it should be reachable from Cust-A VRF on R1. Configure a default Route on R8 towards R4.

#### **R4**

```
ip route vrf Cust-A 10.8.8.0 255.255.255.0 192.1.48.8  
!  
Router BGP 1000  
!  
Address-family ipv4 vrf Cust-A  
  Redistribute static
```

#### **R8**

```
ip route 0.0.0.0 0.0.0.0 192.1.48.4
```

#### **Task 5**

Configure a VRF **Cust-B** with a RD value of 1000:2 on R1 and R4. Use the same extended community for your Route-target import and export. Assign this VRF to the links that connect to Cust-B sites on R1 and R4.

#### **R1**

```
Vrf definition Cust-B  
rd 1000:2  
address-family ipv4  
  route-target both 1000:2  
!  
Interface E 0/2  
Ip vrf forwarding Cust-B  
Ip address 192.1.16.1 255.255.255.0  
No shut
```

#### **R4**

```
Vrf definition Cust-B  
rd 1000:2  
address-family ipv4  
  route-target both 1000:2  
!  
Interface E 0/1  
Ip vrf forwarding Cust-B  
Ip address 192.1.47.4 255.255.255.0  
No shut
```

### Task 6

Configure a static route on R1 in the Cust-B vrf to reach the 10.6.6.0 on R6. Inject this route into BGP such that it should be reachable from Cust-B VRF on R4. Configure a default Route on R6 towards R1.

#### R1

```
ip route vrf Cust-B 10.6.6.0 255.255.255.0 192.1.16.6
!  
Router BGP 1000  
!  
Address-family ipv4 vrf Cust-B  
  Redistribute static
```

#### R6

```
ip route 0.0.0.0 0.0.0.0 192.1.16.1
```

### Task 7

Configure a static route on R4 in the CUST-B vrf to reach the 10.7.7.0 on R7. Inject this route into BGP such that it should be reachable from CUST-B VRF on R1. Configure a default Route on R7 towards R4.

#### R4

```
ip route vrf Cust-B 10.7.7.0 255.255.255.0 192.1.47.7
!  
Router BGP 1000  
!  
Address-family ipv4 vrf Cust-B  
  Redistribute static
```

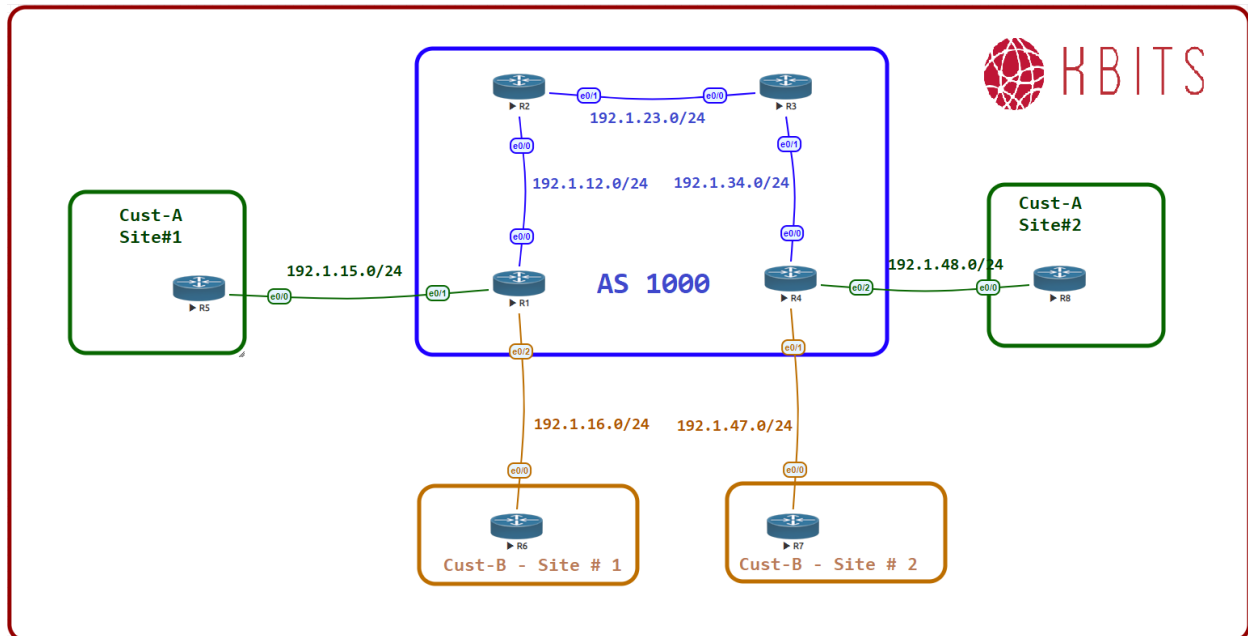
#### R7

```
ip route 0.0.0.0 0.0.0.0 192.1.47.4
```

### NOTE:

Reload the Routers without saving the configs. This will setup the topology for the next lab.

# Lab 4 – Configuring MPLS VPN – PE-CE using EIGRP



**Note:**

Save the Configs on all the routers. **Do not save the configs during the labs.** At the completion of this lab, **reload the routers without saving.** This will allow you to do the next lab based on the same topology.

### Task 1

Configure a VPNv4 neighbor relationship between R1 and R4.

#### R1

```
Router BGP 1000
Neighbor 4.4.4.4 remote-as 1000
Neighbor 4.4.4.4 update-source loopback0
!
Address-family vpnv4
Neighbor 4.4.4.4 activate
```

#### R4

```
Router BGP 1000
Neighbor 1.1.1.1 remote-as 1000
Neighbor 1.1.1.1 update-source loopback0
!
Address-family vpnv4
Neighbor 1.1.1.1 activate
```

### Task 2

Configure a VRF **Cust-A** with a RD value of 1000:1 on R1 and R4. Use the same extended community for your Route-target import and export. Assign this VRF to the links that connect to Cust-A sites on R1 and R4.

#### R1

```
Vrf definition Cust-A
rd 1000:1
address-family ipv4
route-target both 1000:1
!
Interface E 0/1
vrf forwarding Cust-A
Ip address 192.1.15.1 255.255.255.0
No shut
```

#### R4

```
Vrf definition Cust-A
rd 1000:1
address-family ipv4
route-target both 1000:1
!
Interface E 0/2
vrf forwarding Cust-A
Ip address 192.1.48.4 255.255.255.0
No shut
```

### Task 3

Configure EIGRP 100 as the Routing Protocol between R5 and R1-vrf Cust-A. Advertise all the routes on R5 in EIGRP. Advertise the VRF link in EIGRP on R1 under the appropriate address family. Make sure the VRF Cust-A on R4 has reachability to routes learned from R5.

#### R1

```
Router EIGRP 1
!
Address-family ipv4 vrf Cust-A Autonomous-system 100
Network 192.1.15.0
Redistribute BGP 1000 metric 10 10 10 10 10
!
Router BGP 1000
!
Address-family ipv4 vrf Cust-A
Redistribute eigrp 100
```

#### **R5**

```
Router EIGRP 100
Network 192.1.15.0
Network 10.0.0.0
```

#### **Task 4**

Configure EIGRP 100 as the Routing Protocol between R4 and R8-vrf CUST-A. Advertise all the routes on R8 in EIGRP. Advertise the VRF link in RIP on R4 under the appropriate address family. Make sure the VRF CUST-A on R1 has reachability to routes learned from R8.

#### **R4**

```
Router EIGRP 1
!
Address-family ipv4 vrf Cust-A Autonomous-system 100
Network 192.1.48.0
Redistribute BGP 1000 metric 10 10 10 10 10
!
Router BGP 1000
!
Address-family ipv4 vrf Cust-A
Redistribute eigrp 100
```

#### **R8**

```
Router EIGRP 100
Network 192.1.48.0
Network 10.0.0.0
```



### Task 5

Configure a VRF **Cust-B** with a RD value of 1000:2 on R1 and R4. Use the same extended community for your Route-target import and export. Assign this VRF to the links that connect to Cust-B sites on R1 and R4.

R1	R4
<pre>Vrf definition Cust-B rd 1000:2 address-family ipv4  route-target both 1000:2 ! Interface E 0/2 Ip vrf forwarding Cust-B Ip address 192.1.16.1 255.255.255.0 No shut</pre>	<pre>Vrf definition Cust-B rd 1000:2 address-family ipv4  route-target both 1000:2 ! Interface E 0/1 Ip vrf forwarding Cust-B Ip address 192.1.47.4 255.255.255.0 No shut</pre>

### Task 6

Configure EIGRP 200 as the Routing Protocol between R6 and R1-vrf Cust-B. Advertise all the routes on R6 in EIGRP 200. Advertise the VRF link in EIGRP on R1 under the appropriate address family. Make sure the VRF Cust-B on R4 has reachability to routes learned from R6.

<pre>R1 Router EIGRP 1 ! Address-family ipv4 vrf Cust-B Autonomous-system 200 Network 192.1.16.0 Redistribute BGP 100 metric 10 10 10 10 10 ! Router BGP 1000 ! Address-family ipv4 vrf Cust-B Redistribute eigrp 200</pre>
<pre>R6 Router EIGRP 200 Network 192.1.16.0 Network 10.0.0.0</pre>

### Task 7

Configure EIGRP 222 as the Routing Protocol between R7 and R4-vrf Cust-B. Advertise all the routes on R7 in EIGRP 222. Advertise the VRF

link in EIGRP on R4 under the appropriate address family. Make sure the VRF Cust-B on R1 has reachability to routes learned from R7.

#### **R4**

```
Router EIGRP 1
!
Address-family ipv4 vrf Cust-B Autonomous-system 222
Network 192.1.47.0
Redistribute BGP 1000 metric 10 10 10 10 10
!
Router BGP 1000
!
Address-family ipv4 vrf Cust-B
Redistribute eigrp 2222
```

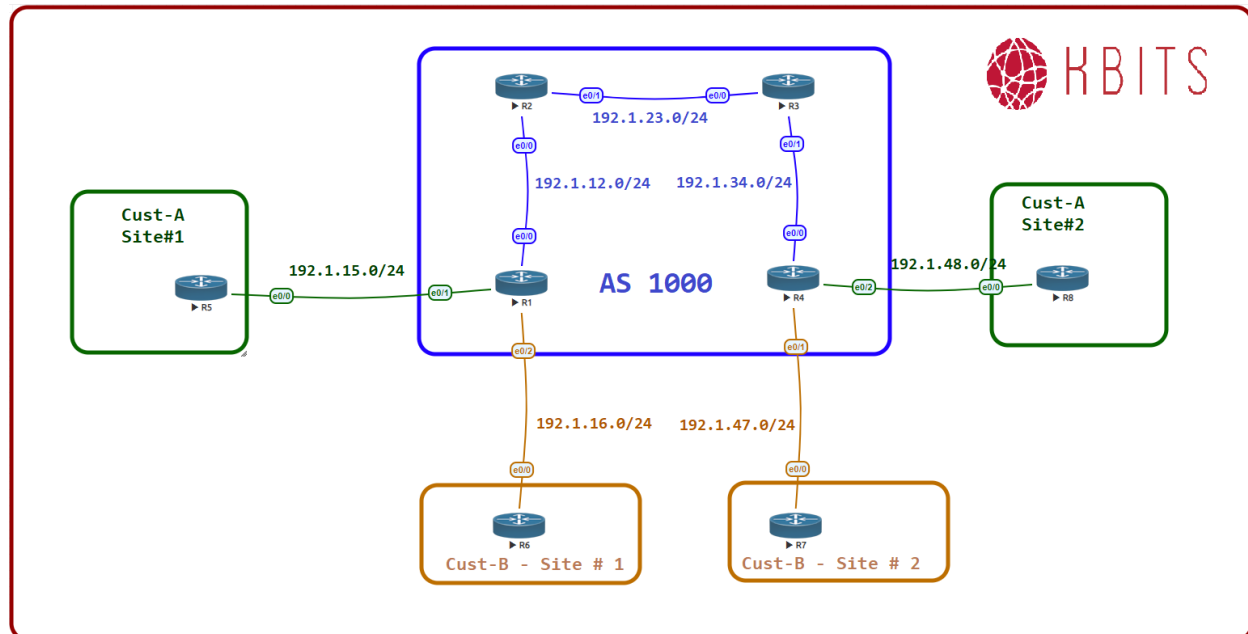
#### **R7**

```
Router EIGRP 222
Network 192.1.47.0
Network 10.0.0.0
```

#### **NOTE:**

Reload the Routers without saving the configs. This will setup the topology for the next lab.

# Lab 5 – Configuring MPLS VPN – PE-CE using BGP – 1



## Note:

Save the Configs on all the routers. **Do not save the configs during the labs.** At the completion of this lab, **reload the routers without saving.** This will allow you to do the next lab based on the same topology.

### Task 1

Configure a VPNv4 neighbor relationship between R1 and R4.

#### R1

```
Router BGP 1000
Neighbor 4.4.4.4 remote-as 1000
Neighbor 4.4.4.4 update-source loopback0
!
Address-family vpnv4
Neighbor 4.4.4.4 activate
```

#### R4

```
Router BGP 1000
Neighbor 1.1.1.1 remote-as 1000
Neighbor 1.1.1.1 update-source loopback0
!
Address-family vpnv4
Neighbor 1.1.1.1 activate
```

### Task 2

Configure a VRF **Cust-A** with a RD value of 1000:1 on R1 and R4. Use the same extended community for your Route-target import and export. Assign this VRF to the links that connect to Cust-A sites on R1 and R4.

#### R1

```
Vrf definition Cust-A
rd 1000:1
address-family ipv4
route-target both 1000:1
!
Interface E 0/1
vrf forwarding Cust-A
Ip address 192.1.15.1 255.255.255.0
No shut
```

#### R4

```
Vrf definition Cust-A
rd 1000:1
address-family ipv4
route-target both 1000:1
!
Interface E 0/2
vrf forwarding Cust-A
Ip address 192.1.48.4 255.255.255.0
No shut
```

### Task 3

Configure BGP as the Routing Protocol between R5 and R1-vrf Cust-A. Advertise all the routes on R5 in BGP. Configure R5 with an AS # of 65005. Configure the BGP neighbor relationship on R1 for the Cust-A VRF. Make sure the VRF Cust-A on R4 has reachability to routes learned from R5.

#### R1

```
Router BGP 1000
!
Address-family ipv4 vrf Cust-A
Neighbor 192.1.15.5 remote-as 65005
```

#### R5

```
Router bgp 65005
Network 10.5.5.0 mask 255.255.255.0
Neighbor 192.1.15.1 remote-as 1000
```

### Task 4

Configure BGP as the Routing Protocol between R8 and R4-vrf Cust-A. Advertise all the routes on R8 in BGP. Configure R8 with an AS # of 65008. Configure the BGP neighbor relationship on R4 for the Cust-A VRF. Make sure the VRF Cust-A on R1 has reachability to routes learned from R8.

#### R4

```
Router BGP 1000
!
Address-family ipv4 vrf Cust-A
Neighbor 192.1.48.8 remote-as 65008
```

#### R8

```
Router 65008
Network 10.8.8.0 mask 255.255.255.0
Neighbor 192.1.48.4 remote-as 1000
```

### Task 5

Configure a VRF **Cust-B** with a RD value of 1000:2 on R1 and R4. Use the same extended community for your Route-target import and export. Assign this VRF to the links that connect to Cust-B sites on R1 and R4.

<b>R1</b>	<b>R4</b>
Vrf definition Cust-B rd 1000:2 address-family ipv4 route-target both 1000:2 ! Interface E 0/2 Ip vrf forwarding Cust-B Ip address 192.1.16.1 255.255.255.0 No shut	Vrf definition Cust-B rd 1000:2 address-family ipv4 route-target both 1000:2 ! Interface E 0/1 Ip vrf forwarding Cust-B Ip address 192.1.47.4 255.255.255.0 No shut

### Task 6

Configure BGP as the Routing Protocol between R6 and R1-vrf Cust-B. Advertise all the routes on R6 in BGP. Configure R6 with an AS # of 65006. Configure the BGP neighbor relationship on R1 for the Cust-B VRF. Make sure the VRF Cust-B on R4 has reachability to routes learned from R6.

<b>R1</b>  Router BGP 1000 ! Address-family ipv4 vrf Cust-B Neighbor 192.1.16.6 remote-as 65006
<b>R6</b>  Router bgp 65006 Network 10.6.6.0 mask 255.255.255.0 Neighbor 192.1.16.1 remote-as 1000

### **Task 7**

Configure BGP as the Routing Protocol between R7 and R4-vrf Cust-B. Advertise all the routes on R7 in BGP. Configure R7 with an AS # of 65007. Configure the BGP neighbor relationship on R4 for the Cust-B VRF. Make sure the VRF Cust-B on R1 has reachability to routes learned from R7.

#### **R4**

```
Router BGP 1000
!  
Address-family ipv4 vrf Cust-B  
Neighbor 192.1.47.7 remote-as 65007
```

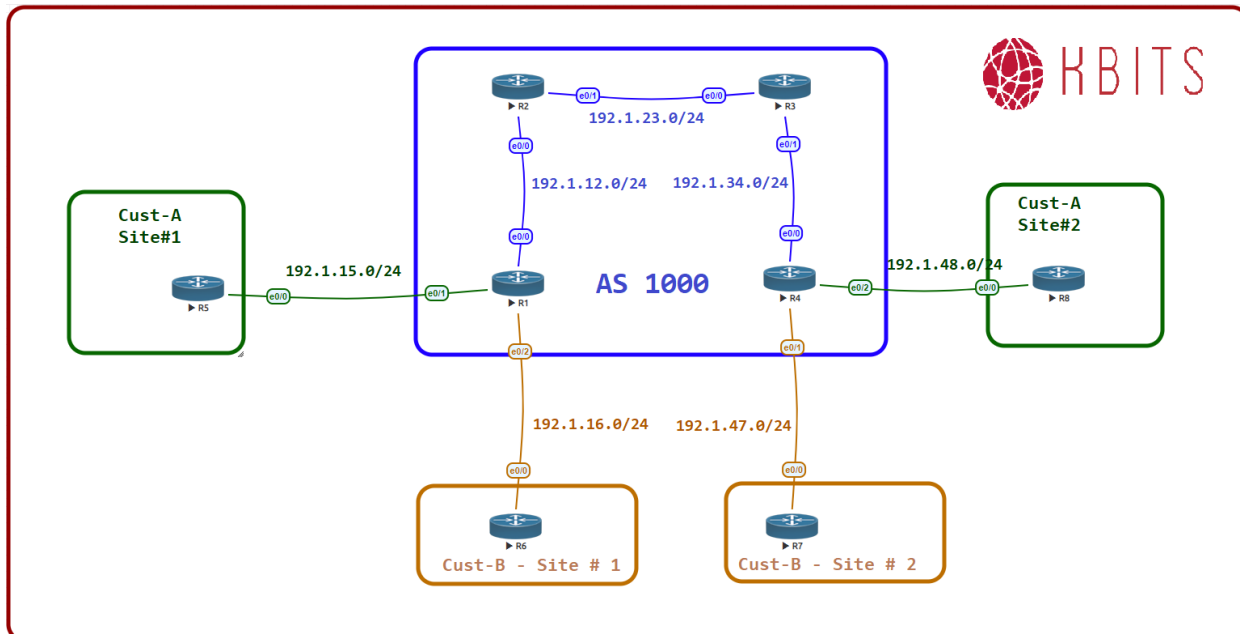
#### **R7**

```
Router bgp 65007  
Network 10.7.7.0 mask 255.255.255.0  
Neighbor 192.1.47.4 remote-as 1000
```

#### **NOTE:**

Reload the Routers without saving the configs. This will setup the topology for the next lab.

# Lab 6 – Configuring MPLS VPN – PE-CE using BGP – 2



**Note:**

Save the Configs on all the routers. **Do not save the configs during the labs.** At the completion of this lab, **reload the routers without saving.** This will allow you to do the next lab based on the same topology.



### Task 1

Configure a VPNv4 neighbor relationship between R1 and R4.

#### R1

```
Router BGP 1000
Neighbor 4.4.4.4 remote-as 1000
Neighbor 4.4.4.4 update-source loopback0
!
Address-family vpnv4
Neighbor 4.4.4.4 activate
```

#### R4

```
Router BGP 1000
Neighbor 1.1.1.1 remote-as 1000
Neighbor 1.1.1.1 update-source loopback0
!
Address-family vpnv4
Neighbor 1.1.1.1 activate
```

### Task 2

Configure a VRF **Cust-A** with a RD value of 1000:1 on R1 and R4. Use the same extended community for your Route-target import and export. Assign this VRF to the links that connect to Cust-A sites on R1 and R4.

#### R1

```
Vrf definition Cust-A
rd 1000:1
address-family ipv4
route-target both 1000:1
!
Interface E 0/1
vrf forwarding Cust-A
Ip address 192.1.15.1 255.255.255.0
No shut
```

#### R4

```
Vrf definition Cust-A
rd 1000:1
address-family ipv4
route-target both 1000:1
!
Interface E 0/2
vrf forwarding Cust-A
Ip address 192.1.48.4 255.255.255.0
No shut
```

### Task 3

Configure BGP as the Routing Protocol between R5 and R1-vrf Cust-A. Advertise all the routes on R5 in BGP. Configure R5 with an AS # of 65001. Configure the BGP neighbor relationship on R1 for the Cust-A VRF. Make sure the VRF Cust-A on R4 has reachability to routes learned from R5.

#### R1

```
Router BGP 1000
!
Address-family ipv4 vrf Cust-A
Neighbor 192.1.15.5 remote-as 65001
```

#### R5

```
Router bgp 65001
Network 10.5.5.0 mask 255.255.255.0
Neighbor 192.1.15.1 remote-as 1000
```

### Task 4

Configure BGP as the Routing Protocol between R8 and R4-vrf Cust-A. Advertise all the routes on R8 in BGP. Configure R8 with an AS # of 65001. Configure the BGP neighbor relationship on R4 for the Cust-A VRF. Make sure the VRF Cust-A on R1 has reachability to routes learned from R8.

#### R4

```
Router BGP 1000
!
Address-family ipv4 vrf Cust-A
Neighbor 192.1.48.8 remote-as 65001
```

#### R8

```
Router 65001
Network 10.8.8.0 mask 255.255.255.0
Neighbor 192.1.48.4 remote-as 1000
```

### Task 5

Configure the PE's (R1 & R4) such that R5 routes are injected into R8's BGP table and vice versa.

#### R1

```
Router BGP 1000
!  
Address-family ipv4 vrf Cust-A  
Neighbor 192.1.15.5 as-override
```

#### R4

```
Router BGP 1000
!  
Address-family ipv4 vrf Cust-A  
Neighbor 192.1.48.8 as-override
```

### Task 6

Configure a VRF **Cust-B** with a RD value of 1000:2 on R1 and R4. Use the same extended community for your Route-target import and export. Assign this VRF to the links that connect to Cust-B sites on R1 and R4.

#### R1

```
Vrf definition Cust-B  
rd 1000:2  
address-family ipv4  
route-target both 1000:2  
!  
Interface E 0/2  
Ip vrf forwarding Cust-B  
Ip address 192.1.16.1 255.255.255.0  
No shut
```

#### R4

```
Vrf definition Cust-B  
rd 1000:2  
address-family ipv4  
route-target both 1000:2  
!  
Interface E 0/1  
Ip vrf forwarding Cust-B  
Ip address 192.1.47.4 255.255.255.0  
No shut
```

### Task 7

Configure BGP as the Routing Protocol between R6 and R1-vrf Cust-B. Advertise all the routes on R6 in BGP. Configure R6 with an AS # of 65002. Configure the BGP neighbor relationship on R1 for the Cust-B VRF. Make sure the VRF Cust-B on R4 has reachability to routes learned from R6.

#### R1

```
Router BGP 1000
!  
Address-family ipv4 vrf Cust-B  
Neighbor 192.1.16.6 remote-as 65002
```

#### R6

```
Router bgp 65002  
Network 10.6.6.0 mask 255.255.255.0  
Neighbor 192.1.16.1 remote-as 1000
```

### Task 8

Configure BGP as the Routing Protocol between R7 and R4-vrf Cust-B. Advertise all the routes on R7 in BGP. Configure R7 with an AS # of 65002. Configure the BGP neighbor relationship on R4 for the Cust-B VRF. Make sure the VRF Cust-B on R1 has reachability to routes learned from R7.

#### R4

```
Router BGP 1000
!  
Address-family ipv4 vrf Cust-B  
Neighbor 192.1.47.7 remote-as 65002
```

#### R7

```
Router bgp 65002  
Network 10.7.7.0 mask 255.255.255.0  
Neighbor 192.1.47.4 remote-as 1000
```

**Task 9**

Configure the CE's (R6 & R7) to allow routes from the remote site to be injected into BGP.

**R6**

```
Router BGP 65002  
Neighbor 192.1.16.1 allowas-in
```

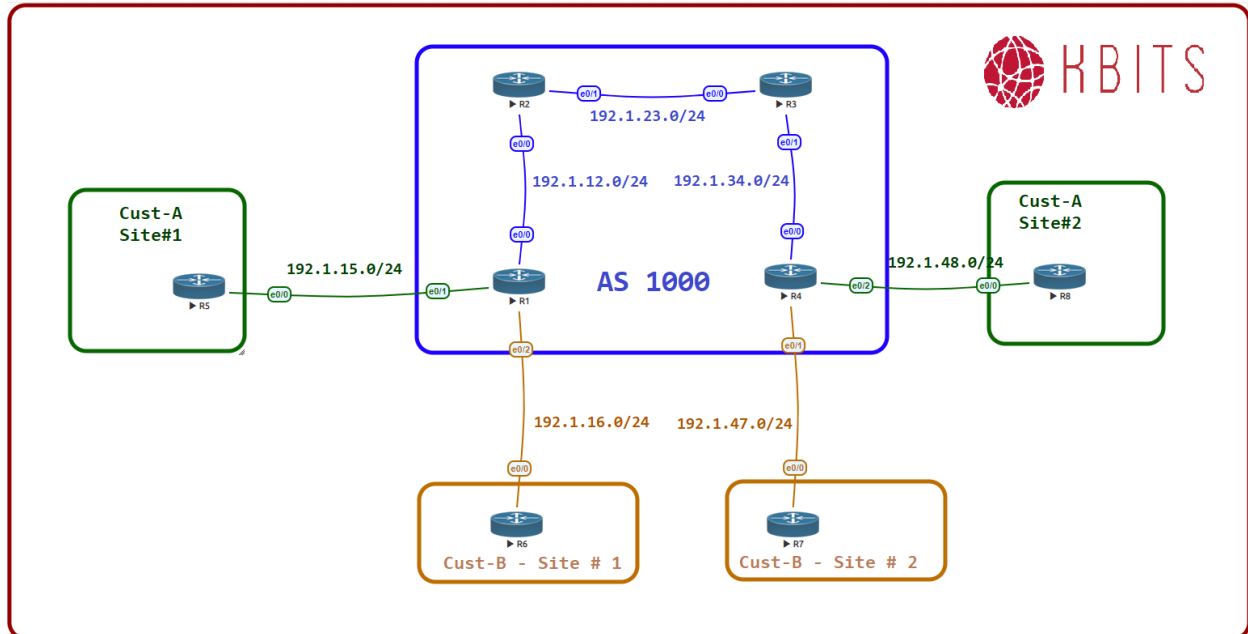
**R7**

```
Router BGP 65002  
Neighbor 192.1.47.4 allowas-in
```

**NOTE:**

Reload the Routers without saving the configs. This will setup the topology for the next lab.

# Lab 7 – Configuring MPLS VPN – PE-CE using OSPF



## Note:

Save the Configs on all the routers. **Do not save the configs during the labs.** At the completion of this lab, **reload the routers without saving.** This will allow you to do the next lab based on the same topology.

### Task 1

Configure a VPNv4 neighbor relationship between R1 and R4.

#### R1

```
Router BGP 1000
Neighbor 4.4.4.4 remote-as 1000
Neighbor 4.4.4.4 update-source loopback0
!
Address-family vpnv4
Neighbor 4.4.4.4 activate
```

#### R4

```
Router BGP 1000
Neighbor 1.1.1.1 remote-as 1000
Neighbor 1.1.1.1 update-source loopback0
!
Address-family vpnv4
Neighbor 1.1.1.1 activate
```

### Task 2

Configure a VRF **Cust-A** with a RD value of 1000:1 on R1 and R4. Use the same extended community for your Route-target import and export. Assign this VRF to the links that connect to Cust-A sites on R1 and R4.

#### R1

```
Vrf definition Cust-A
rd 1000:1
address-family ipv4
route-target both 1000:1
!
Interface E 0/1
vrf forwarding Cust-A
Ip address 192.1.15.1 255.255.255.0
No shut
```

#### R4

```
Vrf definition Cust-A
rd 1000:1
address-family ipv4
route-target both 1000:1
!
Interface E 0/2
vrf forwarding Cust-A
Ip address 192.1.48.4 255.255.255.0
No shut
```

### Task 3

Configure OSPF as the PE-CE Routing protocol in Area 0 between R1 & R5. Advertise all networks on R5 in OSPF. Enable the R1-R5 link on R1 under the Cust-A VRF. Use OSPF process ID 58 on R1. Make sure the VRF Cust-A on R4 has reachability to routes learned from R5.

#### R1

```
Router ospf 58 vrf Cust-A
Network 192.1.15.0 0.0.0.255 area 0
Redistribute bgp 1000
!
Router bgp 1000
Address-family ipv4 vrf Cust-A
Redistribute ospf 58
```

#### R5

```
Router ospf 1
Network 10.5.5.0 0.0.0.255 area 0
Network 192.1.15.0 0.0.0.255 area 0
```

### Task 4

Configure OSPF as the PE-CE Routing protocol in Area 0 between R4 & R8. Advertise all networks on R8 in OSPF. Enable the R4-R8 link on R4 under the Cust-A VRF. Use OSPF process ID 58 on R4. Make sure the VRF Cust-A on R1 has reachability to routes learned from R8.

#### R4

```
Router ospf 58 vrf Cust-A
Network 192.1.48.0 0.0.0.255 area 0
Redistribute bgp 1000
!
Router bgp 1000
Address-family ipv4 vrf Cust-A
Redistribute ospf 58
```

#### R8

```
Router ospf 1
Network 10.8.8.0 0.0.0.255 area 0
Network 192.1.48.0 0.0.0.255 area 0
```



### Task 5

Configure a VRF **Cust-B** with a RD value of 1000:2 on R1 and R4. Use the same extended community for your Route-target import and export. Assign this VRF to the links that connect to Cust-B sites on R1 and R4.

<b>R1</b>  Vrf definition Cust-B rd 1000:2 route-target both 1000:2 ! Interface E 0/2 Ip vrf forwarding Cust-B Ip address 192.1.16.1 255.255.255.0 No shut	<b>R4</b>  Vrf definition Cust-B rd 1000:2 route-target both 1000:2 ! Interface E 0/1 Ip vrf forwarding Cust-B Ip address 192.1.47.4 255.255.255.0 No shut
---	---

### Task 6

Configure OSPF as the PE-CE Routing protocol in Area 0 between R1 & R6. Advertise all networks on R6 in OSPF. Enable the R1-R6 link on R1 under the Cust-B VRF. Use OSPF process ID 6 on R1. Make sure the VRF Cust-B on R4 has reachability to routes learned from R6.

<b>R1</b>  Router ospf 6 vrf Cust-B Network 192.1.16.0 0.0.0.255 area 0 Redistribute bgp 1000 ! Router bgp 1000 Address-family ipv4 vrf Cust-B Redistribute ospf 6	<b>R6</b>  Router ospf 1 Network 10.6.6.0 0.0.0.255 area 0 Network 192.1.16.0 0.0.0.255 area 0
--	--

### Task 7

Configure OSPF as the PE-CE Routing protocol in Area 0 between R4 & R7. Advertise all networks on R7 in OSPF. Enable the R4-R7 link on R4 under the Cust-B VRF. Use OSPF process ID 7 on R4. Make sure the VRF Cust-B on R1 has reachability to routes learned from R7.

#### R4

```
Router ospf 7 vrf Cust-B
Network 192.1.47.0 0.0.0.255 area 0
Redistribute bgp 1000
!
Router bgp 1000
Address-family ipv4 vrf Cust-B
Redistribute ospf 7
```

#### R7

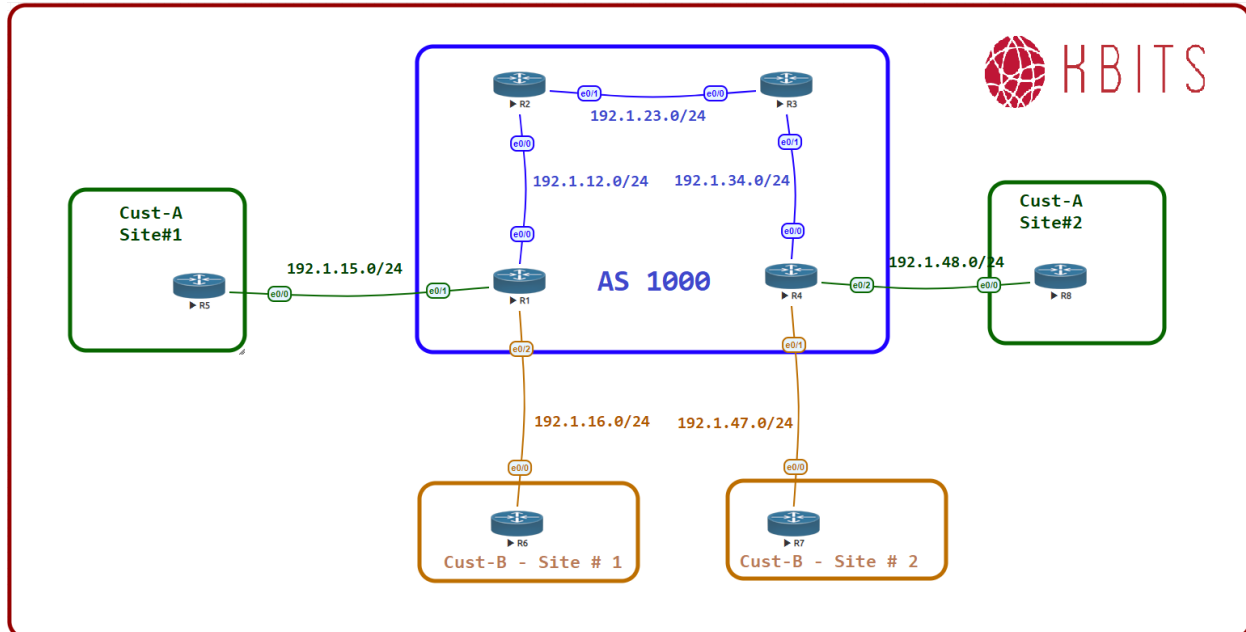
```
Router ospf 1
Network 10.7.7.0 0.0.0.255 area 0
Network 192.1.47.0 0.0.0.255 area 0
```

### NOTE:

For the Cust-A VRF, the OSPF routes from the other site appears as O IA (Inter-Area) routes. This is since PE Routers are using the same process ID (58). The MPLS network is treated as the OSPF Super-Backbone.

For the Cust-B VRF, the OSPF routes from the other site appears as O E2 (External) routes. This is since PE Routers are using different Process ID for the Address Family OSPF process.

# Lab 8 – Configuring MPLS VPN – PE-CE using OSPF – Domain-ID



## Task 1

Configure a Domain-id under OSPF for Cust-B VRF on R1 and R4 as 0.0.0.67 to ensure that OSPF routes are injected as O IA routes on the Customer Routers.

### R1

```
Router ospf 6 vrf Cust-B  
Domain-id 0.0.0.67
```

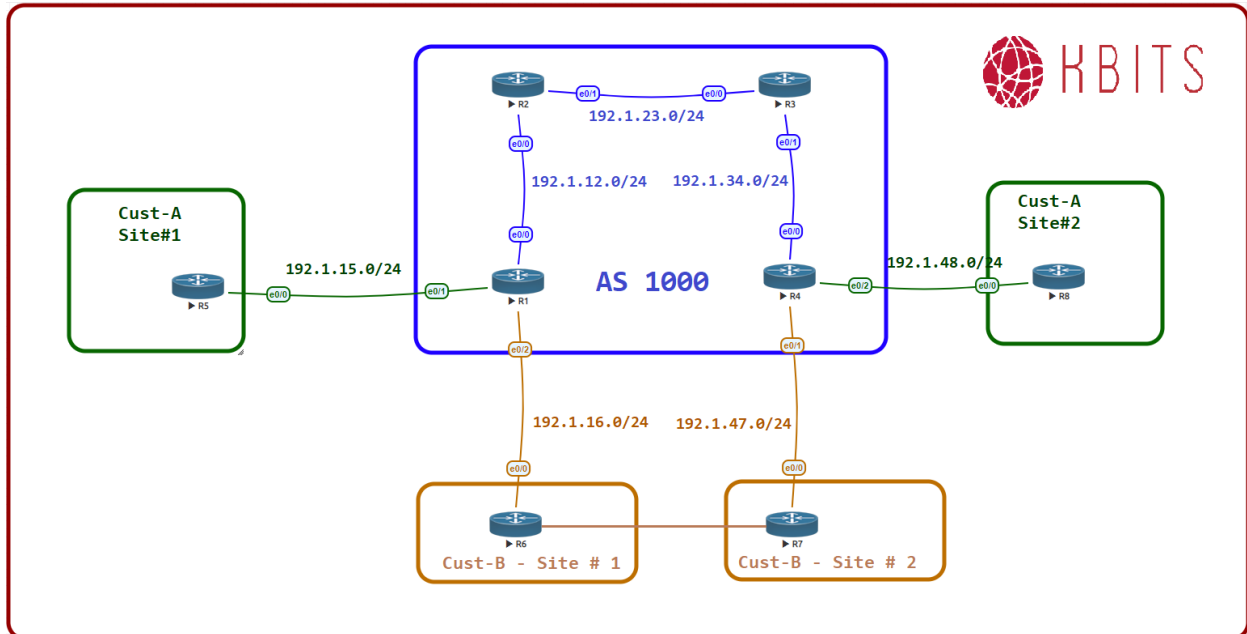
### R4

```
Router ospf 7 vrf Cust-B  
Domain-id 0.0.0.67
```

## NOTE:

For the Cust-B VRF, the OSPF routes from the other site now appear as O IA routes.

# Lab 9 – Configuring MPLS VPN – PE-CE using OSPF – Sham-Link



## Task 1

Configure a Link between R6 and R7 as 10.67.67.0/24. Advertise this link in OSPF. E 0/1 on both routers to connect. As this is a backup (backdoor) link, set the cost on both sides to be 1000.

### R6

```
Interface E 0/1
Ip address 10.67.67.6 255.255.255.0
Ip ospf cost 1000
No shut
!
Router OSPF 1
Network 10.67.67.0 0.0.0.255 area 0
```

### R7

```
Interface E 0/1
Ip address 10.67.67.6 255.255.255.0
Ip ospf cost 1000
No shut
!
Router OSPF 1
Network 10.67.67.0 0.0.0.255 area 0
```

## Task 2

Configure a new loopback each on R1 and R4. This newly created loopback should be part of vrf Cust-B. Advertise this loopback under BGP for the Cust-B vrf. The Loopback information is as follows:

- R1 – Loopback 67 – 172.16.67.1/32
- R4 – Loopback 67 – 172.16.67.4/32

### R1

```
Interface Loopback 67
Ip vrf forwarding Cust-B
Ip address 172.16.67.1 255.255.255.255
!
Router BGP 1000
!
Address-family ipv4 vrf Cust-B
Network 172.16.67.1 mask 255.255.255.255
```

### R4

```
Interface Loopback 67
Ip vrf forwarding Cust-B
Ip address 172.16.67.4 255.255.255.255
!
Router BGP 1000
!
Address-family ipv4 vrf Cust-B
Network 172.16.67.4 mask 255.255.255.255
```

## Task 3

Traffic between Cust-B Sites should be using the new link (Back door) although the cost is much higher than the MPLS cloud. You would like the traffic to go thru the MPLS link instead. Configure a Sham-Link between R1 and R4 based on the new Loopbacks created in the previous step.

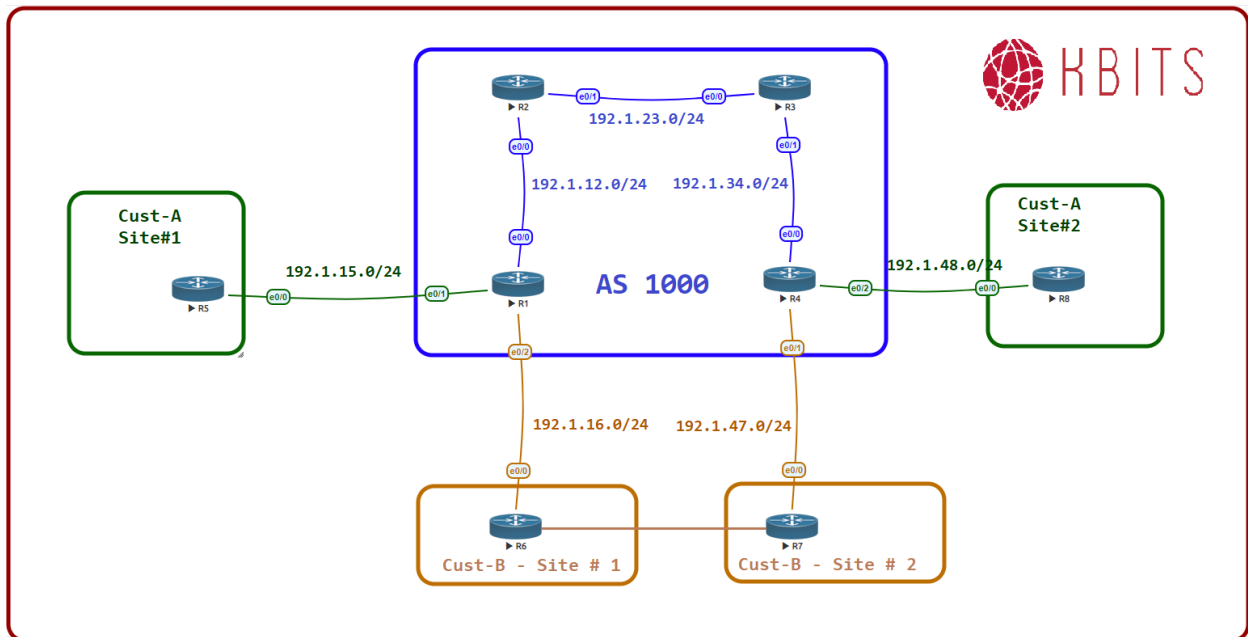
### R1

```
Router ospf 6 vrf Cust-B
area 0 sham-link 172.16.67.1 172.16.67.4
```

### R4

```
Router ospf 8 vrf Cust-B
area 0 sham-link 172.16.67.4 172.16.67.1
```

# Lab 10 – Configuring MPLS VPN Extranets



## Task 1

Configure R1 such that it sets the RT for the 10.5.5.0/24 route in the Cust-A vrf using a Route-Target of 1000:99. These routes will be later imported into Cust-B.

### R1

```
access-list 55 permit 10.5.5.0 0.0.0.255
!  
route-map EM-CustA permit 10  
match ip address 55  
set extcommunity rt 1000:99
```

### Task 2

Configure R1 such that it sets the RT for the 10.6.6.0/24 route in the Cust-B vrf using a Route-Target of 1000:99. These routes will be later imported into Cust-A.

#### R1

```
access-list 66 permit 10.6.6.0 0.0.0.255
!  
route-map EM-CustB permit 10  
match ip address 66  
set extcommunity rt 1000:99
```

### Task 3

Configure R1 Cust-A & Cust-B vrf's to export routes using the Route-map create in the previous steps. Also import the common RT to allow routes to be inter-exchanged between them.

#### R1

```
Vrf definition Cust-A  
Address-family ipv4  
Export map EM-CustA  
Route-target import 1000:99  
!  
Vrf definition Cust-B  
Address-family ipv4  
Export map EM-CustB  
Route-target import 1000:99
```

### Task 4

Configure R4 such that it sets the RT for the 10.8.8.0/24 route in the Cust-A vrf using a Route-Target of 1000:99. These routes will be later imported into Cust-B.

#### R4

```
access-list 88 permit 10.8.8.0 0.0.0.255  
!  
route-map EM-CustA permit 10  
match ip address 88  
set extcommunity rt 1000:99
```

### Task 5

Configure R4 such that it sets the RT for the 10.7.7.0/24 route in the Cust-B vrf using a Route-Target of 1000:99. These routes will be later imported into Cust-A.

#### R4

```
access-list 77 permit 10.7.7.0 0.0.0.255
!  
route-map EM-CustB permit 10  
match ip address 77  
set extcommunity rt 1000:99
```

### Task 6

Configure R4 Cust-A & Cust-B vrf's to export routes using the Route-map create in the previous steps. Also import the common RT to allow routes to be inter-exchanged between them.

#### R4

```
Vrf definition Cust-A  
Address-family ipv4  
Export map EM-CustA  
Route-target import 1000:99  
!  
Vrf definition Cust-B  
Address-family ipv4  
Export map EM-CustB  
Route-target import 1000:99
```

### NOTE:

**Reload the Routers without saving the configs.** This will setup the topology for the next lab.



# Implementing SD-WAN

**Authored By:**

**Khawar Butt**

CCIE # 12353

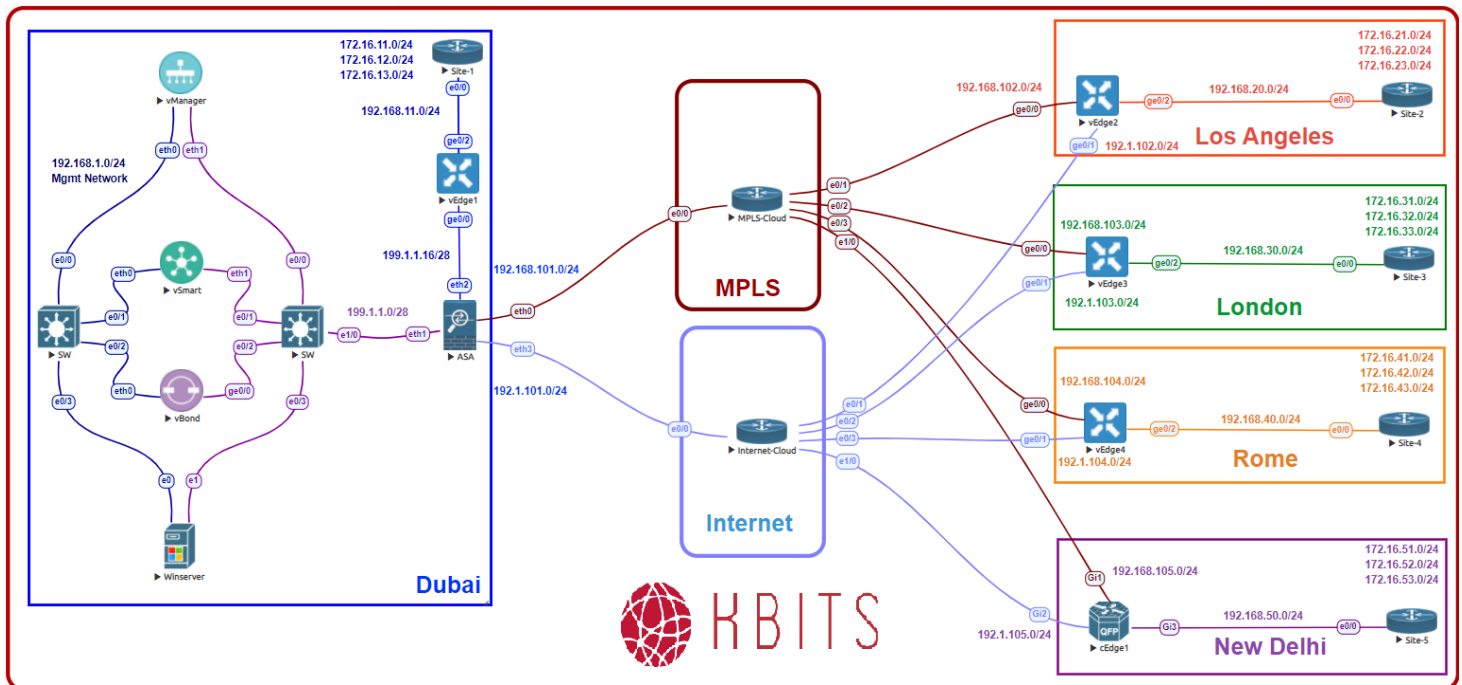
Hepta CCIE#12353

CCDE # 20110020

**Implementing SD-WAN**



# Lab 1 – Configuring the WAN Components



## Interface Configuration

### HQ

Interface	IP Address	Subnet Mask
E 0/0	199.1.1.14	255.255.255.240
E 0/1	199.1.1.30	255.255.255.240
E 0/2	192.168.101.1	255.255.255.0
E 0/3	192.1.101.1	255.255.255.0

### MPLS Cloud

Interface	IP Address	Subnet Mask
E 0/0	192.168.101.254	255.255.255.0
E 0/1	192.168.102.254	255.255.255.0
E 0/2	192.168.103.254	255.255.255.0
E 0/3	192.168.104.254	255.255.255.0
E 1/0	192.168.105.254	255.255.255.0

## Internet Cloud

Interface	IP Address	Subnet Mask
E 0/0	192.1.101.254	255.255.255.0
E 0/1	192.1.102.254	255.255.255.0
E 0/2	192.1.103.254	255.255.255.0
E 0/3	192.1.104.254	255.255.255.0
E 1/0	192.1.105.254	255.255.255.0

## Task 1 – HQ Router Configuration

- Configure the Interfaces based on the Logical Diagram
- Configure OSPF as the IGP to communicate with the MPLS Cloud. Enable all the interfaces.
- Make sure OSPF only sends and receives OSPF packets on the link towards the MPLS Cloud using the Passive-interface command.
- Configure a default route on the router towards the Internet. The IP Address of the Internet Router is 192.1.101.254
- Configure BGP between vEdge1(199.1.1.17) in 65001 and HQ router. Redistribute OPSF into BGP.

### HQ Router

```
Hostname HQ
!
Interface E 0/0
ip address 199.1.1.14 255.255.255.240
no shut
!
Interface E 0/1
ip address 199.1.1.30 255.255.255.240
no shut
!
Interface E 0/2
ip address 192.168.101.1 255.255.255.0
no shut
!
Interface E 0/3
ip address 192.1.101.1 255.255.255.0
no shut
!
router ospf 1
network 192.168.101.0 0.0.0.255 area 0
network 199.1.1.0 0.0.0.255 area 0
passive-interface default
no passive-interface E0/2
!
Router bgp 65001
Neighbor 192.1.1.17 remote-as 65001
Redistribute ospf 1
!
ip route 0.0.0.0 0.0.0.0 192.1.101.254
```

## Task 2 – MPLS Cloud Router Configuration

- Configure the Interfaces based on the Logical Diagram.
- Configure OSPF as the IGP on all the interfaces.

### MPLS Cloud Router

```
no ip domain-lookup
!
line con 0
  exec-timeout 0 0
  logging synchronous
!
hostname MPLS
!
interface Ethernet0/0
  ip address 192.168.101.254 255.255.255.0
  no shut
!
interface Ethernet0/1
  ip address 192.168.102.254 255.255.255.0
  no shut
!
interface Ethernet0/2
  ip address 192.168.103.254 255.255.255.0
  no shut
!
interface Ethernet0/3
  ip address 192.168.104.254 255.255.255.0
  no shut
!
interface Ethernet1/0
  ip address 192.168.105.254 255.255.255.0
  no shut
!
router ospf 1
  network 192.168.101.0 0.0.0.255 area 0
  network 192.168.102.0 0.0.0.255 area 0
  network 192.168.103.0 0.0.0.255 area 0
  network 192.168.104.0 0.0.0.255 area 0
  network 192.168.105.0 0.0.0.255 area 0
```

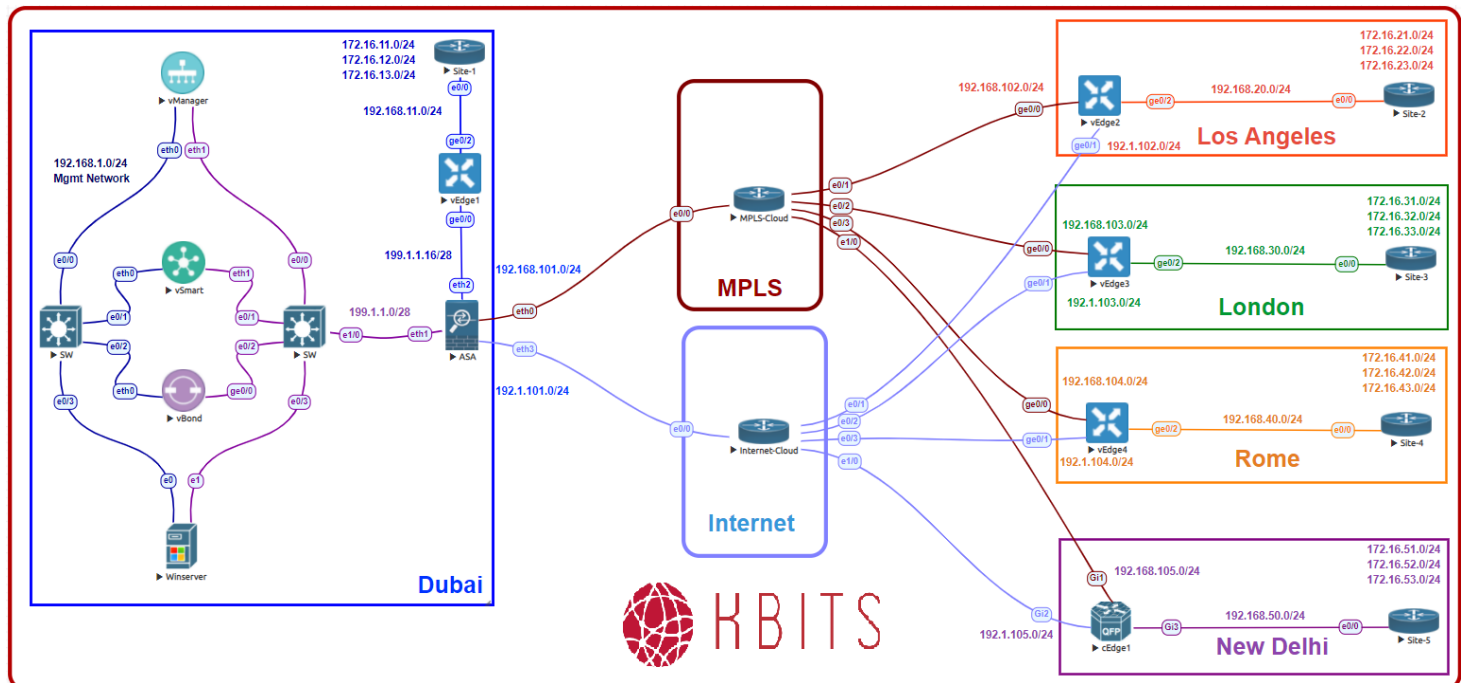
### Task 3 – Internet Cloud Router Configuration

- Configure the Interfaces based on the Logical Diagram
- Configure a Static Route on the Router for the 199.1.1.0/24 network. The Next Hop should point towards the Internet IP of the HQ Router.

#### Internet Cloud Router

```
no ip domain lookup
!
line con 0
  exec-timeout 0 0
  logging synchronous
!
hostname Internet
!
interface Ethernet0/0
  ip address 192.1.101.254 255.255.255.0
  no shut
!
interface Ethernet0/1
  ip address 192.1.102.254 255.255.255.0
  no shut
!
interface Ethernet0/2
  ip address 192.1.103.254 255.255.255.0
  no shut
!
interface Ethernet0/3
  ip address 192.1.104.254 255.255.255.0
  no shut
!
interface Ethernet1/0
  ip address 192.1.105.254 255.255.255.0
  no shut
!
ip route 199.1.1.0 255.255.255.0 192.1.101.1
```

# Lab 2 - Installing the Enterprise Certificate Server



## Task 1 - Configure the Interfaces

### First Ethernet Interface:

IP Address: 192.168.1.5  
Subnet Mask: 255.255.255.0

### Third Ethernet Interface:

IP Address: 199.1.1.5  
Subnet Mask: 255.255.255.240  
Default Gateway: 199.1.1.14

## Task 2 - Configure the Timezone and Time

Configure the appropriate Timezone and Time on the Windows Server.

## Task 3 - Installing the Enterprise Root Certificate Server

- Open **Server Manager**
- Click **Roles**

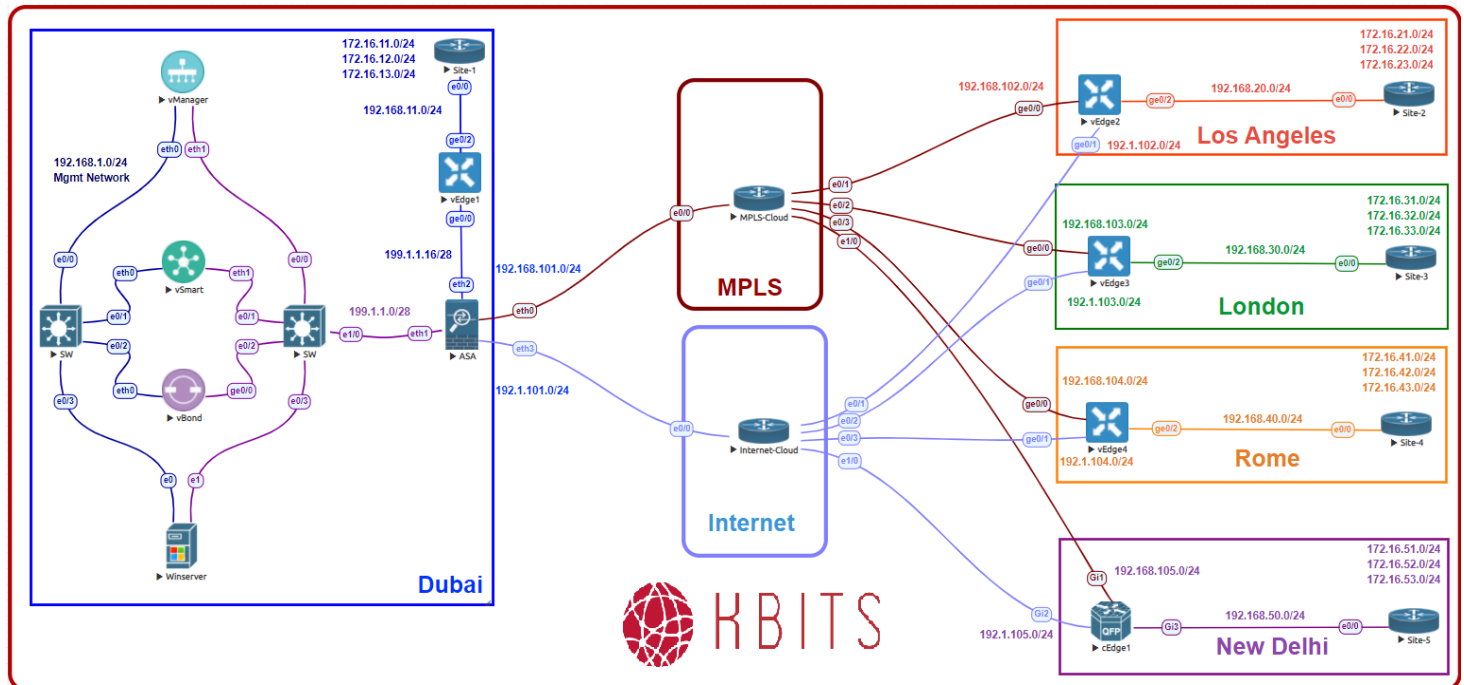
- Click **Add Roles**
- Click **Next**
- Select the "**Active Directory Certificate Services**" and click **Next**
- Click **Next**
- Select "**Certification Authority Web Enrollment**" and click **Next**
- Leave it as Standalone and click **Next**
- Leave it as Root CA and click **Next**
- Leave "Create a new private key" and click **Next**
- Leave the default for the Cryptography for CA and click **Next**
- Set the Common name as **KBITS-CA** and click **Next**
- Leave the default for the Validity Period and click **Next**
- Click **Next**
- Click **Install**

#### **Task 4 – Install WinSCP**

- **Double-click** the WinSCP Installation file.
- Do a Default Installation.



# Lab 3 – Initializing vManage – CLI



## Task 1 – Configuring the System Component

- Configure the System parameters based on the following:
  - Host-name : vManage1
  - Organization: KBITS
  - System-IP: 10.1.1.101
  - Site ID: 1
  - vbond Address: 199.1.1.3
  - Timezone: Based on the appropriate Timezone

**Note: Default username:** admin **Default password:** admin

### vManage

```

config
!
system
 host-name vManage1
 system-ip 10.1.1.101
 site-id 1
 organization-name KBITS
 clock timezone Asia/Muscat
    
```

```
vbond 199.1.1.3
!  
commit
```

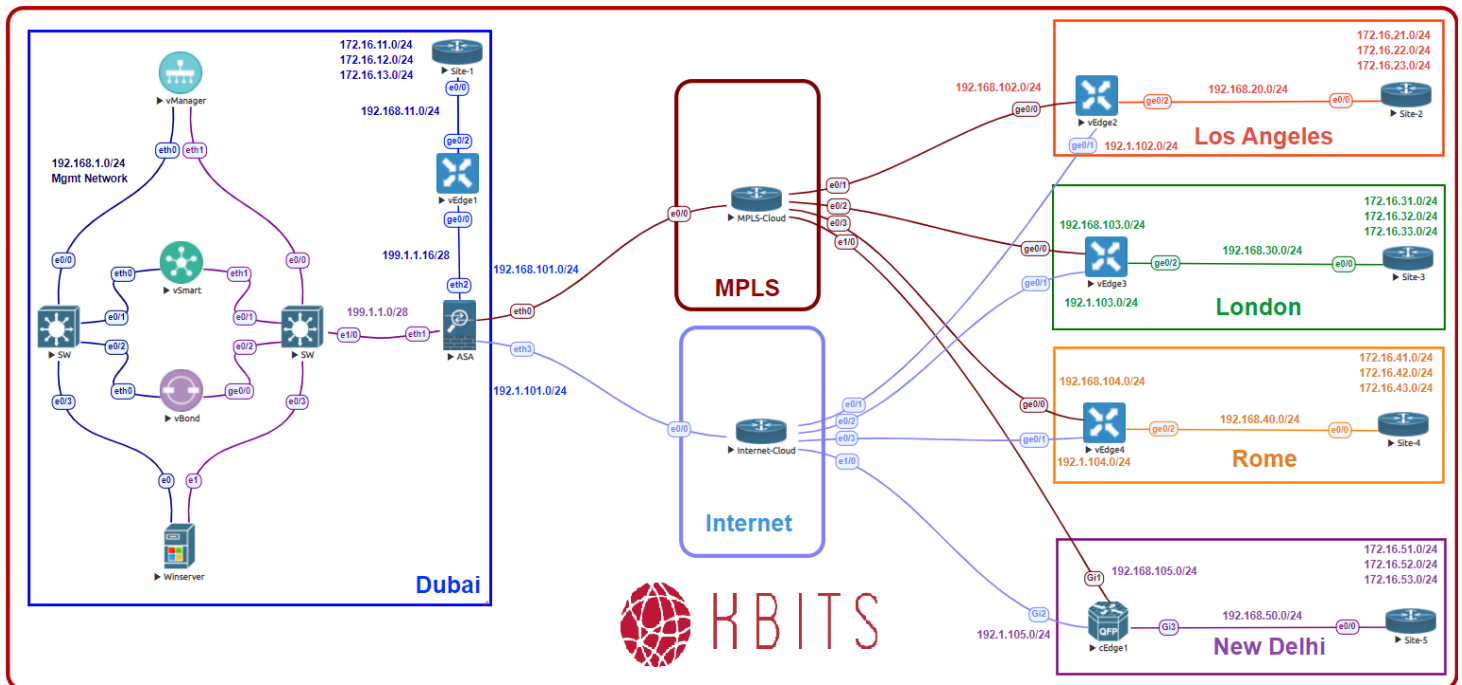
## Task 2 – Configured the vpn parameters

- Configure the VPN parameters based on the following:
  - vpn 0
    - Interface eth1
    - IP Address: 199.1.1.1/28
    - Tunnel Interface
    - Tunnel Services (All, NetConf, SSHD)
    - Default Route: 199.1.1.14
  - vpn 512
    - Interface eth0
    - IP Address: 192.168.1.1/24

## vManage

```
config
!  
vpn 0  
no interface eth0  
interface eth1  
ip address 199.1.1.1/28  
tunnel-interface  
allow-service all  
allow-service netconf  
allow-service sshd  
no shut  
ip route 0.0.0.0/0 199.1.1.14  
!  
vpn 512  
interface eth0  
ip address 192.168.1.1/24  
no shut  
!  
commit
```

# Lab 4 – Initializing vManage - GUI



## Task 1 – Organization name & vBond Address

- Log into the vManage from the Server by browsing to <https://192.168.1.1:8443> using a username of **admin** and a password of **admin**.
- Navigate to **Administration** -> **Settings**
- Click **Edit** on the Organization name and set it to **KBITS**. Confirm the Organization name. Click **OK**.
- Click **Edit** on the **vBond** address and change it to 199.1.1.3. Confirm and click **OK**.

## Task 2 – Configure Controller Authorization as Enterprise Root and Download the Root Certificate.

- Browse to <http://192.168.1.5/certsrv>
- Click **“Download Root Certificate”**.
- Select **“Base 64”**.

- Click “**Download CA Certificate**”.
- Open Explorer and navigate to the downloads folder.
- Change the name of the Downloaded file “**Certnew**” to “**RootCert**”.
- Open the “**RootCert.cer**” file using Notepad.
- Copy using **CTRL-A** and **CTRL-C**.
- In vManage, Navigate to **Administration -> Settings -> Controller Certificate Authorization**.
- Change the “**Certificate Signing by:**” to “**Enterprise Root Certificate**”.
- Paste the RootCert.cer that you had copied by using **CTRL-V**.
- Set the CSR Parameters with the Organization name, City, State, Country. Set the Time to 3 Years and save.

### **Task 3 – Generate a CSR for vManage**

- Navigate to **Configuration -> Certificates -> Controllers -> vManage -> Generate CSR**.
- It will open a window with the CSR. Copy by using **CTRL-A** and **CTRL-C**.

### **Task 4 – Request a Certificate from the CA Server**

- Browse to <http://192.168.1.5/certsrv>
- Click “**Request a Certificate**”.
- Select “**Advanced**”.
- Paste the CSR in the box by using **CTRL-V** and click **Submit**.

### **Task 5 – Issue the Certificate from the CA Server**

- Open Server Manager and navigate to **Active Directory Certificate Server -> KBITS-CA -> Pending Requests**.

- Right-Click the request and click **“Issue”**.

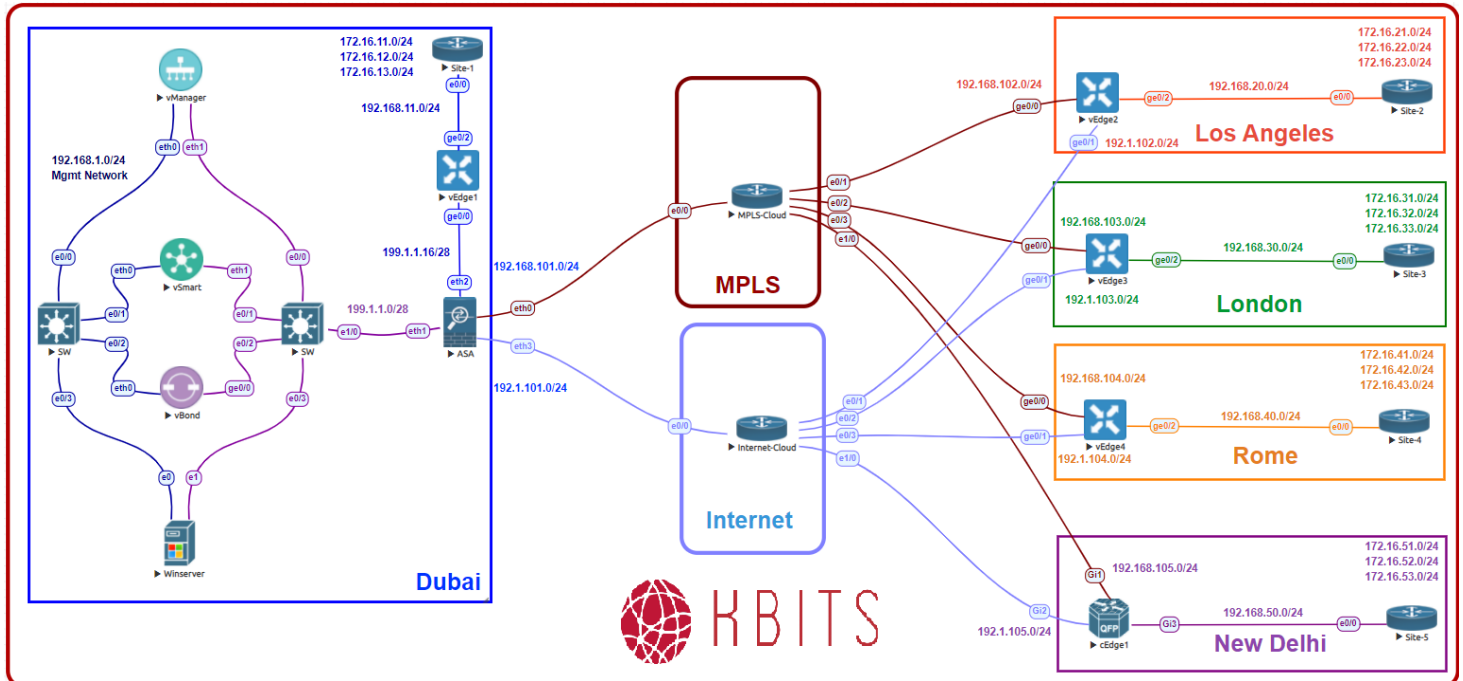
### **Task 6 – Downloading the Issued Certificate**

- Browse to <http://192.168.1.5/certsrv>
- Click **“Check on Pending request”**.
- The issued certificate link will show up. Click on the link.
- Select **“Base 64”** and click **“Download”**
- Open Explorer and navigate to the downloads folder.
- Change the name of the Downloaded file **“Certnew”** to **“vManage”**.
- Open the **“vManage.cer”** file using Notepad.
- Copy using **CTRL-A** and **CTRL-C**.

### **Task 6 – Installing the Identity Certificate for vManage**

- In vManage, Navigate to **Configuration -> Certificates -> Controllers**
- Click on the **“Install”** button at the top right corner
- Paste the Certificate (CTRL-V).
- The Identity certificate should be installed on vManage.

# Lab 5 - Initializing vBond - CLI



## Task 1 - Configuring the System Component

- Configure the System parameters based on the following:
  - Host-name : vBond1
  - Organization: KBITS
  - System-IP: 10.1.1.103
  - Site ID: 1
  - vbond Address: 199.1.1.3
  - Timezone: Based on the appropriate Timezone

**Note: Default username:** admin **Default password:** admin

### vBond

```

config
!
system
host-name vBond1
system-ip 10.1.1.103
site-id 1
organization-name KBITS
clock timezone Asia/Muscat
vbond 199.1.1.3 local
    
```

```
!  
Commit
```

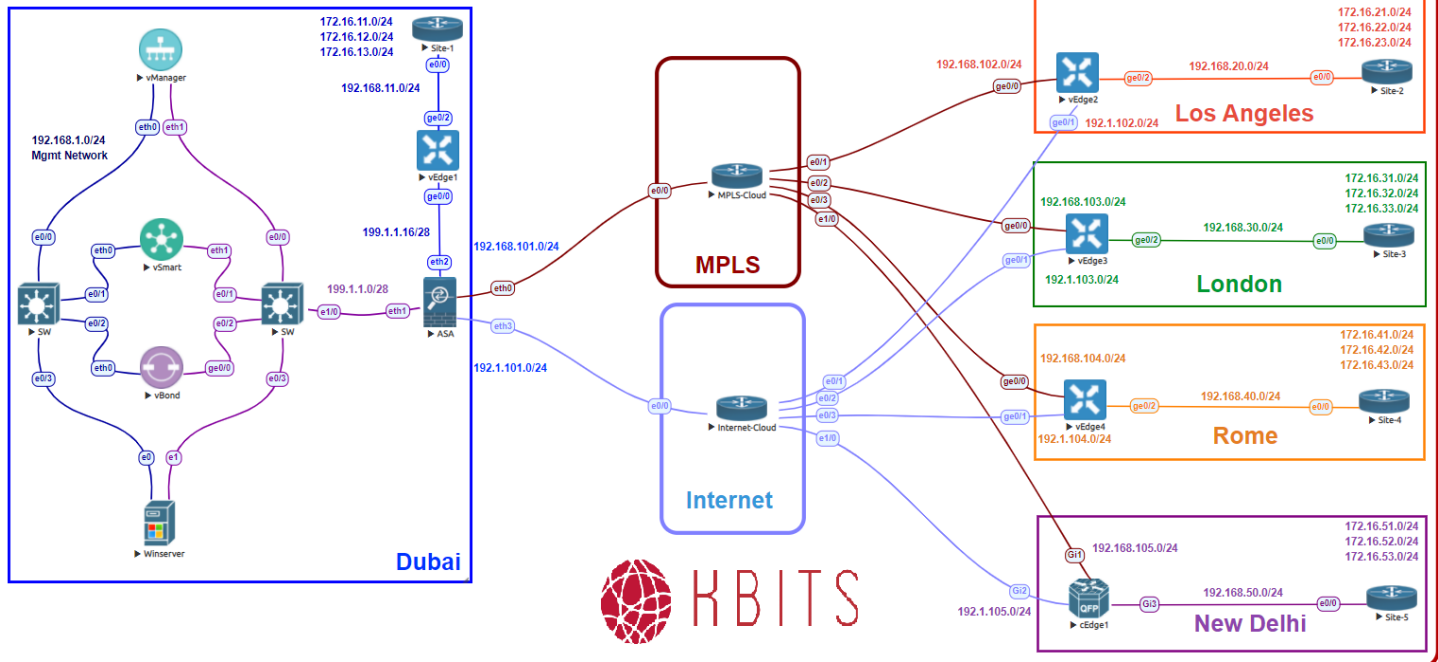
## Task 2 – Configured the vpn parameters

- Configure the VPN parameters based on the following:
  - vpn 0
    - Interface ge0/0
    - IP Address: 199.1.1.3/28
    - Tunnel Interface
    - Tunnel Services (All, NetConf, SSHD)
    - Encapsulation: IPsec
    - Default Route: 199.1.1.14
  - vpn 512
    - Interface eth0
    - IP Address: 192.168.1.3/24

### vBond

```
config  
!  
vpn 0  
no interface eth0  
interface ge0/0  
ip address 199.1.1.3/28  
tunnel-interface  
encapsulation ipsec  
allow-service all  
allow-service netconf  
allow-service sshd  
no shut  
ip route 0.0.0.0/0 199.1.1.14  
!  
vpn 512  
interface eth0  
ip address 192.168.1.3/24  
no shut  
!  
commit
```

# Lab 6 – Initializing vBond - GUI



## Task 1 – Add vBond to vManage

- Navigate to **Configuration -> Devices -> Controllers -> Add Controllers -> vBond** and specify the following to add the vBond in vManage.
  - IP Address: **199.1.1.3**
  - Username: **Admin**
  - Password: **Admin**
  - Check Generate CSR
  - Click **OK**

## Task 2 – View the generated CSR for vBond and Copy it

- Navigate to **Configuration -> Certificates -> Controllers -> vBond -> View CSR**.
- It will open a window with the CSR. Copy by using **CTRL-A** and **CTRL-C**.

## Task 3 – Request a Certificate from the CA Server

- Browse to <http://192.168.1.5/certsrv>



- Click **“Request a Certificate”**.
- Select **“Advanced”**.
- Paste the CSR in the box by using **CTRL-V** and click **Submit**.

#### **Task 4 – Issue the Certificate from the CA Server**

- Open Server Manager and navigate to **Active Directory Certificate Server -> KBITS-CA -> Pending Requests**.
- Right-Click the request and click **“Issue”**.

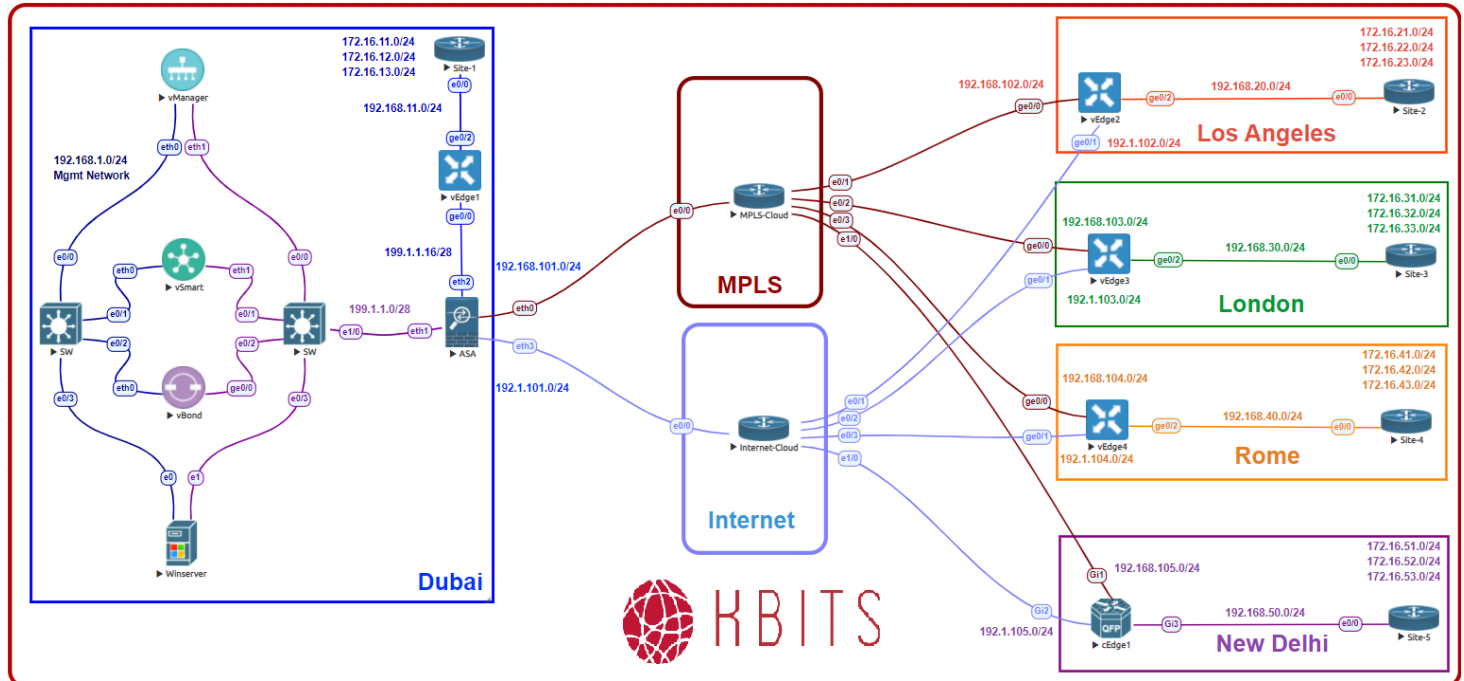
#### **Task 6 – Downloading the Issued Certificate**

- Browse to <http://192.168.1.5/certsrv>
- Click **“Check on Pending request”**.
- The issued certificate link will show up. Click on the link.
- Select **“Base 64”** and click **“Download”**
- Open Explorer and navigate to the downloads folder.
- Change the name of the Downloaded file **“Certnew”** to **“vBond”**.
- Open the **“vBond.cer”** file using Notepad.
- Copy using **CTRL-A** and **CTRL-C**.

#### **Task 6 – Installing the Identity Certificate for vManage**

- In vManage, Navigate to **Configuration -> Certificates -> Controllers**
- Click on the **“Install”** button at the top right corner
- Paste the Certificate (CTRL-V).
- The Identity certificate should be installed for vBond and pushed to it.

# Lab 7 – Initializing vSmart – CLI



## Task 1 – Configuring the System Component

- Configure the System parameters based on the following:
  - Host-name : vSmart1
  - Organization: KBITS
  - System-IP: 10.1.1.102
  - Site ID: 1
  - vbond Address: 199.1.1.3
  - Timezone: Based on the appropriate Timezone

**Note: Default username:** admin **Default password:** admin

### vSmart

```

config
!
system
host-name vSmart1
system-ip 10.1.1.102
site-id 1
organization-name KBITS
clock timezone Asia/Muscat
vbond 199.1.1.3
    
```

```
!  
Commit
```

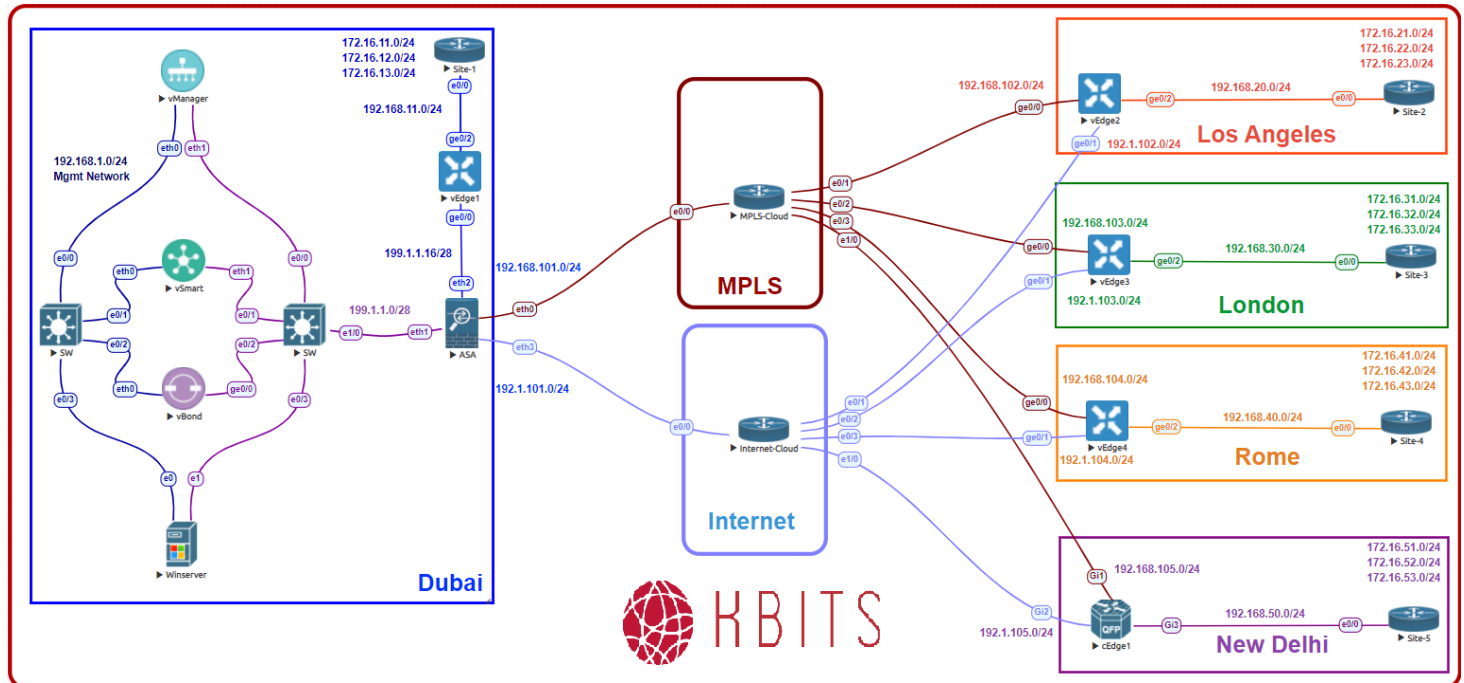
## Task 2 – Configured the vpn parameters

- Configure the VPN parameters based on the following:
  - vpn 0
    - Interface Eth1
    - IP Address: 199.1.1.2/28
    - Tunnel Interface
    - Tunnel Services (All, NetConf, SSHD)
    - Default Route: 199.1.1.14
  - vpn 512
    - Interface eth0
    - IP Address: 192.168.1.2/24

### vSmart

```
config  
!  
vpn 0  
no interface eth0  
interface eth1  
ip address 199.1.1.2/28  
tunnel-interface  
allow-service all  
allow-service netconf  
allow-service sshd  
no shut  
ip route 0.0.0.0/0 199.1.1.14  
!  
vpn 512  
interface eth0  
ip address 192.168.1.2/24  
no shut  
!  
Commit
```

# Lab 8 – Initializing vSmart - GUI



## Task 1 – Add vSmart to vManage

- Navigate to **Configuration -> Devices -> Controllers -> Add Controllers -> vSmart** and specify the following to add the vBond in vManage.
  - IP Address: **199.1.1.2**
  - Username: **Admin**
  - Password: **Admin**
  - Check Generate CSR
  - Click **OK**

## Task 2 – View the generated CSR for vSmart and Copy it

- Navigate to **Configuration -> Certificates -> Controllers -> vSmart -> View CSR**.
- It will open a window with the CSR. Copy by using **CTRL-A** and **CTRL-C**.

## Task 3 – Request a Certificate from the CA Server

- Browse to <http://192.168.1.5/certsrv>

- Click **“Request a Certificate”**.
- Select **“Advanced”**.
- Paste the CSR in the box by using **CTRL-V** and click **Submit**.

#### **Task 4 – Issue the Certificate from the CA Server**

- Open Server Manager and navigate to **Active Directory Certificate Server -> KBITS-CA -> Pending Requests**.
- Right-Click the request and click **“Issue”**.

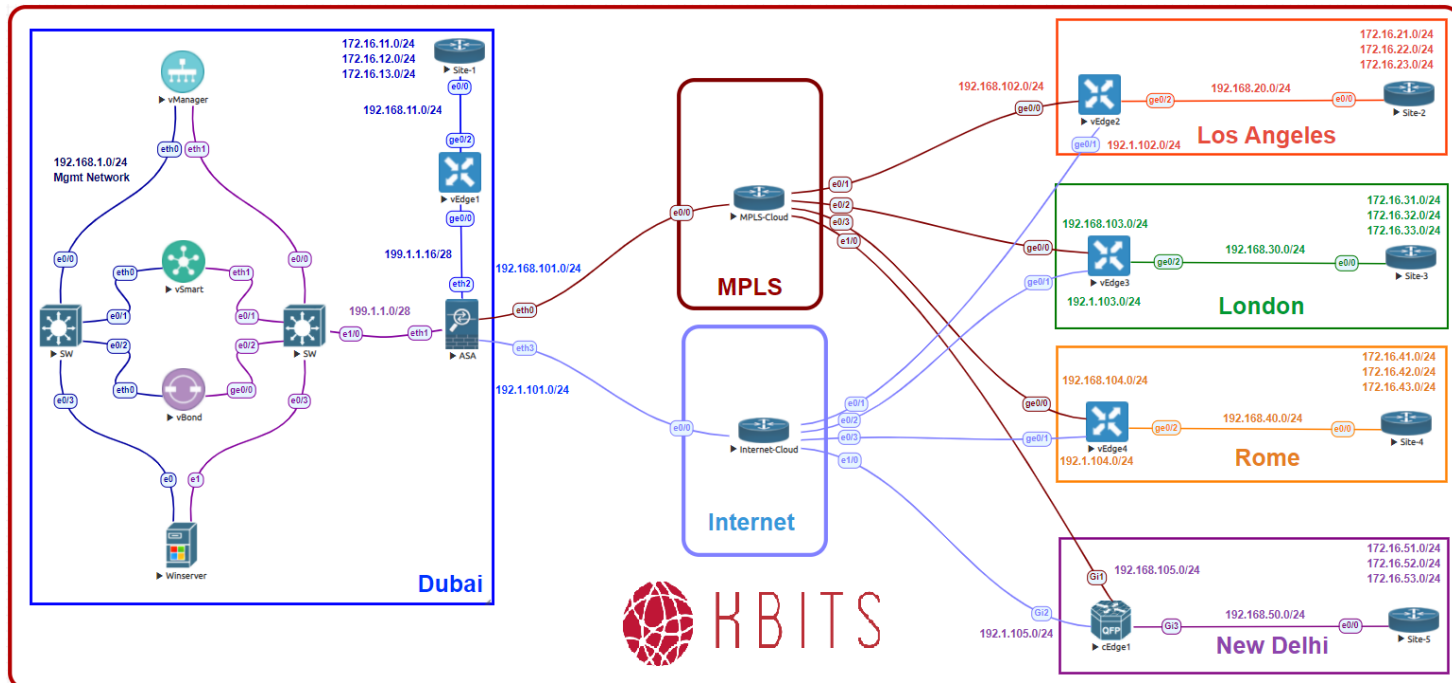
#### **Task 6 – Downloading the Issued Certificate**

- Browse to <http://192.168.1.5/certsrv>
- Click **“Check on Pending request”**.
- The issued certificate link will show up. Click on the link.
- Select **“Base 64”** and click **“Download”**
- Open Explorer and navigate to the downloads folder.
- Change the name of the Downloaded file **“Certnew”** to **“vSmart”**.
- Open the **“vSmart.cer”** file using Notepad.
- Copy using **CTRL-A** and **CTRL-C**.

#### **Task 6 – Installing the Identity Certificate for vManage**

- In vManage, Navigate to **Configuration -> Certificates -> Controllers**
- Click on the **“Install”** button at the top right corner
- Paste the Certificate (CTRL-V).
- The Identity certificate should be installed for vSmart and pushed to it.

# Lab 9 – Initializing vEdge – CLI



## Task 1 – Upload the WAN Edge List

- On the vManage Main windows, Navigate to **Configuration -> Devices**. Click on “**Upload WAN Edge List**”.
- Select the file you downloaded from the PNP Portal. Upload it and check the **Validate** option.

## vEDGE-1

### Task 1 – Configuring the System Component

- Configure the System parameters based on the following:
  - Host-name : vEdge1
  - Organization: KBITS
  - System-IP: 10.2.2.201
  - Site ID: 1
  - vbond Address: 199.1.1.3
  - Timezone: Based on the appropriate Timezone

**Note: Default username:** admin **Default password:** admin

### **vEdge1**

```
config
!  
system
  host-name vEdge1
  system-ip 10.2.2.201
  site-id 1
  organization-name KBITS
  clock timezone Asia/Muscat
  vbond 199.1.1.3
!  
commit
```

### **Task 2 – Configure the vpn parameters**

- Configure the VPN parameters based on the following:
  - vpn 0
    - Interface ge0/0
    - IP Address: 199.1.1.17/28
    - Tunnel Interface
    - Encapsulation IPsec
    - Tunnel Services (All, NetConf, SSHD)
    - Default Route: 199.1.1.30
  - vpn 512
    - Interface eth0
    - IP Address: DHCP Client

### **vEdge1**

```
config
!  
vpn 0
  no interface eth0
  interface ge0/0
  ip address 199.1.1.17/28
  tunnel-interface
  encapsulation ipsec
  allow-service all
  allow-service netconf
  allow-service sshd
```

```
no shut
ip route 0.0.0.0/0 199.1.1.30
!
vpn 512
interface eth0
 ip dhcp-client
 no shutdown
commit
```

## vEDGE-2

### Task 1 - Configuring the System Component

- Configure the System parameters based on the following:
  - Host-name : vEdge2
  - Organization: KBITS
  - System-IP: 10.2.2.202
  - Site ID: 2
  - vbond Address: 199.1.1.3
  - Timezone: Based on the appropriate Timezone

**Note: Default username:** admin **Default password:** admin

```
vEdge-2
config
!
system
 host-name vEdge2
 system-ip 10.2.2.202
 site-id 2
 organization-name KBITS
 clock timezone Asia/Muscat
 vbond 199.1.1.3
!
commit
```

### Task 2 - Configure the vpn parameters

- Configure the VPN parameters based on the following:
  - vpn 0
    - Interface ge0/0



- IP Address: 192.168.102.2/24
  - Tunnel Interface
  - Encapsulation IPSec
  - Tunnel Services (All, NetConf, SSHD)
  - Default Route: 192.168.102.254
- vpn 512
    - Interface eth0
    - IP Address: DHCP Client

## **vEdge2**

```

config
!
vpn 0
no interface eth0
interface ge0/0
ip address 192.168.102.2/24
tunnel-interface
encapsulation ipsec
allow-service all
allow-service netconf
allow-service sshd
no shut
ip route 0.0.0.0/0 192.168.102.254
!
vpn 512
interface eth0
ip dhcp-client
no shutdown
!
commit

```

## **vEDGE-3**

### **Task 1 – Configuring the System Component**

- Configure the System parameters based on the following:
  - Host-name : vEdge3
  - Organization: KBITS
  - System-IP: 10.2.2.203
  - Site ID: 3
  - vbond Address: 199.1.1.3

- Timezone: Based on the appropriate Timezone

**Note: Default username:** admin **Default password:** admin

### **vEdge-3**

```
config
!
system
host-name vEdge3
system-ip 10.2.2.203
site-id 3
organization-name KBITS
clock timezone Asia/Muscat
vbond 199.1.1.3
!
Commit
```

### **Task 2 – Configure the vpn parameters**

- Configure the VPN parameters based on the following:
  - vpn 0
    - Interface ge0/0
    - IP Address: 192.168.103.3/24
    - Tunnel Interface
    - Encapsulation IPsec
    - Tunnel Services (All, NetConf, SSHD)
    - Default Route: 192.168.103.254
  - vpn 512
    - Interface eth0
    - IP Address: DHCP Client

### **vEdge3**

```
config
!
vpn 0
no interface eth0
interface ge0/0
ip address 192.168.103.3/24
tunnel-interface
encapsulation ipsec
allow-service all
allow-service netconf
```

```
allow-service sshd
no shut
ip route 0.0.0.0/0 192.168.103.254
!
vpn 512
interface eth0
ip dhcp-client
no shutdown
!
Commit
```

## vEDGE-4

### Task 1 – Configuring the System Component

- Configure the System parameters based on the following:
  - Host-name : vEdge4
  - Organization: KBITS
  - System-IP: 10.2.2.204
  - Site ID: 4
  - vbond Address: 199.1.1.3
  - Timezone: Based on the appropriate Timezone

**Note:** Default username: admin Default password: admin

### vEdge-4

```
config
!
system
host-name vEdge4
system-ip 10.2.2.204
site-id 4
organization-name KBITS
clock timezone Asia/Muscat
vbond 199.1.1.3
!
Commit
```

### Task 2 – Configure the vpn parameters

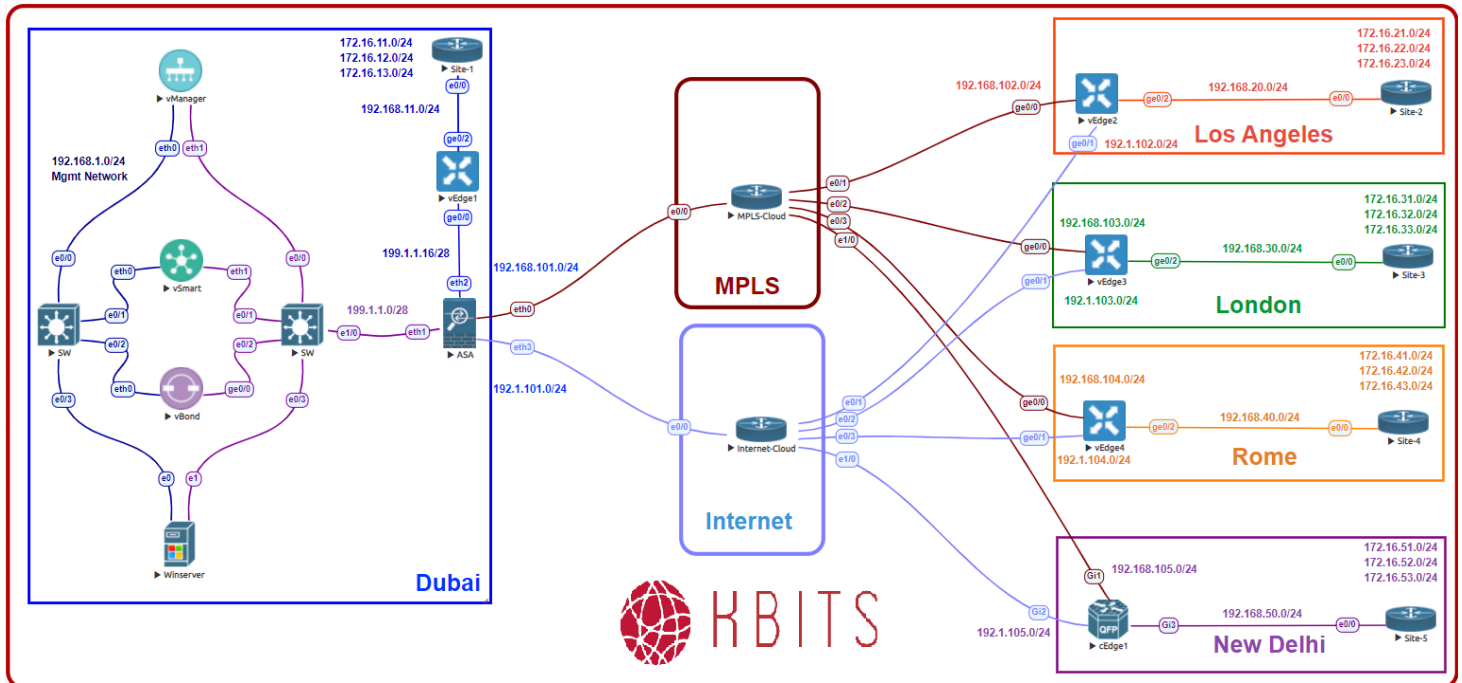
- Configure the VPN parameters based on the following:

- vpn 0
  - Interface ge0/0
  - IP Address: 192.168.104.4/24
  - Tunnel Interface
  - Encapsulation IPsec
  - Tunnel Services (All, NetConf, SSHD)
  - Default Route: 192.168.104.254
  
- vpn 512
  - Interface eth0
  - IP Address: DHCP Client

#### **vEdge4**

```
config
!
vpn 0
no interface eth0
interface ge0/0
ip address 192.168.104.4/24
tunnel-interface
encapsulation ipsec
allow-service all
allow-service netconf
allow-service sshd
no shut
ip route 0.0.0.0/0 192.168.104.254
!
vpn 512
interface eth0
ip dhcp-client
no shutdown
!
Commit
```

# Lab 10 – Registering vEdges in vManage



## vEDGE-1

### Task 1 – Upload the Root Certificate to the vEdge

- On the Windows Server, open **WINSCP** application.
- **Connect** to vEdge1 using the following information:
  - IP Address : 199.1.1.17
  - Protocol - SFTP
  - Username : admin
  - Password : admin
- Copy the **RootCert.cer** file from the Downloads folder to the **/home/admin** folder on the vEdge1

### Task 2 – Install the Root Certificate on vEdge1

- Connect to the console of vEdge1 and issue the following command:

```
request root-cert-chain install /home/admin/RootCert.cer
```

### Task 3 - Activate vEdge on vManage

- Navigate to **Configuration -> Devices**
- Note and use the **Chassis Number** and **Token number** for the 1st vEdge from vManage.
- Use the information from the previous step in the following command on the vEdge1 console.

```
request vedge-cloud activate chassis-number XXXXXXXX-XXXX-  
XXXX-XXXX-XXXXXXXXXXXX token  
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

- You should see the vEdge in the vManage console with a Certificate issued.

## vEDGE-2

### Task 1 – Upload the Root Certificate to the vEdge

- On the Windows Server, open **WINSCP** application.
- **Connect** to vEdge1 using the following information:
  - IP Address : 192.168.102.2
  - Protocol - SFTP
  - Username : admin
  - Password : admin
- Copy the **RootCert.cer** file from the Downloads folder to the **/home/admin** folder on the vEdge2

### Task 2 – Install the Root Certificate on vEdge2

- Connect to the console of vEdge2 and issue the following command:

```
request root-cert-chain install /home/admin/RootCert.cer
```

### Task 3 - Activate vEdge on vManage

- Navigate to **Configuration -> Devices**
- Note and use the **Chassis Number** and **Token number** for the 2<sup>nd</sup> vEdge from vManage.
- Use the information from the previous step in the following command on the vEdge2 console.

```
request vedge-cloud activate chassis-number XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX token  
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

- You should see the vEdge in the vManage console with a Certificate issued.

## vEDGE-3

### Task 1 – Upload the Root Certificate to the vEdge

- On the Windows Server, open **WINSCP** application.
- **Connect** to vEdge1 using the following information:
  - IP Address : 192.168.103.3
  - Protocol - SFTP
  - Username : admin
  - Password : admin
- Copy the **RootCert.cer** file from the Downloads folder to the **/home/admin** folder on the vEdge3

### Task 2 – Install the Root Certificate on vEdge3

- Connect to the console of vEdge3 and issue the following command:

```
request root-cert-chain install /home/admin/RootCert.cer
```

### Task 3 - Activate vEdge on vManage

- Navigate to **Configuration -> Devices**
- Note and use the **Chassis Number** and **Token number** for the 3<sup>rd</sup> vEdge from vManage.
- Use the information from the previous step in the following command on the vEdge3 console.

```
request vedge-cloud activate chassis-number XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX token  
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

- You should see the vEdge in the vManage console with a Certificate issued.



## vEDGE-4

### Task 1 – Upload the Root Certificate to the vEdge

- On the Windows Server, open **WINSCP** application.
- **Connect** to vEdge1 using the following information:
  - IP Address : 192.168.104.4
  - Protocol - SFTP
  - Username : admin
  - Password : admin
- Copy the **RootCert.cer** file from the Downloads folder to the **/home/admin** folder on the vEdge4

### Task 2 – Install the Root Certificate on vEdge4

- Connect to the console of vEdge4 and issue the following command:

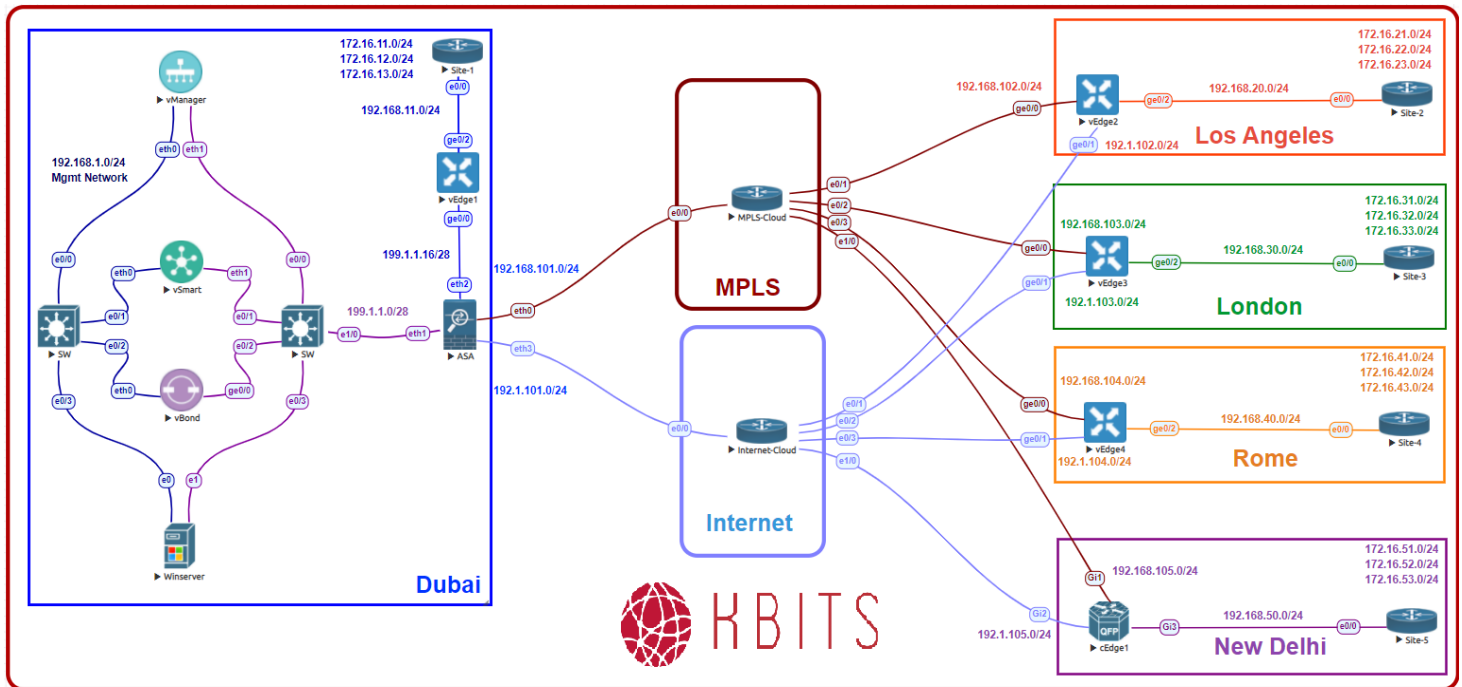
```
request root-cert-chain install /home/admin/RootCert.cer
```

### Task 3 - Activate vEdge on vManage

- Navigate to **Configuration -> Devices**
- Note and use the **Chassis Number** and **Token number** for the 4<sup>th</sup> vEdge from vManage.
- Use the information from the previous step in the following command on the vEdge4 console.

```
request vedge-cloud activate chassis-number XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX token  
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```
- You should see the vEdge in the vManage console with a Certificate issued.

# Lab 11 – Initializing cEdge – CLI



## cEDGE-1

### Task 1 – Configuring the System Component

- Configure the System parameters based on the following:
  - Host-name : cEdge5
  - Organization: KBITS
  - System-IP: 10.2.2.205
  - Site ID: 5
  - vbond Address: 199.1.1.3
  - Timezone: Based on the appropriate Timezone

**Note: Default username:** admin **Default password:** admin

### cEdge1

```
config-transaction
!  
hostname cEdge1  
!  
system
```

```
system-ip 10.2.2.205
site-id 5
organization-name KBITS
vbond 199.1.1.3
exit
!
clock timezone GST 4
commit
```

## **Task 2 – Configure the Interface and Tunnel Parameters**

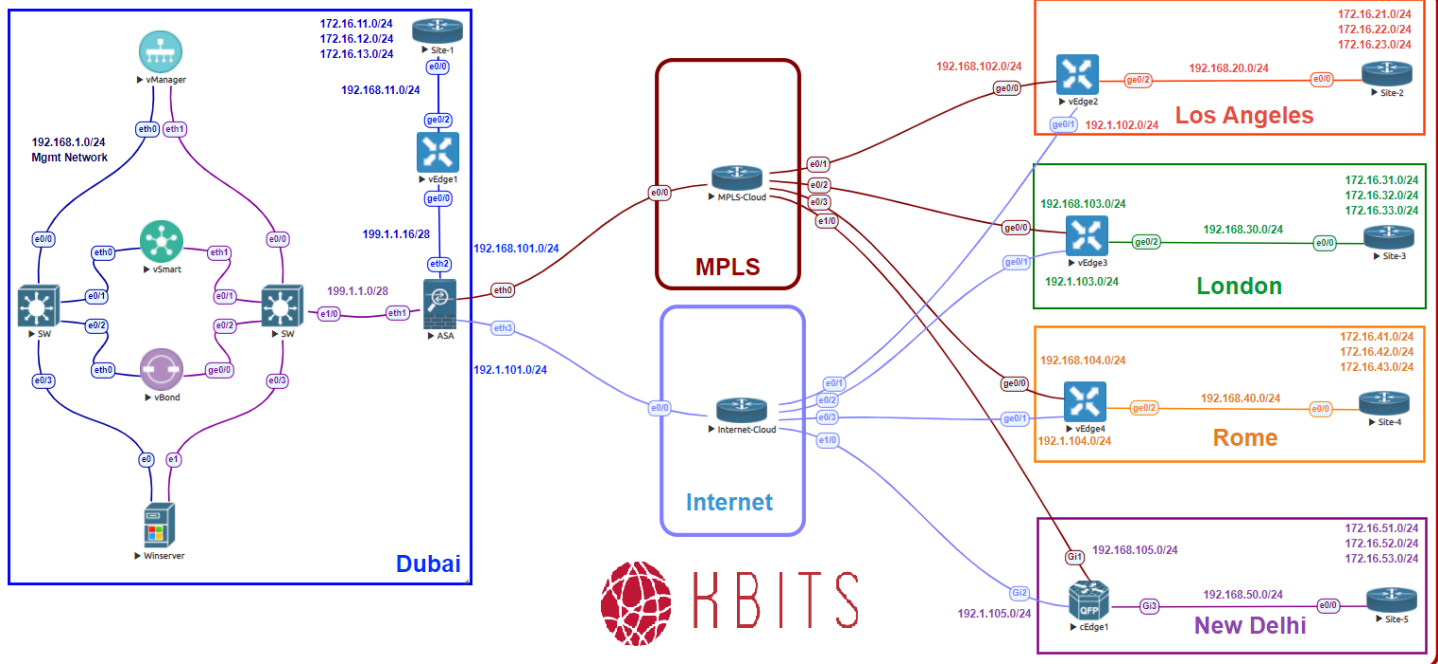
- Configure the Interface parameters based on the following:
  - GigabitEthernet1 Parameters
    - IP Address: 192.168.105.5/24
    - Default Route: 192.168.105.254
  - Tunnel Parameters Parameters
    - Tunnel Interface: Tunnel1
    - Tunnel Source: GigabitEthernet1
    - Tunnel Mode: SDWAN
  - SDWAN Interface Parameters
    - Interface: GigabitEthernet1
    - Encapsulation: IPSec
    - Color: default
    - Tunnel Services (All, NetConf, SSHD)

### **cEdge1**

```
config-transaction
!
interface GigabitEthernet1
no shutdown
ip address 192.168.105.5 255.255.255.0
!
ip route 0.0.0.0 0.0.0.0 192.168.105.254
!
interface Tunnel1
no shutdown
ip unnumbered GigabitEthernet1
tunnel source GigabitEthernet1
tunnel mode sdwan
exit
!
sdwan
interface GigabitEthernet1
```

```
tunnel-interface
encapsulation ipsec
color default
allow-service all
allow-service sshd
allow-service netconf
exit
exit
commit
```

# Lab 12 – Registering cEdges in vManage



## cEDGE-1

### Task 1 – Upload the Root Certificate to the cEdge

- Open the **TFTP Application** on the Windows Server.
- Configure the Default Folder as the **Downloads** Folder and using the 199.1.1.5 as the TFTP Interface.
- Connect to the console of cEdge1 and copy the RootCert.cer file to flash: using the following command:

**copy tftp://199.1.1.5/RootCert.cer flash:**

### Task 2 – Install the Root Certificate on cEdge1

- Connect to the console of cEdge1 and issue the following command:

**request platform software sdwan root-cert-chain install  
bootflash:Root.cer**

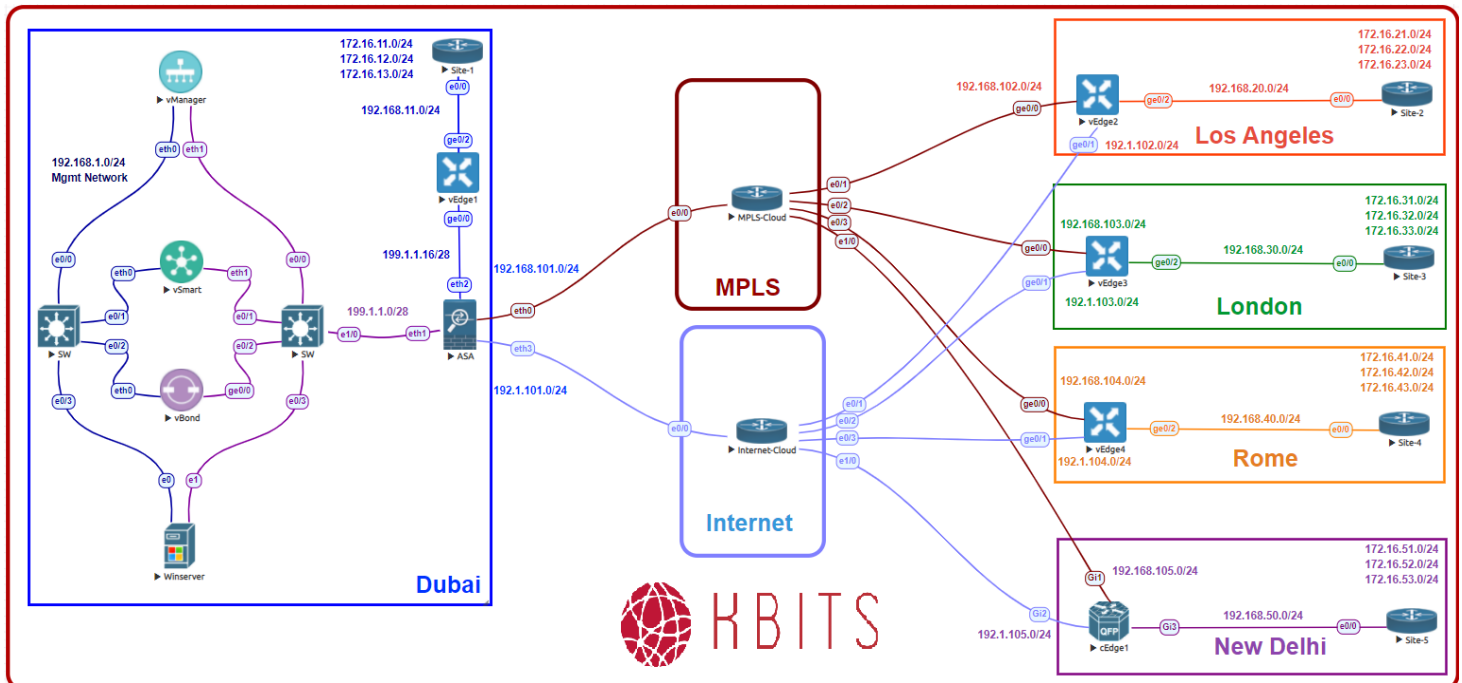
### Task 3 - Activate cEdge on vManage

- Navigate to **Configuration -> Devices**
- Note and use the **Chassis Number** and **Token number** for the 1st CSR Device from vManage.
- Use the information from the previous step in the following command on the cEdge1 console.

```
request platform software sdwan vedge_cloud activate chassis-  
number CSR-XXXXXXXX-XXXX-XXXX-XXXX-  
XXXXXXXXXXXXX token  
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

- You should see the vEdge in the vManage console with a Certificate issued.

# Lab 13 – Configuring Feature Template – System



## Task 1 – Configure the System Template to be used by all vEdge-Cloud Devices

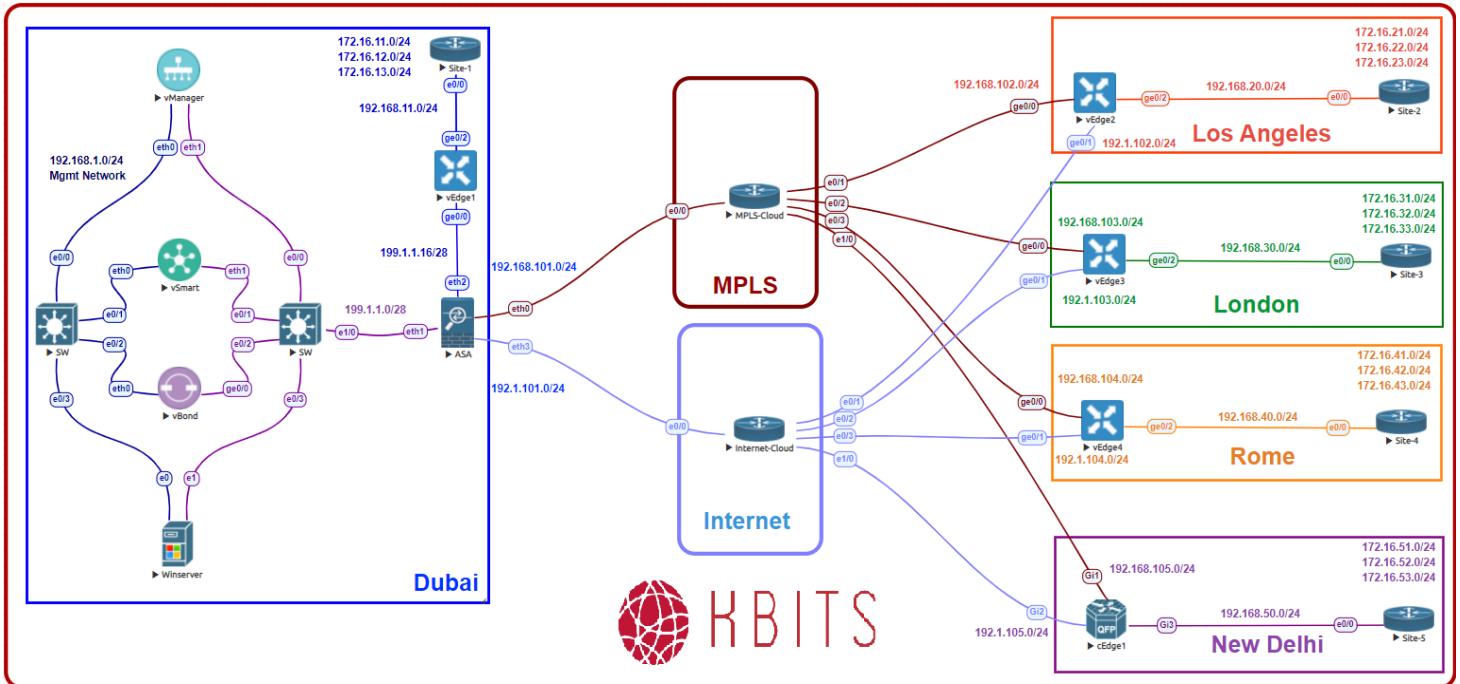
- In vManage, Navigate to Configuration -> **Templates** -> **Feature** -> **vEdge Cloud** -> **Basic Information** -> **System**
- Configure the System parameters based on the following:
  - Template Name : **VE-System**
  - Description : **VE-System**
  - Site ID -> Device Specific
  - System IP -> Device Specific
  - Hostname -> Device Specific
  - Timezone -> Global : **Asia/Muscat**
  - Console Baud Rate -> **Default**
- Click **Save** to save the Template.

## **Task 2 – Configure the System Template to be used by all cEdge-Cloud Devices**

- In vManage, Navigate to Configuration -> **Templates** -> **Feature** -> **CSR Cloud** -> **Basic Information** -> **System**
- Configure the System parameters based on the following:
  - Template Name : **CE-System**
  - Description : **CE-System**
  - Site ID -> Device Specific
  - System IP ->Device Specific
  - Hostname -> Device Specific
  - Timezone -> Global : **Asia/Muscat**
  - Console Baud Rate -> **Default**
- Click **Save** to save the Template.



# Lab 14 – Configuring Feature Template – Banner



## Task 1 – Configure the Banner Template to be used by all vEdge-Cloud Devices

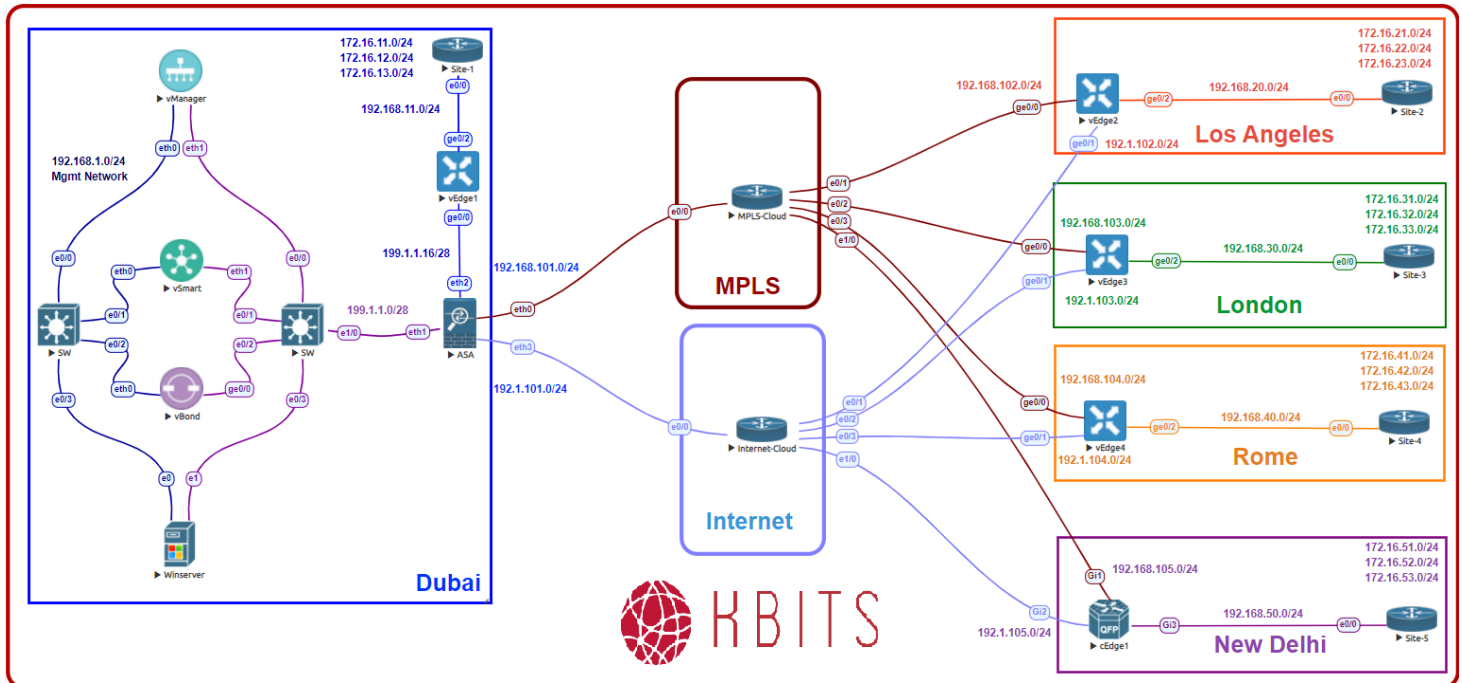
- In vManage, Navigate to Configuration -> **Templates** -> **Feature** -> **vEdge Cloud** -> **Basic Information** -> **Banner**
- Configure the Banner parameters based on the following:
  - Template Name : **VE-Banner**
  - Description : **VE-Banner**
  - Banner: **KBITS Authorized Users Only !!!!!!!!!!!**
  - MOTD: **Welcome of SD-WAB !!!!!!!!!!!!!!!**
- Click **Save** to save the Template.

## Task 2 – Configure the Banner Template to be used by all cEdge-Cloud Devices

- In vManage, Navigate to Configuration -> **Templates** -> **Feature** -> **CSR Cloud** -> **Basic Information** -> **Banner**

- Configure the Banner parameters based on the following:
  - Template Name : **CE-Banner**
  - Description : **CE-Banner**
  - Banner: KBITS Authorized Users Only !!!!!!!!!!!
  - MOTD: Welcome of SD-WAB !!!!!!!!!!!!!!!
  
- Click **Save** to save the Template.

# Lab 15 - Configuring Feature Templates - VPN & VPN Interfaces for VPN 0 & 512 — Branch Site(vEdges)



## Task 1 – Configure a VPN Template to be used by all Branch vEdge-Cloud Devices for VPN 0

- In vManage, Navigate to Configuration -> **Templates** -> **Feature** -> **vEdge Cloud** -> **VPN** -> **VPN**
- Configure the VPN parameters based on the following:

- Template Name : **BR-VE-VPN-VPNO**
- Description : **BR-VE-VPN-VPNO**

### Basic Configuration

- VPN -> Global : **0**
- Name -> Global : **Transport VPN**

### IPv4 Route

- Prefix -> Global : 0.0.0.0/0
- Next Hop -> Device Specific

- Click **Save** to save the Template.

## **Task 2 – Configure a VPN Template to be used by all Branch vEdge-Cloud Devices for VPN 512**

- In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> VPN -> VPN**
- Configure the VPN parameters based on the following:
  - Template Name : **BR-VE-VPN-VPN512**
  - Description : **BR-VE-VPN-VPN512**

### **Basic Configuration**

- VPN -> Global : **512**
  - Name -> Global : **MGMT VPN**
- Click **Save** to save the Template.

## **Task 3 – Configure a VPN Interface Template to be used by all Branch vEdge-Cloud Devices for VPN 0 for Interface G0/0**

- In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> VPN -> VPN Interface Ethernet**
- Configure the VPN parameters based on the following:
  - Template Name : **BR-VE-VPNINT-VPN0-G0**
  - Description : **BR-VE-VPNINT-VPN0-G0**

### **Basic Configuration**

- Shutdown -> Global : **No**
- Interface Name -> Global : **ge0/0**
- IPv4 Address -> Static -> Device Specific

### **Tunnel**

- Tunnel Interface -> Global : **On**
- Color -> Global : **MPLS**

### **Allow Service**

- All -> Global : **On**
- NETCONF -> Global : **On**
- SSH -> Global : **On**

- Click **Save** to save the Template.

#### **Task 4 – Configure a VPN Interface Template to be used by all Branch vEdge-Cloud Devices for VPN 0 for Interface G0/1**

- In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> VPN -> VPN Interface Ethernet**
- Configure the VPN parameters based on the following:
  - Template Name : **BR-VE-VPNINT-VPN0-G1**
  - Description : **BR-VE-VPNINT-VPN0-G1**

##### **Basic Configuration**

- Shutdown -> Global : **No**
- Interface Name -> Global : **ge0/1**
- IPv4 Address -> Static -> Device Specific

##### **Tunnel**

- Tunnel Interface -> Global : **On**
- Color -> Global : **BIZ-Internet**

##### **Allow Service**

- All -> Global : **On**
- NETCONF -> Global : **On**
- SSH -> Global : **On**

Click **Save** to save the Template.

#### **Task 5 – Configure a VPN Interface Template to be used by all Branch vEdge-Cloud Devices for VPN 512 for Interface Eth0**

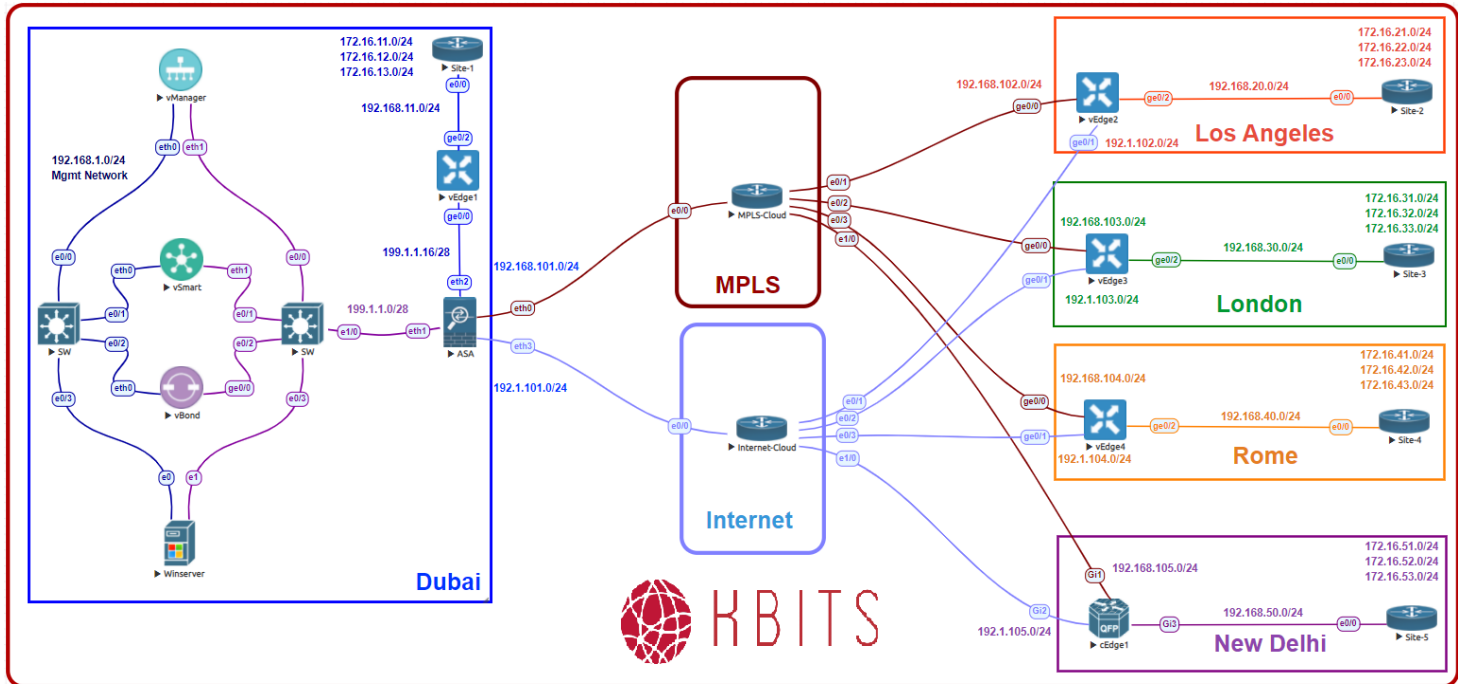
- In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> VPN -> VPN Interface Ethernet**
- Configure the VPN parameters based on the following:
  - Template Name : **BR-VE-VPNINT-VPN512-E0**
  - Description : **BR-VE-VPNINT-VPN512-E0**

##### **Basic Configuration**

- Shutdown -> Global : **No**
- Interface Name -> Global : **eth0**
- IPv4 Address -> Dynamic

- Click **Save** to save the Template

# Lab 16 - Configuring Feature Templates – External Routing - OSPF for VPN 0 – Branch Site(vEdges)



## Task 1 – Configure a OSPF Template to be used by all Branch vEdge-Cloud Devices for VPN 0

➤ In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> Other Templates -> OSPF**

➤ Configure the OSPF parameters based on the following:

- Template Name : **BR-VE-OSPF-VPN0**
- Description : **BR-VE-OSPF-VPN0**

### Area Configuration

- Area Number -> Global : **0**
- Area Type -> Default

### Interface Configuration

- Interface Name: ge0/0

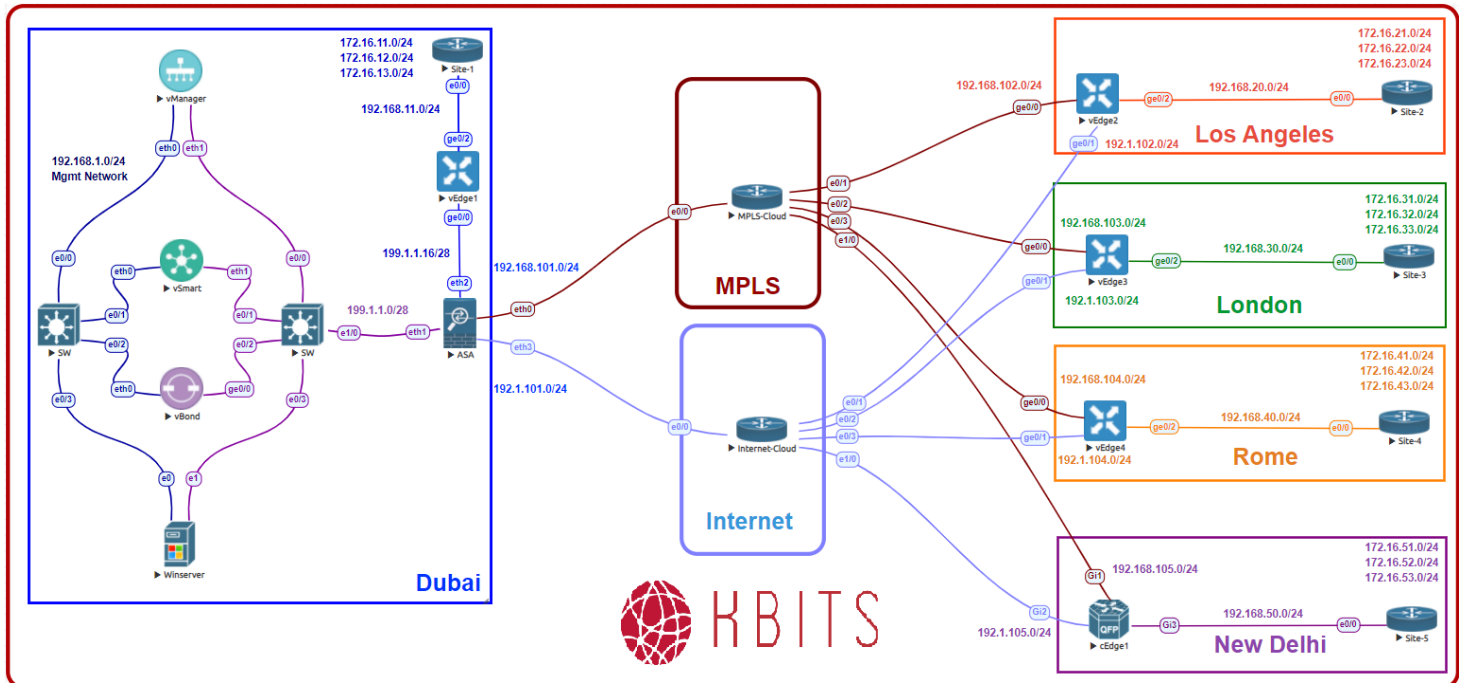
### Advanced

- OSPF Network Type: Point-to-Point

➤ Click **Add** to add the Interface and Click **Add** to add OSPF.

➤ Click **Save** to save the Template.

# Lab 17 - Configuring and Deploying Device Templates for vEdge – Branch Site(vEdge2)



## Task 1 – Configure a Device Template for Branch vEdge Devices.

- In vManage, Navigate to Configuration -> **Templates -> Device -> Create Template -> vEdge Cloud**
- Configure the Device Template based on the following:

- Template Name : **BR-VE-TEMP**
- Description : **BR-VE-TEMP**

### Basic Information

- System -> **VE-System**

### Transport & Management

- VPN 0 : **BR-VE-VPN-VPNO**
- VPN Interface : **BR-VE-VPNINT-VPNO-G0**
- VPN Interface : **BR-VE-VPNINT-VPNO-G1**
- OSPF : **BR-VE-OSPF-VPNO**



- VPN 512 : **BR-VE-VPN-VPN512**
- VPN Interface : **BR-VE-VPNINT-VPN512-E0**

➤ Click **Save** to save the Template.

### **Task 2 – Attach vEdge2 to the Device Template**

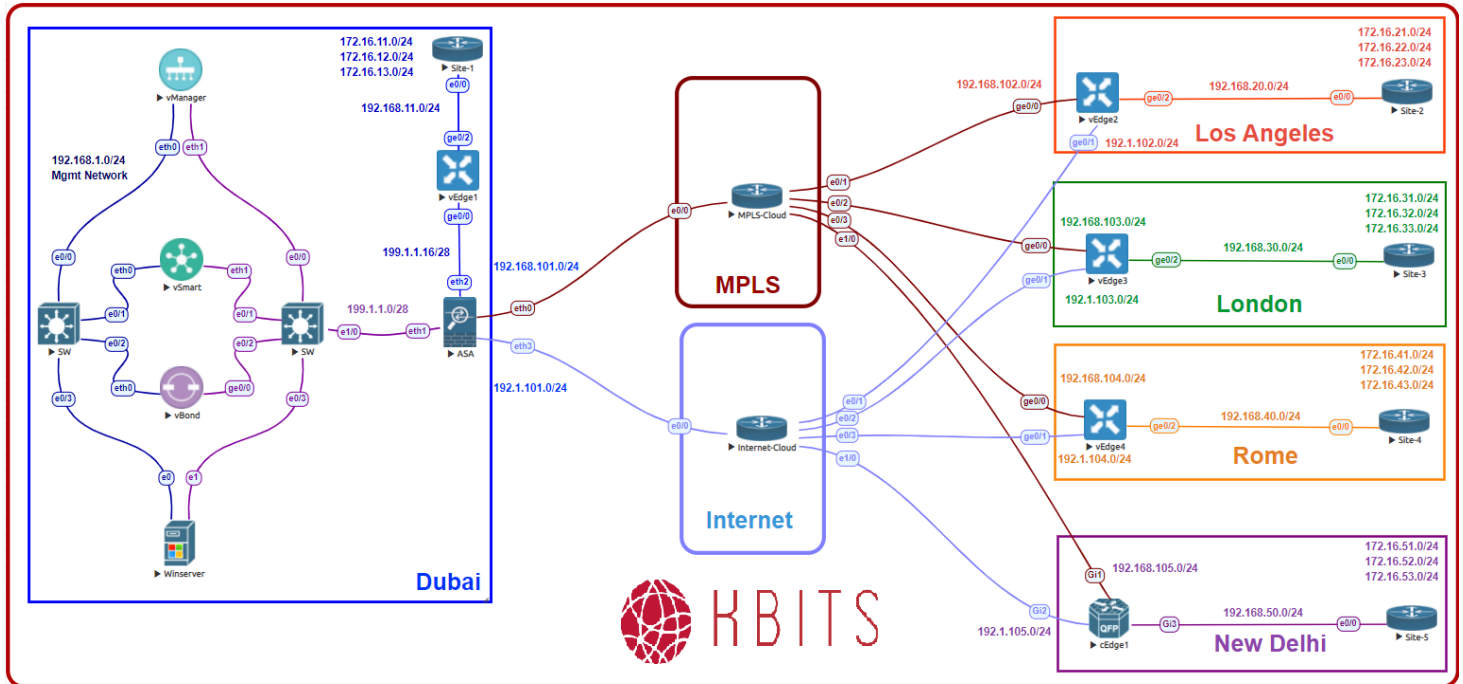
- In vManage, Navigate to Configuration -> **Templates -> Device -> BR-VE-TEMP.**
- Click on “...” towards the right-hand side.
- Click **Attach Devices.**
- Select **vEdge2** and click the “ -> “ button.
- Click **Attach.**

### **Task 3 – Configure the Variable Parameters for the Feature Templates**

- **vEdge2** will appear in the window.
- Click on “...” towards the right-hand side.
- Click **Edit Device Template.**
- Configure the variables based on the following:
  - Default Gateway for VPN0 : **192.1.102.254**
  - Interface IP for ge0/1 : **192.1.102.2/24**
  - Interface IP for ge0/0 : **192.168.102.2/24**
  - Hostname : **vEdge-2**
  - System IP : **10.2.2.202**
  - Site ID : **2**
- Click **Update.**
- Verify the Configuration & Click **Configure Devices.**
- Wait for it to update the device. It should come back with Status of **Success.**

- Verify the configuration on **vEdge2**. You can do that by verify OSPF Neighbor relationship with the MPLS Router by issuing the **Show ospf neighbor** command on **vEdge2**.
- Type **Show Ip route** on **vEdge2** to verify that you are receiving OSPF routes from the MPLS Router.

# Lab 18 - Configuring Internal Routing Protocols on the Internal Routing Devices – HQ & All Branches



## Interface Configuration

### Site-1

Interface	IP Address	Subnet Mask
E 0/0	192.168.11.11	255.255.255.0
Loopback1	172.16.11.1	255.255.255.0
Loopback2	172.16.12.1	255.255.255.0
Loopback3	172.16.13.1	255.255.255.0

### Site-2

Interface	IP Address	Subnet Mask
E 0/0	192.168.20.22	255.255.255.0
Loopback1	172.16.21.1	255.255.255.0
Loopback2	172.16.22.1	255.255.255.0
Loopback3	172.16.23.1	255.255.255.0
Loopback4	172.16.234.2	255.255.255.255

### Site-3

Interface	IP Address	Subnet Mask
E 0/0	192.168.30.33	255.255.255.0
Loopback1	172.16.31.1	255.255.255.0
Loopback2	172.16.32.1	255.255.255.0
Loopback3	172.16.33.1	255.255.255.0
Loopback4	172.16.234.3	255.255.255.255

### Site-4

Interface	IP Address	Subnet Mask
E 0/0	192.168.40.44	255.255.255.0
Loopback1	172.16.41.1	255.255.255.0
Loopback2	172.16.42.1	255.255.255.0
Loopback3	172.16.43.1	255.255.255.0
Loopback4	172.16.234.4	255.255.255.255

### Site-5

Interface	IP Address	Subnet Mask
E 0/0	192.168.50.55	255.255.255.0
Loopback1	172.16.51.1	255.255.255.0
Loopback2	172.16.52.1	255.255.255.0
Loopback3	172.16.53.1	255.255.255.0

### Task 1 – Internal Site Router Configurations

- Configure the Interfaces based on the Logical Diagram
- Configure OSPF as the IGP to communicate with the vEdge/cEdge devices. Enable all the interfaces under OSPF.
- Configure the Loopback Interfaces as OSPF Network Point-to-point Interfaces.

### Site-1

```
no ip domain-lookup
line con 0
  logging sync
  no exec-timeout
!
hostname Site-1
!
interface E 0/0
  ip address 192.168.11.11 255.255.255.0
  no shutdown
!
interface Loopback1
  ip address 172.16.11.1 255.255.255.0
  ip ospf network point-to-point
!
interface Loopback2
  ip address 172.16.12.1 255.255.255.0
  ip ospf network point-to-point
!
interface Loopback3
  ip address 172.16.13.1 255.255.255.0
  ip ospf network point-to-point
!
router ospf 1
  network 192.168.11.0 0.0.0.255 area 0
  network 172.16.0.0 0.0.255.255 area 0
```

## Site-2

```
no ip domain-lookup
line con 0
  logging sync
  no exec-timeout
!
Hostname Site-2
!
Interface E 0/0
  ip address 192.168.20.22 255.255.255.0
  no shutdown
!
Interface Loopback1
  ip address 172.16.21.1 255.255.255.0
  ip ospf network point-to-point
!
Interface Loopback2
  ip address 172.16.22.1 255.255.255.0
  ip ospf network point-to-point
!
Interface Loopback3
  ip address 172.16.23.1 255.255.255.0
  ip ospf network point-to-point
!
Interface Loopback4
  ip address 172.16.234.2 255.255.255.255
  ip ospf network point-to-point
!
router ospf 1
  network 192.168.20.0 0.0.0.255 area 0
  network 172.16.0.0 0.0.255.255 area 0
```

### Site-3

```
no ip domain-lookup
line con 0
  logging sync
  no exec-timeout
!
Hostname Site-3
!
Interface E 0/0
  ip address 192.168.30.33 255.255.255.0
  no shutdown
!
Interface Loopback1
  ip address 172.16.31.1 255.255.255.0
  ip ospf network point-to-point
!
Interface Loopback2
  ip address 172.16.32.1 255.255.255.0
  ip ospf network point-to-point
!
Interface Loopback3
  ip address 172.16.33.1 255.255.255.0
  ip ospf network point-to-point
!
Interface Loopback4
  ip address 172.16.234.3 255.255.255.255
  ip ospf network point-to-point
!
router ospf 1
  network 192.168.30.0 0.0.0.255 area 0
  network 172.16.0.0 0.0.255.255 area 0
```

#### Site-4

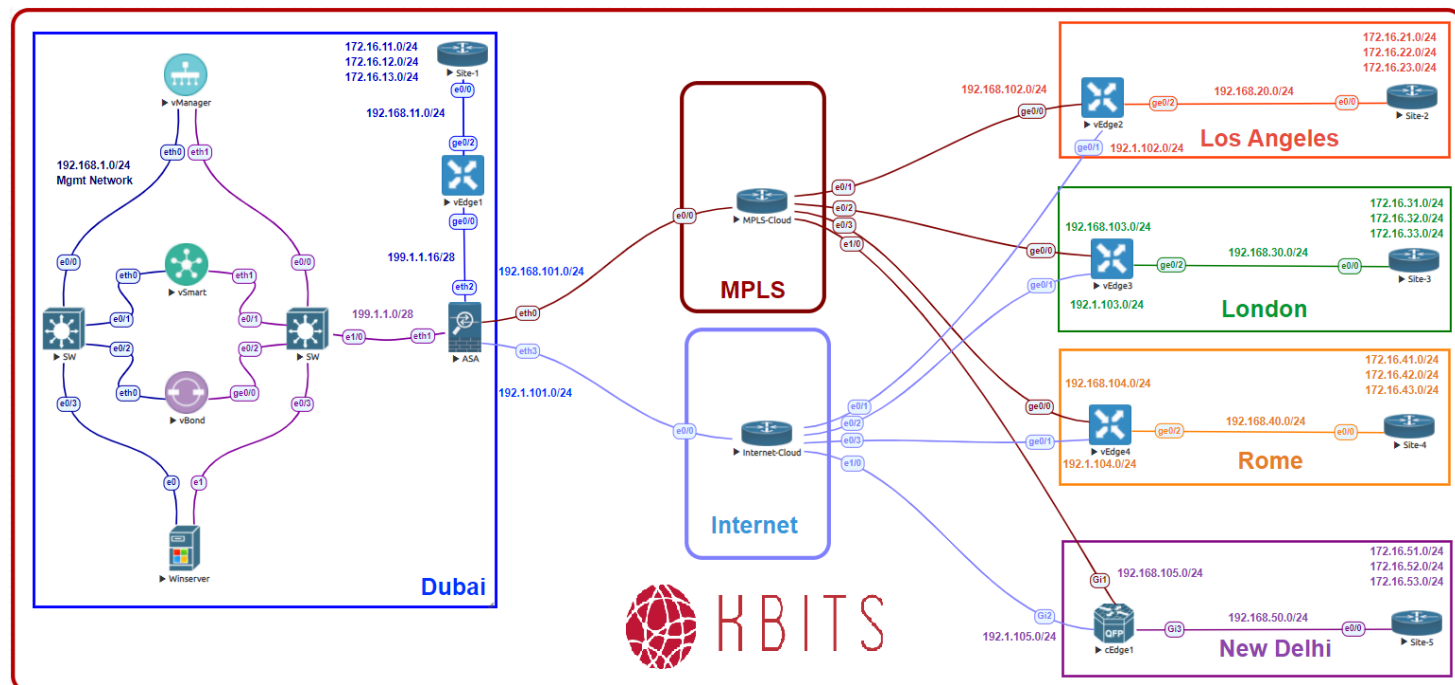
```
no ip domain-lookup
line con 0
  logging sync
  no exec-timeout
!
Hostname Site-4
!
Interface E 0/0
  ip address 192.168.40.44 255.255.255.0
  no shutdown
!
Interface Loopback1
  ip address 172.16.41.1 255.255.255.0
  ip ospf network point-to-point
!
Interface Loopback2
  ip address 172.16.42.1 255.255.255.0
  ip ospf network point-to-point
!
Interface Loopback3
  ip address 172.16.43.1 255.255.255.0
  ip ospf network point-to-point
!
Interface Loopback4
  ip address 172.16.234.4 255.255.255.255
  ip ospf network point-to-point
!
router ospf 1
  network 192.168.40.0 0.0.0.255 area 0
  network 172.16.0.0 0.0.255.255 area 0
```



### Site-5

```
no ip domain-lookup
line con 0
  logging sync
  no exec-timeout
!
hostname Site-5
!
interface E 0/0
  ip address 192.168.50.55 255.255.255.0
  no shutdown
!
interface Loopback1
  ip address 172.16.51.1 255.255.255.0
  ip ospf network point-to-point
!
interface Loopback2
  ip address 172.16.52.1 255.255.255.0
  ip ospf network point-to-point
!
interface Loopback3
  ip address 172.16.53.1 255.255.255.0
  ip ospf network point-to-point
!
router ospf 1
  network 192.168.50.0 0.0.0.255 area 0
  network 172.16.0.0 0.0.255.255 area 0
```

# Lab 19 - Configuring Feature Templates – Service VPN – VPN, VPN Interface and Internal Routing – Branch Site(vEdges)



## Task 1 – Configure a VPN Template to be used by all Branch vEdge-Cloud Devices for VPN 1

- In vManage, Navigate to Configuration -> **Templates** -> **Feature** -> **vEdge Cloud** -> **VPN** -> **VPN**
- Configure the VPN parameters based on the following:

- Template Name : **BR-VE-VPN-VPN1**
- Description : **BR-VE-VPN-VPN1**

### Basic Configuration

- VPN -> Global : **1**
- Name -> Global : **Data VPN**
- Click **Save** to save the Template.

## Task 2 – Configure a VPN Interface Template to be used by all Branch vEdge-Cloud Devices for VPN 1 for Interface G0/2

➤ In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> VPN -> VPN Interface Ethernet**

➤ Configure the VPN parameters based on the following:

- Template Name : **BR-VE-VPNINT-VPN1-G2**
- Description : **BR-VE-VPNINT-VPN1-G2**

#### **Basic Configuration**

- Shutdown -> Global : **No**
- Interface Name -> Global : **ge0/2**
- IPv4 Address -> Static -> Device Specific

➤ Click **Save** to save the Template.

### **Task 3 – Configure a OSPF Template to be used by all Branch vEdge-Cloud Devices for VPN 1**

➤ In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> Other Templates -> OSPF**

➤ Configure the OSPF parameters based on the following:

- Template Name : **BR-VE-OSPF-VPN1**
- Description : **BR-VE-OSPF-VPN1**

#### **Redistribution**

- Protocol : **OMP**

#### **Area Configuration**

- Area Number -> Global : **0**
- Area Type -> Default

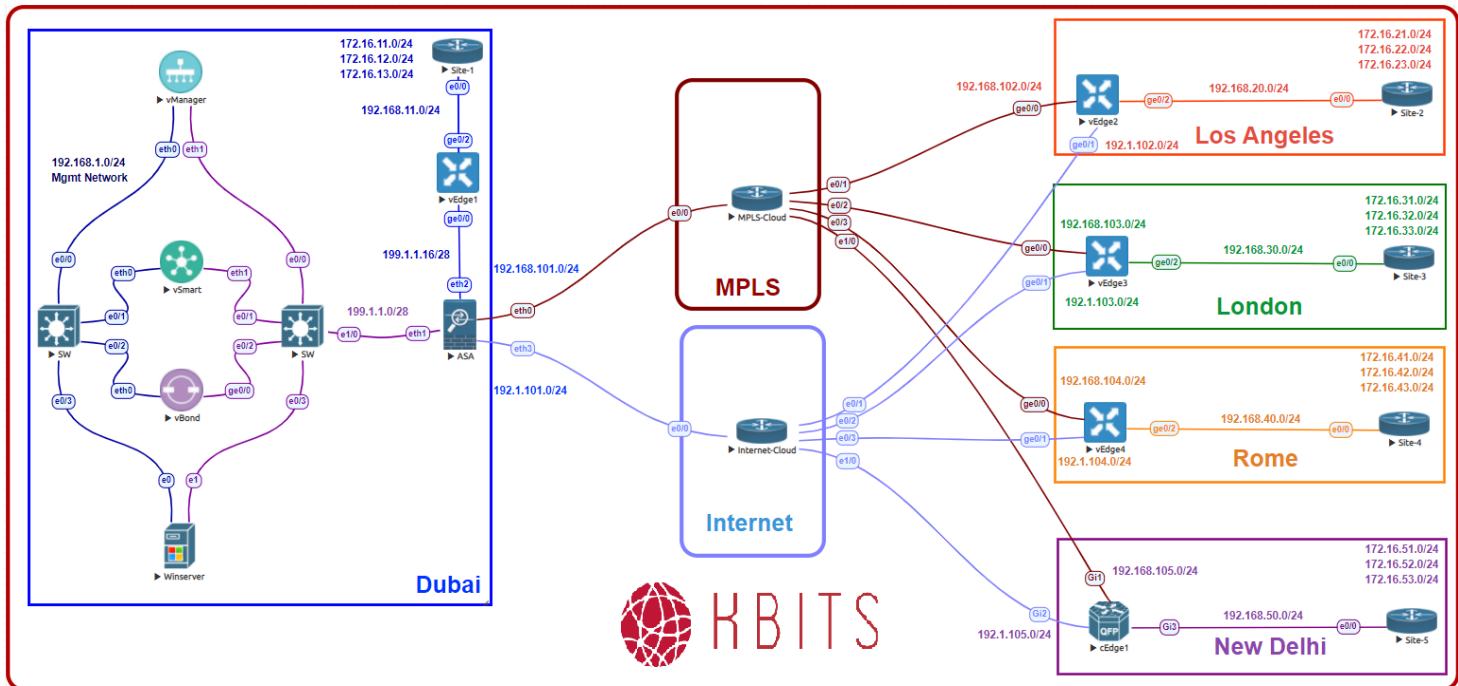
#### **Interface Configuration**

- Interface Name: **ge0/2**

➤ Click **Add** to add the Interface and Click **Add** to add OSPF.

➤ Click **Save** to save the Template.

# Lab 20 - Implementing a Service VPN using Templates – Branch Site(vEdge2)



## Task 1 – Edit the BR-VE-TEMP Device Template for Branch vEdge Devices.

- In vManage, Navigate to Configuration -> **Templates** -> **Device** -> **BR-VE-TEMP** -> “...” -> **Edit**
- Edit the BR-VE-TEMP Device Template based on the following:

### Service VPN

- VPN 1 : **BR-VE-VPN-VPN1**
- VPN Interface : **BR-VE-VPNINT-VPN1-G2**
- OSPF : **BR-VE-OSPF-VPN1**

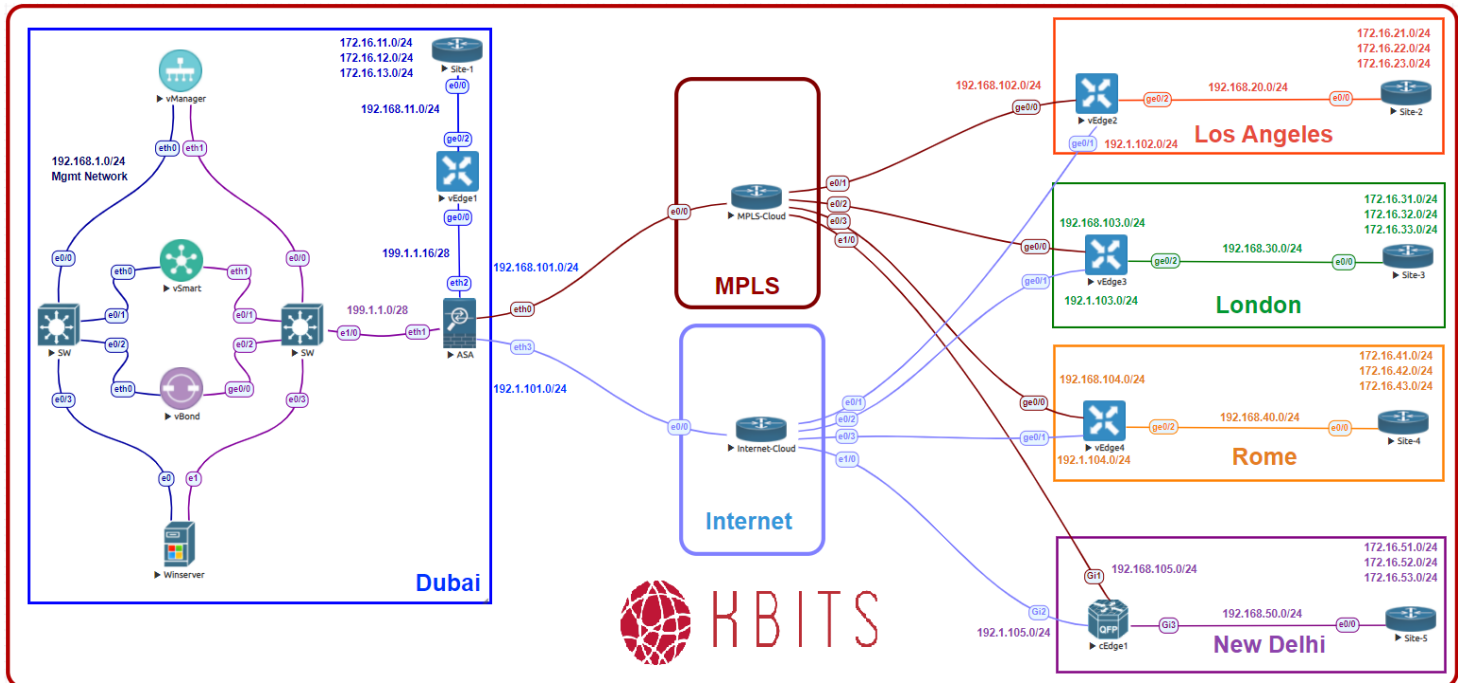
- Click **Save** to save the Template.

## Task 2 – Configure the Variable Parameters for the Feature Templates

- **vEdge2** will appear in the window.

- Click on “...” towards the right-hand side & click **Edit Device Template.**
- Configure the variables based on the following:
  - Interface IP for ge0/2 :**192.168.20.2/24**
- Click **Update.**
- Verify the Configuration & Click **Configure Devices.**
- Wait for it to update the device. It should come back with Status of **Success.**
- Verify the configuration on **vEdge2**. You can do that by verify OSPF Neighbor relationship with the Site-2 Router by issuing the **Show ospf neighbor** command on **vEdge2**.
- Type **Show Ip route** on **vEdge2** to verify that you are receiving OSPF routes from the Internal Site Router.

# Lab 21 - Pushing Template to configure other Branch Sites - – Branch Site(vEdge3 & vEdge4)



## Task 1 – Attach the BR-VE-TEMP Device Template for Branch vEdge Devices.

- In vManage, Navigate to Configuration -> **Templates** -> **Device** -> **BR-VE-TEMP** -> “...” -> **Attach Devices**.
- Click **Attach Devices**.
- Select **vEdge3** & **vEdge4** and click the “ -> ” button.
- Click **Attach**.
- **vEdge3** & **vEdge4** will appear in the window.
- Click on “...” towards the right-hand side for both devices, one at a time click **Edit Device Template**.
- Configure the variables based on the following:

### vEdge-3

- Interface IP for ge0/2 :**192.168.30.3/24**
- Default Gateway for VPN0 : **192.1.103.254**
- Interface IP for ge0/1 :**192.1.103.3/24**
- Interface IP for ge0/0 :**192.168.103.3/24**
- Hostname : **vEdge-3**
- System IP : **10.2.2.203**
- Site ID : **3**

- Click **Update**.

### vEdge-4

- Interface IP for ge0/2 :**192.168.40.4/24**
- Default Gateway for VPN0 : **192.1.104.254**
- Interface IP for ge0/1 :**192.1.104.4/24**
- Interface IP for ge0/0 :**192.168.104.4/24**
- Hostname : **vEdge-4**
- System IP : **10.2.2.204**
- Site ID : **4**

- Click **Update**.

- Verify the Configuration & Click **Configure Devices**.

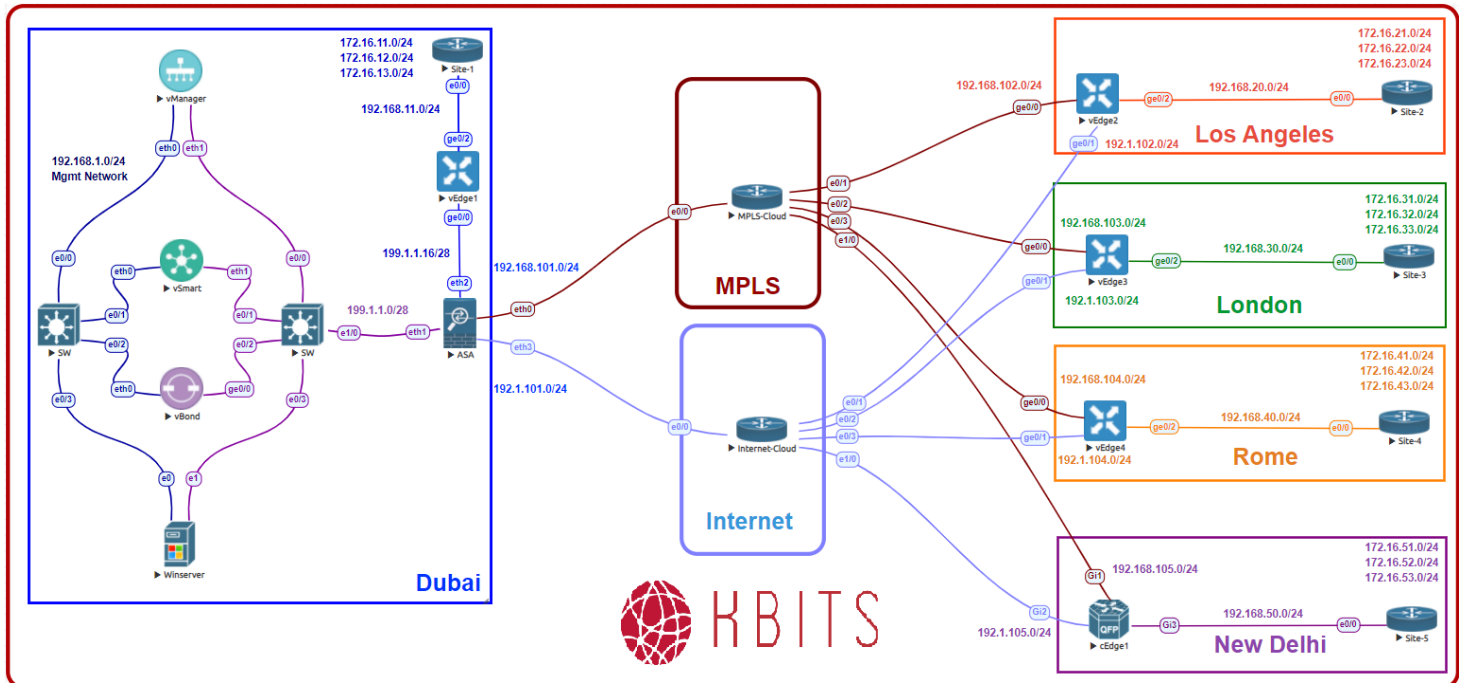
- Wait for it to update the device. It should come back with Status of **Success**.

- Verify the configuration on **vEdge3** & **vEdge4**. You can do that by verify OSPF Neighbor relationship with the Internal Site Router by issuing the **Show ospf neighbor** command on the **vEdges**.

- Type **Show Ip route** on **Internal Site Routers** to verify that you are receiving OSPF routes from the other Sites.

- Verify reachability between the sites by Pinging the Internal Loopback to Loopback networks.

# Lab 22 – Configuring Feature Templates for HQ-Site(vEdge1) – VPNs, VPN Interfaces, External & Internal Routing



## VPN 0

### Task 1 – Configure a VPN Template for HQ vEdge-Cloud Devices for VPN 0

- In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> VPN -> VPN**
- Configure the VPN parameters based on the following:

- Template Name : **HQ-VE-VPN-VPN0**
- Description : **HQ-VE-VPN-VPN0**

#### Basic Configuration

- VPN -> Global : **0**
- Name -> Global : **Transport VPN**

#### IPv4 Route

- Prefix -> Global : **0.0.0.0/0**



- Next Hop -> Device Specific
- Click **Save** to save the Template.

### **Task 2 – Configure a VPN Interface Template to be used by HQ vEdge-Cloud Devices for VPN 0 for Interface G0/0**

- In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> VPN -> VPN Interface Ethernet**

- Configure the VPN parameters based on the following:

- Template Name : **HQ-VE-VPNINT-VPNO-GO**
- Description : **HQ-VE-VPNINT-VPNO-GO**

#### **Basic Configuration**

- Shutdown -> Global : **No**
- Interface Name -> Global : **ge0/0**
- IPv4 Address -> Static -> Device Specific

#### **Tunnel**

- Tunnel Interface -> Global : **On**
- Color -> Default

#### **Allow Service**

- All -> Global : **On**
- NETCONF -> Global : **On**
- SSH -> Global : **On**

- Click **Save** to save the Template.

### **Task 3 – Configure a BGP Template to be used by HQ vEdge-Cloud Devices for VPN 0**

- In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> Other Templates -> BGP**

- Configure the BGP parameters based on the following:

- Template Name : **HQ-VE-BGP-VPNO**
- Description : **HQ-VE-BGP-VPNO**

#### **Basic Configuration**

- Shutdown -> Global : **No**
- AS Number -> Global : **65001**

#### **Neighbor**

- Address -> Global : **199.1.1.30**
  - Remote AS -> Global : **65001**
  - Address Family -> Global : **On**
  - Address Family -> Global : **IPv4-Unicast**
- Click **Add** to add the Interface and Click **Add** to add BGP Neighbor.
  - Click **Save** to save the Template.

## VPN 512

### Task 1 – Configure a VPN Template to be used by HQ vEdge-Cloud Devices for VPN 512

- In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> VPN -> VPN**
- Configure the VPN parameters based on the following:
  - Template Name : **HQ-VE-VPN-VPN512**
  - Description : **HQ-VE-VPN-VPN512**

**Basic Configuration**

  - VPN -> Global : **512**
  - Name -> Global : **MGMT VPN**
- Click **Save** to save the Template.

### Task 2 – Configure a VPN Interface Template to be used by HQ vEdge-Cloud Devices for VPN 512 for Interface Eth0

- In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> VPN -> VPN Interface Ethernet**
- Configure the VPN parameters based on the following:
  - Template Name : **HQ-VE-VPNINT-VPN512-E0**
  - Description : **HQ-VE-VPNINT-VPN512-E0**

**Basic Configuration**

  - Shutdown -> Global : **No**
  - Interface Name -> Global : **eth0**
  - IPv4 Address -> Dynamic

- Click **Save** to save the Template

## VPN 1

### Task 1 – Configure a VPN Template for HQ vEdge-Cloud Devices for VPN 1

- In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> VPN -> VPN**
- Configure the VPN parameters based on the following:
  - Template Name : **HQ-VE-VPN-VPN1**
  - Description : **HQ-VE-VPN-VPN1**
- Basic Configuration**
  - VPN -> Global : **1**
  - Name -> Global : **Data VPN**
- Click **Save** to save the Template.

### Task 2 – Configure a VPN Interface Template to be used by HQ vEdge-Cloud Devices for VPN 1 for Interface G0/2

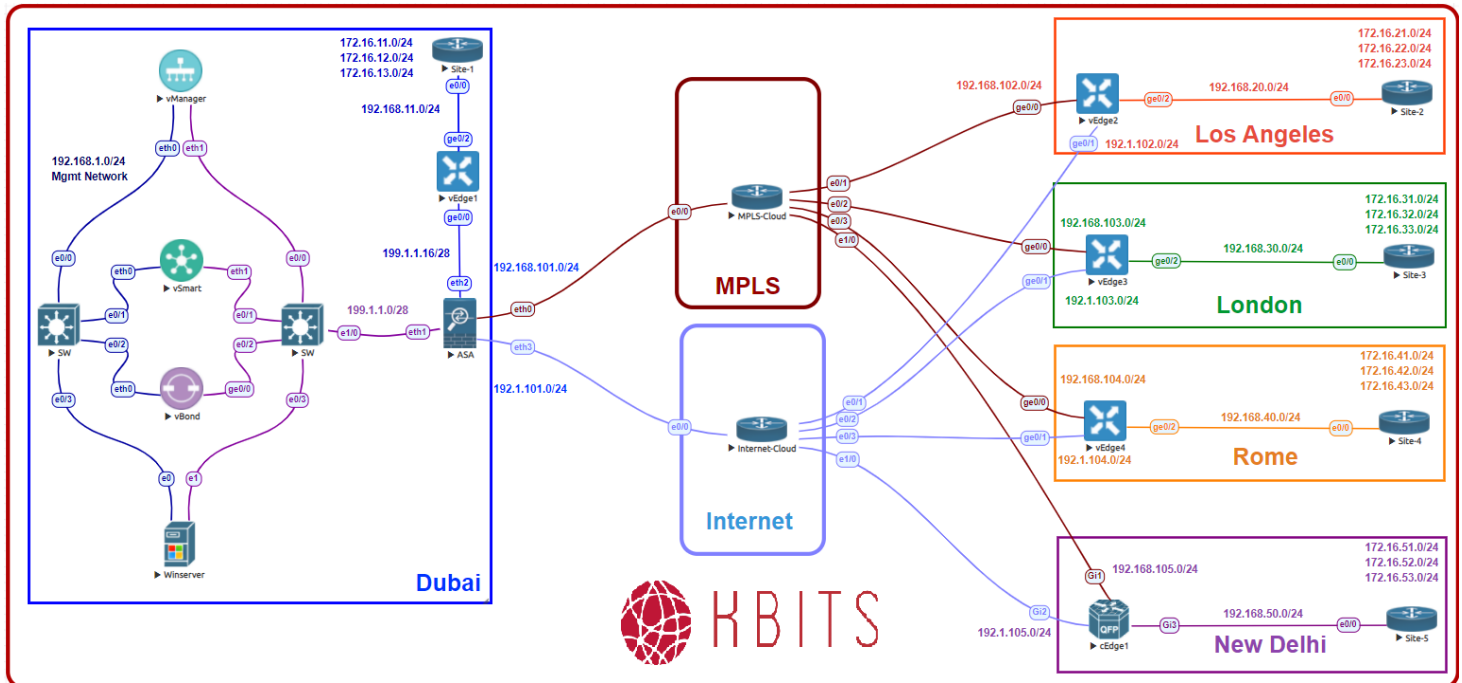
- In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> VPN -> VPN Interface Ethernet**
- Configure the VPN parameters based on the following:
  - Template Name : **HQ-VE-VPNINT-VPN1-G2**
  - Description : **HQ-VE-VPNINT-VPN1-G2**
- Basic Configuration**
  - Shutdown -> Global : **No**
  - Interface Name -> Global : **ge0/2**
  - IPv4 Address -> Static -> Device Specific
- Click **Save** to save the Template.

### Task 3 – Configure a OSPF Template to be used by HQ vEdge-Cloud Devices for VPN 1

- In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> Other Templates -> OSPF**

- Configure the OSPF parameters based on the following:
  - Template Name : **HQ-VE-OSPF-VPN1**
  - Description : **HQ-VE-OSPF-VPN1**
  
  - Redistribution**
  - Protocol : **OMP**
  
  - Area Configuration**
  - Area Number -> Global : **0**
  - Area Type -> Default
  - Interface Configuration**
  - Interface Name: ge0/2
  
- Click **Add** to add the Interface and Click **Add** to add OSPF.
  
- Click **Save** to save the Template.

# Lab 23 - Configuring Device Templates for HQ-Site(vEdge1) to deploy VPN 0, 1 and 512.



## Task 1 – Configure a Device Template for HQ vEdge Devices.

➤ In vManage, Navigate to Configuration -> **Templates -> Device -> Create Template -> vEdge Cloud**

➤ Configure the Device Template based on the following:

- Template Name : **HQ-VE-TEMP**
- Description : **HQ-VE-TEMP**

### Basic Information

- System -> **VE-System**

### Transport & Management

- VPN 0 : **HQ-VE-VPN-VPN0**
- VPN Interface : **HQ-VE-VPNINT-VPN0-GO**
- BGP : **HQ-VE-BGP-VPN0**
- VPN 512 : **HQ-VE-VPN-VPN512**

- VPN Interface : **HQ-VE-VPNINT-VPN512-E0**

#### **Service VPN**

- VPN 1 : **HQ-VE-VPN-VPN1**
- VPN Interface : **HQ-VE-VPNINT-VPN1-G2**
- OSPF : **HQ-VE-OSPF-VPN1**

- Click **Save** to save the Template.

### **Task 2 – Attach vEdge1 to the Device Template**

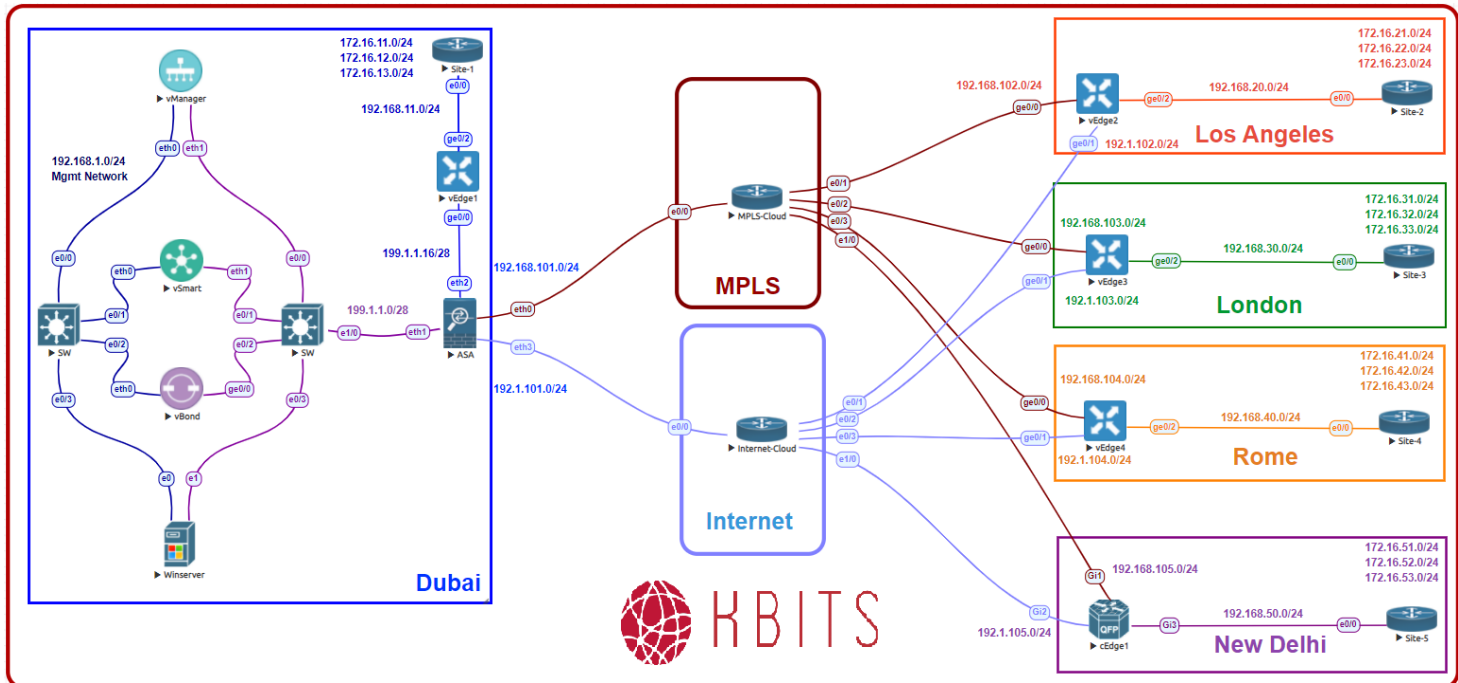
- In vManage, Navigate to Configuration -> **Templates -> Device -> HQ-VE-TEMP.**
- Click on “...” towards the right-hand side.
- Click **Attach Devices.**
- Select **vEdge1** and click the “->” button.
- Click **Attach.**

### **Task 3 – Configure the Variable Parameters for the Feature Templates**

- **vEdge1** will appear in the window.
- Click on “...” towards the right-hand side.
- Click **Edit Device Template.**
- Configure the variables based on the following:
  - Interface IP for ge0/2 : **192.168.11.1/24**
  - Default Gateway for VPN0 : **199.1.1.30**
  - Interface IP for ge0/0 : **199.1.1.17/28**
  - Hostname : **vEdge-1**
  - System IP : **10.2.2.201**
  - Site ID : **1**
- Click **Update.**
- Verify the Configuration & Click **Configure Devices.**

- Wait for it to update the device. It should come back with Status of **Success**.
- Verify the configuration on **vEdge1**. You can do that by verify OSPF Neighbor relationship with the Internal Router by issuing the **Show ospf neighbor** command on **vEdge1**.
- Type **Show Ip route** on **vEdge2** to verify that you are receiving OSPF routes from the MPLS Router.
- Type **Show Ip route** on **Internal Site Routers** to verify that you are receiving OSPF routes from the other Sites.
- Verify reachability between the sites by Pinging the Internal Loopback to Loopback networks.

# Lab 24 – Configuring Feature Templates for CSR – VPNs, VPN Interfaces, External & Internal Routing



## VPN 0

### Task 1 – Configure a VPN Template by CSR for VPN 0

- In vManage, Navigate to Configuration -> **Templates** -> **Feature** -> **CSR1000v** -> **VPN** -> **VPN**
- Configure the VPN parameters based on the following:

- Template Name : **BR-CSR-VPN-VPN0**
- Description : **BR-CSR -VPN-VPN0**

#### Basic Configuration

- VPN -> Global : **0**
- Name -> Global : **Transport VPN**

#### IPv4 Route

- Prefix -> Global : **0.0.0.0/0**
- Next Hop -> Device Specific



- Click **Save** to save the Template.

### **Task 2 – Configure a VPN Interface Template to be used by CSR for VPN 0 for Interface GigabitEthernet1**

- In vManage, Navigate to Configuration -> **Templates -> Feature -> CSR1000v -> VPN -> VPN Interface Ethernet**

- Configure the VPN parameters based on the following:

- Template Name : **BR-CSR-VPNINT-VPN0-G1**
- Description : **BR-CSR-VPNINT-VPN0-G1**

#### **Basic Configuration**

- Shutdown -> Global : **No**
- Interface Name -> Global : **GigabitEthernet1**
- IPv4 Address -> Static -> Device Specific

#### **Tunnel**

- Tunnel Interface -> Global : **On**
- Color -> Default

#### **Allow Service**

- All -> Global : **On**
- NETCONF -> Global : **On**
- SSH -> Global : **On**

- Click **Save** to save the Template.

### **Task 3 – Configure a VPN Interface Template to be used by CSR for VPN 0 for Interface GigabitEthernet2**

- In vManage, Navigate to Configuration -> **Templates -> Feature -> CSR1000v -> VPN -> VPN Interface Ethernet**

- Configure the VPN parameters based on the following:

- Template Name : **BR-CSR-VPNINT-VPN0-G2**
- Description : **BR-CSR-VPNINT-VPN0-G2**

#### **Basic Configuration**

- Shutdown -> Global : **No**
- Interface Name -> Global : **GigabitEthernet2**
- IPv4 Address -> Static -> Device Specific

### Tunnel

- Tunnel Interface -> Global : **On**
- Color -> Default
- Allow Service**
- All -> Global : **On**
- NETCONF -> Global : **On**
- SSH -> Global : **On**

➤ Click **Save** to save the Template.

### Task 4 – Configure a OSPF Template to be used by CSR for VPN 0

- In vManage, Navigate to Configuration -> **Templates -> Feature -> CSR1000v -> Other Templates -> OSPF**
- Configure the OSPF parameters based on the following:
  - Template Name : **BR-CSR-OSPF-VPN0**
  - Description : **BR-CSR-OSPF-VPN0**

### Area Configuration

- Area Number -> Global : **0**
- Area Type -> Default

### Interface Configuration

- Interface Name: **GigabitEthernet1**
- OSPF Network Type: **Point-to-Point**

- Click **Add** to add the Interface and Click **Add** to add OSPF.
- Click **Save** to save the Template.

## VPN 512

### Task 1 – Configure a VPN Template to be used by CSR for VPN 512

- In vManage, Navigate to Configuration -> **Templates -> Feature -> CSR1000v -> VPN -> VPN**
- Configure the VPN parameters based on the following:
  - Template Name : **BR-CSR-VPN-VPN512**
  - Description : **BR-CSR-VPN-VPN512**

### Basic Configuration

- VPN -> Global : **512**
- Name -> Global : **MGMT VPN**

➤ Click **Save** to save the Template.

### Task 2 – Configure a VPN Interface Template to be used by CSR for VPN 512 for Interface GigabitEthernet4

➤ In vManage, Navigate to Configuration -> **Templates -> Feature -> CSR1000v -> VPN -> VPN Interface Ethernet**

➤ Configure the VPN parameters based on the following:

- Template Name : **BR-CSR-VPNINT-VPN512-G4**
- Description : **BR-CSR-VPNINT-VPN512-G4**

### Basic Configuration

- Shutdown -> Global : **No**
- Interface Name -> Global : **GigabitEthernet4**
- IPv4 Address -> Dynamic

➤ Click **Save** to save the Template

## VPN 1

### Task 1 – Configure a VPN Template for CSR for VPN 1

➤ In vManage, Navigate to Configuration -> **Templates -> Feature -> CSR1000v -> VPN -> VPN**

➤ Configure the VPN parameters based on the following:

- Template Name : **BR-CSR-VPN-VPN1**
- Description : **BR-CSR-VPN-VPN1**

### Basic Configuration

- VPN -> Global : **1**
- Name -> Global : **Data VPN**

➤ Click **Save** to save the Template.

### Task 2 – Configure a VPN Interface Template to be used by CSR for VPN 1 for Interface G3

➤ In vManage, Navigate to Configuration -> **Templates -> Feature -> CSR -> VPN -> VPN Interface Ethernet**

➤ Configure the VPN parameters based on the following:

- Template Name : **BR-CSR-VPNINT-VPN1-G3**
- Description : **BR-CSR-VPNINT-VPN1-G3**

#### **Basic Configuration**

- Shutdown -> Global : **No**
- Interface Name -> Global : **GigabitEthernet3**
- IPv4 Address -> Static -> Device Specific

➤ Click **Save** to save the Template.

### **Task 3 – Configure a OSPF Template to be used by CSR for VPN 1**

➤ In vManage, Navigate to Configuration -> **Templates -> Feature -> CSR -> Other Templates -> OSPF**

➤ Configure the OSPF parameters based on the following:

- Template Name : **BR-CSR-OSPF-VPN1**
- Description : **BR-CSR-OSPF-VPN1**

#### **Redistribution**

- Protocol : **OMP**

#### **Area Configuration**

- Area Number -> Global : **0**
- Area Type -> Default

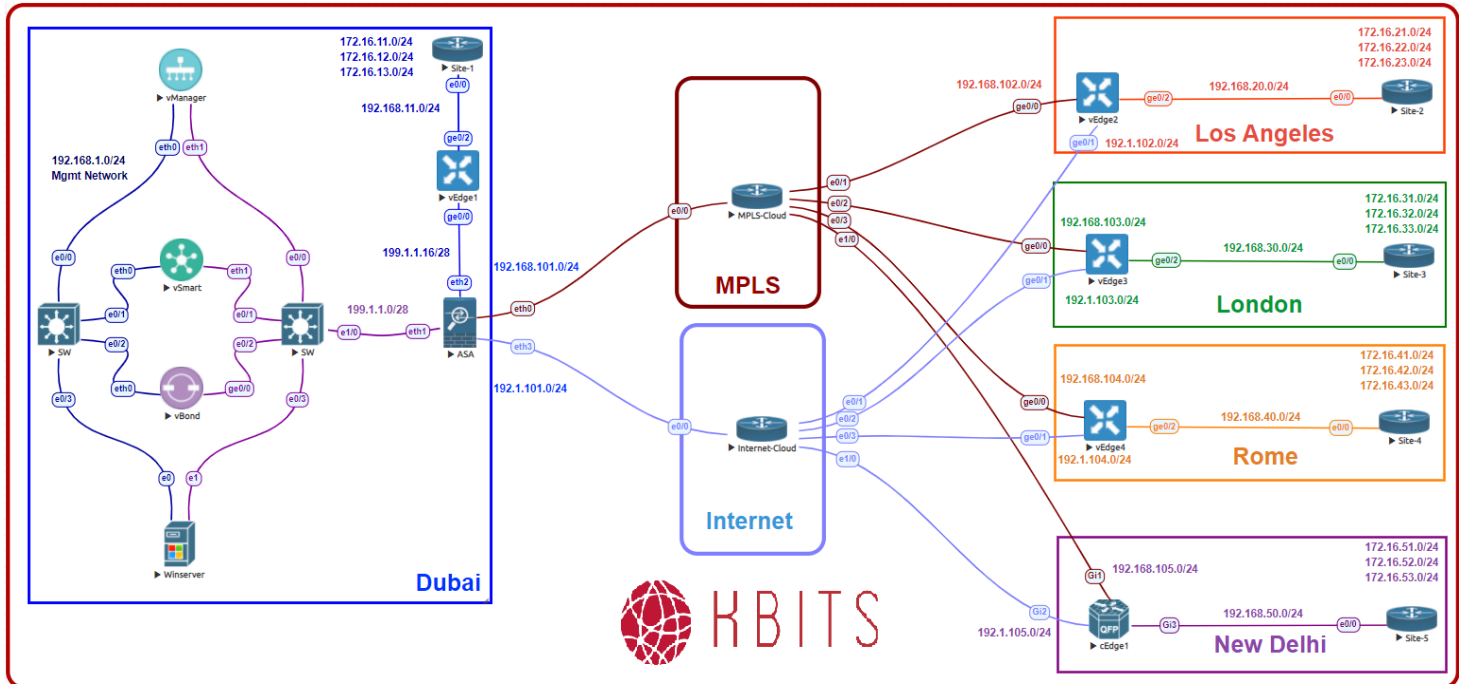
#### **Interface Configuration**

- Interface Name: **GigabitEthernet3**

➤ Click **Add** to add the Interface and Click **Add** to add OSPF.

➤ Click **Save** to save the Template.

# Lab 25 - Configuring Device Templates for CSR to deploy VPN 0, 1 and 512



## Task 1 – Configure a Device Template for CSR Branch Devices.

- In vManage, Navigate to Configuration -> **Templates** -> **Device** -> **Create Template** -> **CSR1000v**
- Configure the Device Template based on the following:

- Template Name : **BR-CSR-TEMP**
- Description : **BR-CSR-TEMP**

### Basic Information

- System -> **CE-System**

### Transport & Management

- VPN 0 : **BR-CSR-VPN-VPN0**
- VPN Interface : **BR-CSR-VPNINT-VPN0-G1**
- VPN Interface : **BR-CSR-VPNINT-VPN0-G2**
- OSPF : **BR-CSR-OSPF-VPN0**
- VPN 512 : **BR-CSR-VPN-VPN512**
- VPN Interface : **BR-CSR-VPNINT-VPN512-E0**

### Service VPN

- VPN 1 : **BR-CSR-VPN-VPN1**
- VPN Interface : **BR-CSR-VPNINT-VPN1-G3**
- OSPF : **BR-CSR-OSPF-VPN1**

➤ Click **Save** to save the Template.

### Task 2 – Attach cEdge1 to the Device Template

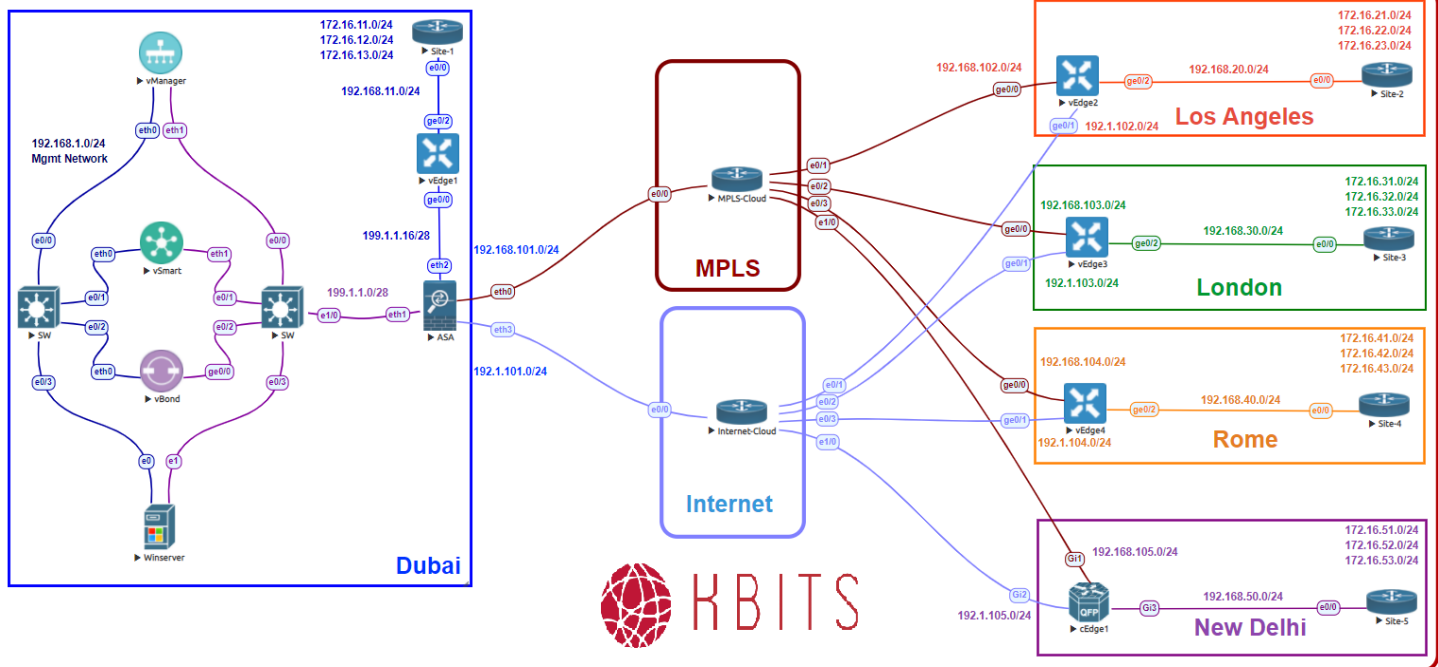
- In vManage, Navigate to Configuration -> **Templates -> Device -> BR-CSR-TEMP.**
- Click on “...” towards the right-hand side.
- Click **Attach Devices.**
- Select **cEdge1** and click the “ -> “ button.
- Click **Attach.**

### Task 3 – Configure the Variable Parameters for the Feature Templates

- **cEdge1** will appear in the window.
- Click on “...” towards the right-hand side.
- Click **Edit Device Template.**
- Configure the variables based on the following:
  - Interface IP for GigabitEthernet3 : **192.168.50.5/24**
  - Default Gateway for VPN0 : **192.1.105.254**
  - Interface IP for GigabitEthernet2 : **192.1.105.5/24**
  - Interface IP for GigabitEthernet1 : **192.168.105.5/24**
  - Hostname : **cEdge-1**
  - System IP : **10.2.2.205**
  - Site ID : **5**
- Click **Update.**
- Verify the Configuration & Click **Configure Devices.**

- Wait for it to update the device. It should come back with Status of **Success**.
- Verify the configuration on **cEdge1**. You can do that by verify OSPF Neighbor relationship with the Internal Router by issuing the **Show ip ospf neighbor** command on **cEdge1**.
- Type **Show Ip route** on **cEdge1** to verify that you are receiving OSPF routes from the MPLS Router.
- Type **Show Ip route** on **Internal Site Routers** to verify that you are receiving OSPF routes from the other Sites.
- Verify reachability between the sites by Pinging the Internal Loopback to Loopback networks.

# Lab 26 - Configuring and Deploying Feature and Device Templates for vSmart Controllers



## Task 1 – Configure a VPN Template to be used by vSmart Controllers for VPN 0

- In vManage, Navigate to Configuration -> **Templates** -> **Feature** -> **vSmart** -> **VPN** -> **VPN**
- Configure the VPN parameters based on the following:

- Template Name : **vSmart-VPN-VPNO**
- Description : **vSmart-VPN-VPNO**

### Basic Configuration

- VPN -> Global : **0**
- Name -> Global : **Transport VPN**

### IPv4 Route

- Prefix -> Global : **0.0.0.0/0**
- Next Hop -> Global : **199.1.1.14**

- Click **Save** to save the Template.



## Task 2 – Configure a VPN Template to be used by vSmart Controllers for VPN 512

- In vManage, Navigate to Configuration -> **Templates -> Feature -> vSmart -> VPN -> VPN**
- Configure the VPN parameters based on the following:
  - Template Name : **vSmart -VPN-VPN512**
  - Description : **vSmart -VPN-VPN512**
- Basic Configuration**
  - VPN -> Global : **512**
  - Name -> Global : **MGMT VPN**
- Click **Save** to save the Template.

## Task 3 – Configure a VPN Interface Template to be used by vSmart Controllers for VPN 0 for Interface Eth1

- In vManage, Navigate to Configuration -> **Templates -> Feature -> vSmart -> VPN -> VPN Interface Ethernet**
- Configure the VPN parameters based on the following:
  - Template Name : **vSmart-VPNINT-VPN0-E1**
  - Description : **vSmart-VPNINT-VPN0-E1**
- Basic Configuration**
  - Shutdown -> Global : **No**
  - Interface Name -> Global : **eth1**
  - IPv4 Address -> Static -> Device Specific
- Tunnel**
  - Tunnel Interface -> Global : **On**
  - Color -> default
- Allow Service**
  - All -> Global : **On**
  - NETCONF -> Global : **On**
  - SSH -> Global : **On**
- Click **Save** to save the Template.

#### **Task 4 – Configure a VPN Interface Template to be used vSmart Controllers for VPN 512 for Interface Eth0**

- In vManage, Navigate to Configuration -> **Templates -> Feature -> vSmart -> VPN -> VPN Interface Ethernet**
- Configure the VPN parameters based on the following:
  - Template Name : **vSmart-VPNINT-VPN512-E0**
  - Description : **vSmart-VPNINT-VPN512-E0**
  - Basic Configuration**
    - Shutdown -> Global : **No**
    - Interface Name -> Global : **eth0**
    - IPv4 Address -> Static -> Device-Specific
- Click **Save** to save the Template

#### **Task 5 – Configure a Device Template for vSmart Controllers.**

- In vManage, Navigate to Configuration -> **Templates -> Device -> Create Template -> vSmart**
- Configure the Device Template based on the following:
  - Template Name : **vSmart-TEMP**
  - Description : **vSmart-TEMP**
  - Basic Information**
    - System -> **VE-System**
  - Transport & Management**
    - VPN 0 : **vSmart-VPN-VPNO**
    - VPN Interface : **vSmart-VPNINT-VPNO-E1**
  
    - VPN 512 : **vSmart-VPN-VPN512**
    - VPN Interface : **vSmart-VPNINT-VPN512-E0**
- Click **Save** to save the Template.

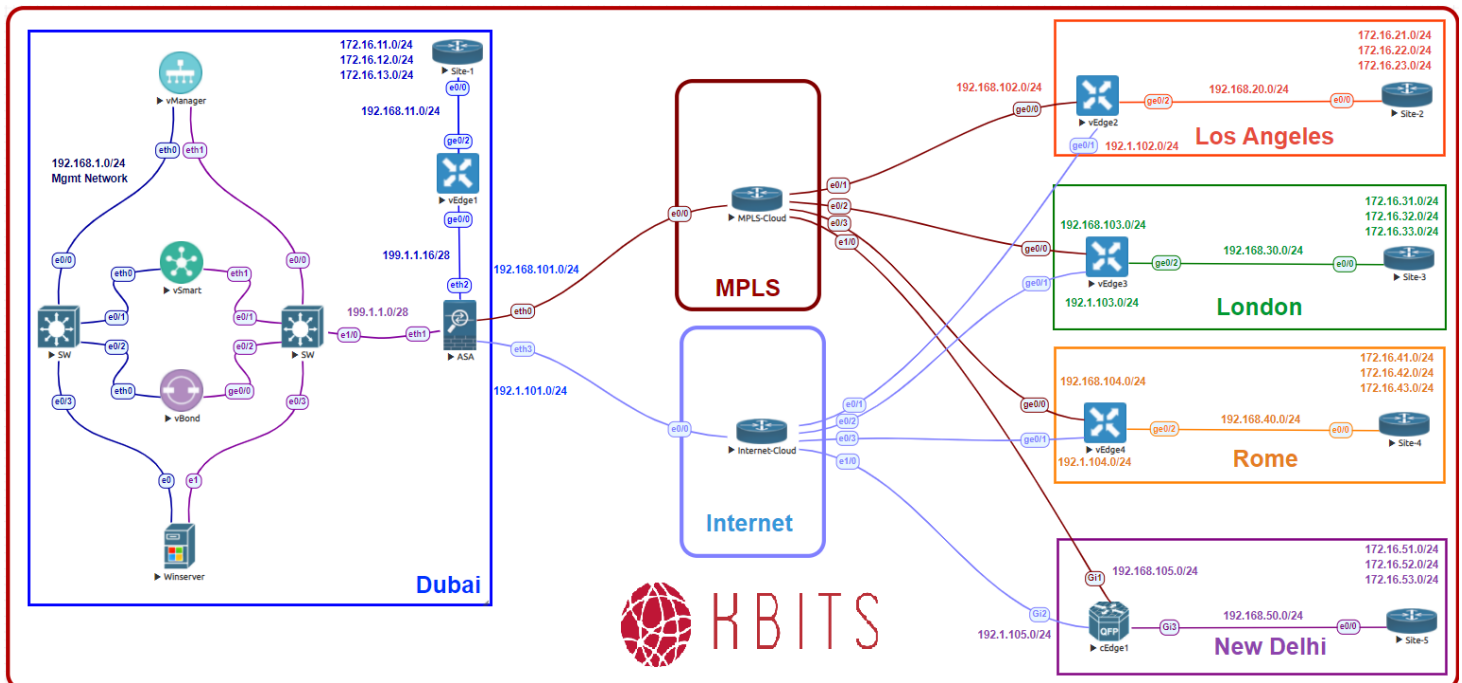
## Task 6 – Attach vSmart to the Device Template

- In vManage, Navigate to Configuration -> **Templates** -> **Device** -> **vSmart-TEMP**
- Click on “...” towards the right-hand side.
- Click **Attach Devices**.
- Select **vSmart** and click the “ -> “ button.
- Click **Attach**.

## Task 7 – Configure the Variable Parameters for the Feature Templates

- **vSmart** will appear in the window.
- Click on “...” towards the right-hand side.
- Click **Edit Device Template**.
- Configure the variables based on the following:
  - Interface IP for Eth1 : **199.1.1.2/28**
  - Interface IP for Eth0 : **192.168.1.2/24**
  - Hostname : **vSmart-1**
  - System IP : **10.1.1.102**
  - Site ID : **1**
- Click **Update**.
- Verify the Configuration & Click **Configure Devices**.
- Wait for it to update the device. It should come back with Status of **Success**.

# Lab 27 - Configuring Application Aware Policies using Telnet and Web



## Requirements:

- Los Angeles & London Sites should use the MPLS Transport for Telnet Traffic and the Biz-Internet Transport for Web Traffic.
- Telnet Should have a SLA based on the following:
  - Loss – 5%
  - Latency – 200
  - Jitter – 100ms
- Web Should have a SLA based on the following:
  - Loss – 10%
  - Latency – 500
  - Jitter – 100ms
- Create the Sites for Los Angeles and London.
- Create the VPN for VPN ID 1.

## **Task 1 – Configure Groups of Interests/List that will be used for Telnet & Web Application Aware Routing (AAR) Policy**

- In vManage, Navigate to **Configuration -> Policies -> Custom Options -> Centralized Policy -> Lists.**
- Click **SLA Class** and select **New SLA Class list.** Create 2 policies based on the following:
  - Name : **SLA-Telnet**
  - Loss : **5%**
  - Latency : **200**
  - Jitter : **100ms**
  
  - Name : **SLA-Web**
  - Loss : **10%**
  - Latency : **500**
  - Jitter : **100ms**
- Click **VPN** and select **New VPN list.** Create 1 policy based on the following:
  - Name : **VPN1**
  - ID : **1**
- Click **Site** and select **New Site list.** Create 2 policies based on the following:
  - Name : **Los Angeles**
  - Site ID : **2**
  
  - Name : **London**
  - Site ID : **3**

## Task 2 – Configure an AAR policy based on the Requirements

- In vManage, Navigate to **Configuration -> Policies -> Custom Options -> Centralized Policy -> Traffic Policy**.
- Configure 2 App Routes based on the following:
  - Policy Name : **TELNET-WEB-Policy**
  - Description : **TELNET-WEB-Policy**

### Telnet Sequence

#### Match Conditions:

- Protocol : **6**
- Port : **23**

#### Action

- SLA Class List: **SLA-Telnet**
- Color : **mpls**
- Backup Preferred Color: **biz-internet**
  
- Click **Save Match and Actions** to save the Sequence.

### Web Sequence

#### Match Conditions:

- Protocol : **6**
- Port : **80**

#### Action

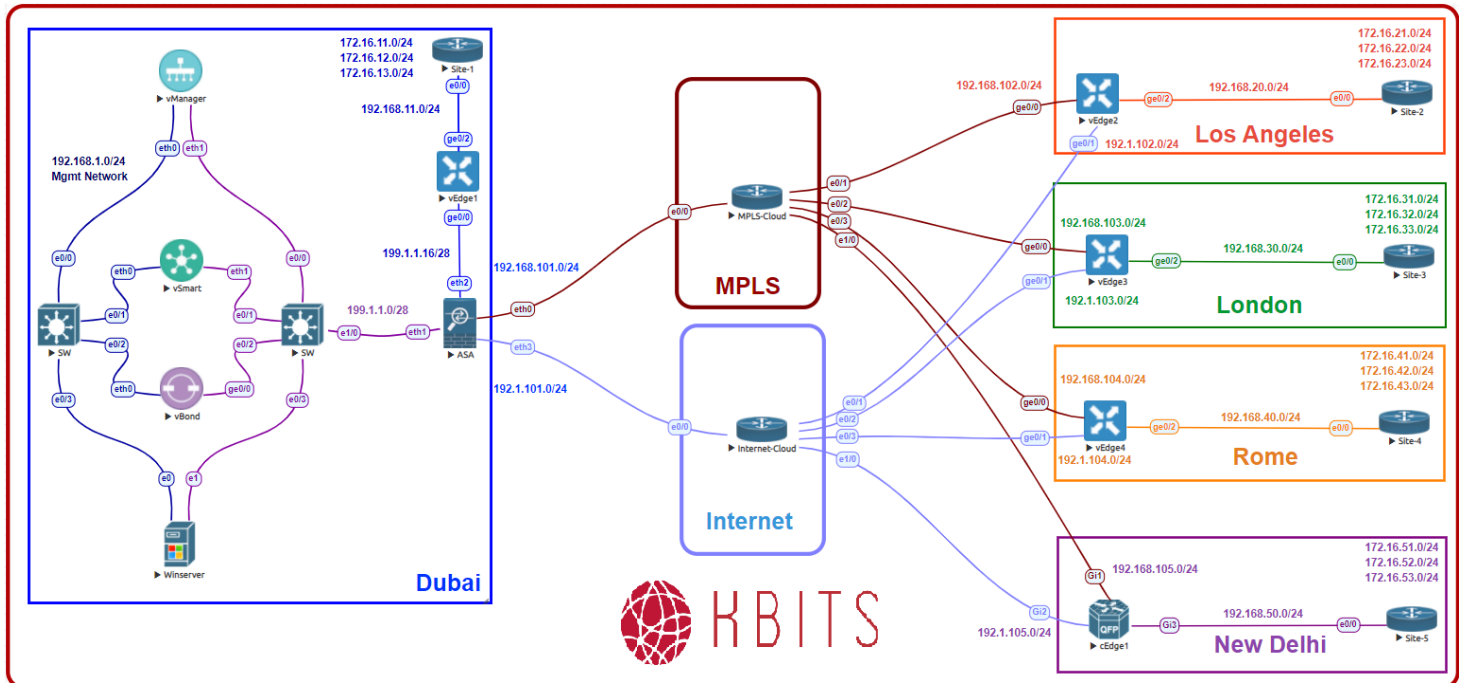
- SLA Class List: **SLA-Web**
- Color : **biz-internet**
- Backup Preferred Color: **mpls**
  
- Click **Save Match and Actions** to save the Sequence.
  
- **Save the Policy.**

## Task 3 – Create a Centralized Policy and call the Traffic Policy

- In vManage, Navigate to **Configuration -> Policies -> Custom Options -> Centralized Policy -> Add Centralized Policy**
- Click **Next** on the **“Group of Interests”** page as we have already created the required lists.
- Click **Next** on the **“Topology and VPN Membership”** page as we are not using any Control Policies.

- Click **Add Policy** on the “**Configure Traffic Rules**” page.
- Click “**Import Existing**” and select the **TELNET-WEB-POLICY** from the drop-down list and click **Import**.
- Click **Next** to move to the “**Apply Policy to Sites and VPNs**” Page.
- Click the “**Application-Aware Policy**” tab.
- The **TELNET-WEB-Policy** will be there. Click “**New Site List and VPN List**” button.
- Select **Los Angeles** and **London** in the Site List.
- Select **VPN1** in the Site List.
- Click **Add**.
- Assign the Policy a name and Description based on the following:
  - Policy Name : **Main-Central-Policy**
  - Description : **Main-Central-Policy**
- Click the **Save Policy** button towards the button.
- Activate the policy.
- Wait for it to push the policy to the reachable vSmart Controller(s).
- Verify the policy by using the **Monitor -> Network -> vEdge2 -> Troubleshooting -> Simulate Flows** Tool.
- Telnet from Los Angeles or London should only use the **mpls** transport.
- Web from Los Angeles or London should only use the **biz-internet** transport.
- Normal Ping from Los Angeles or London should use both the Transports.

# Lab 28 - Configuring Application Aware Policies using Chat Applications



## Requirements:

- Rome should use the Internet Transport for AOL-Messenger, MSN Messenger & Whatsapp Messenger application. It should not use the MPLS Transport at all.
- The Chat applications should have a SLA based on the following:
  - Loss – 10%
  - Latency – 600
  - Jitter – 100ms
- Create a Site for Rome

## Task 1 – Configure Groups of Interests/List that will be used for Chat-based Application Aware Routing (AAR) Policy

- In vManage, Navigate to **Configuration -> Policies -> Custom Options -> Centralized Policy -> Lists**.
- Click **Applications** and select **New Application list**. Create a policy based on the following:
  - Name : **Chat-Apps**



- Appls: Aol-Messenger, MSN-Messenger & WhatsApp Messenger
- Click **SLA Class** and select **New SLA Class list**. Create a policy based on the following:
  - Name : **SLA-CHATS**
  - Loss : **25%**
  - Latency : **600**
  - Jitter : **100ms**
- Click **Site** and select **New Site list**. Create a policy based on the following:
  - Name : **Rome**
  - Site ID : **4**

## Task 2 – Configure an AAR policy based on the Requirements

➤ In vManage, Navigate to **Configuration -> Policies -> Custom Options -> Centralized Policy -> Traffic Policy**.

➤ Configure 1 App Routes based on the following:

- Policy Name : **CHAT-Policy**
- Description : **CHAT-Policy**

### Telnet Sequence

#### Match Conditions:

- Application List: **Chat-Apps**
- **Action**
- SLA Class List: **SLA-CHATS**
- Color : **mpls**
- Backup Preferred Color: **biz-internet**

- Click **Save Match and Actions** to save the Sequence.

### Web Sequence

#### Match Conditions:

- Protocol : **6**
- Port : **80**
- **Action**
- SLA Class List: **SLA-Web**
- Color : **biz-internet**
- Strict: **Checked**

- Click **Save Match and Actions** to save the Sequence.

- **Save the Policy.**

## Task 3 – Modify the existing Centralized Policy “Main-Central-Policy” and call the Traffic Policy

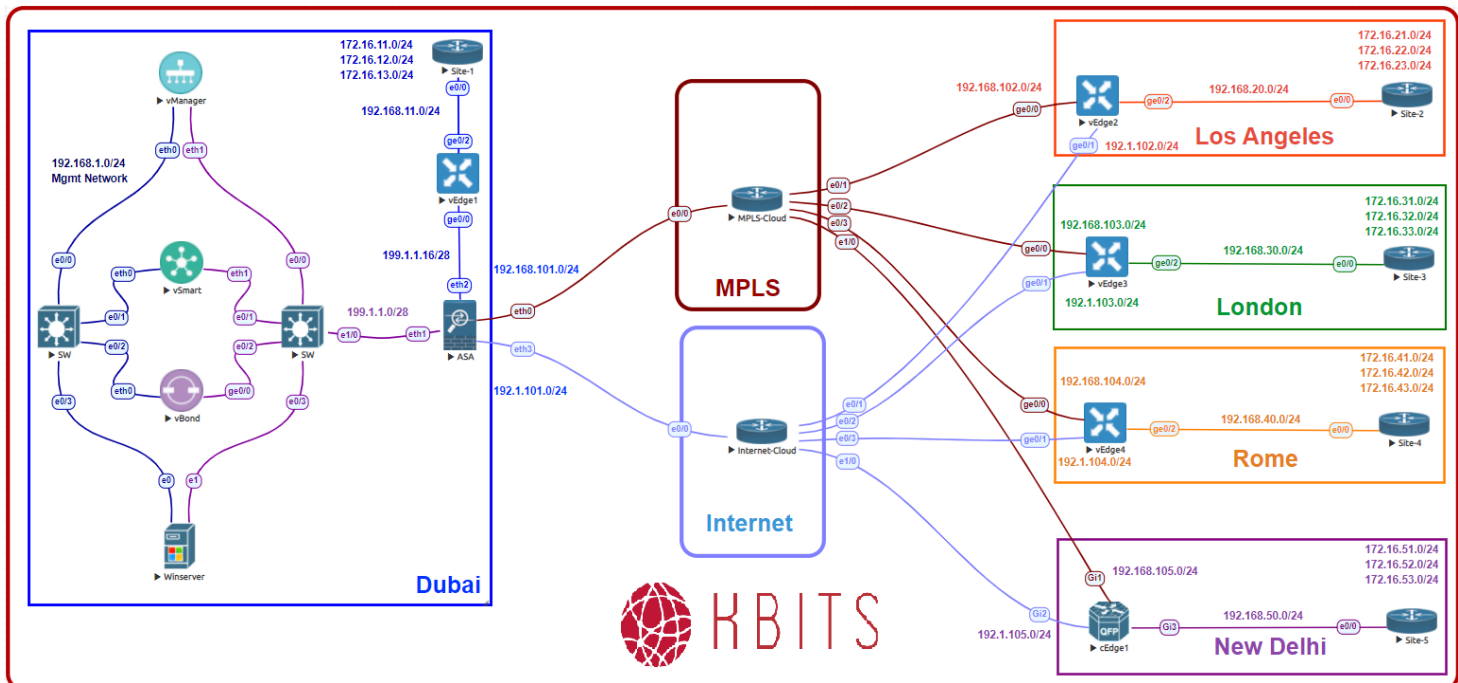
➤ In vManage, Navigate to **Configuration -> Policies -> Custom Options -> Centralized Policy -> Main-Central-Policy -> Click “...” -> Edit**.

➤ Click **Traffic Rules** on the **Top** of the page.

➤ Click **Add Policy**.

- Click “**Import Existing**” and select the **CHAT-POLICY** from the drop-down list and click **Import**.
- Click **Policy Application** on the **Top** of the page.
- Click the “**Application-Aware Policy**” tab.
- The **CHAT-Policy** will be there. Click “**New Site List and VPN List**” button.
- Select **Rome** in the Site List.
- Select **VPN1** in the Site List.
- Click **Add**.
- Click the **Save Policy** button towards the button.
- **Activate** the policy.
- Wait for it to push the policy to the reachable vSmart Controller(s).
- Verify the policy by using the **Monitor -> Network -> vEdge3 -> Troubleshooting -> Simulate Flows** Tool.
- Normal Ping from Rome should use both the Transports.
- Use **Aol-messenger** as the application and simulate from Rome. It should only use the **biz-internet** transport.
- Use **Aol-messenger** as the application and simulate from Los Angeles or London. It should use both the Transports.

# Lab 29 - Manipulating Traffic flow using TLOCs



## Requirements:

- Rome should only use the MPLS TLOC as the preferred color while communicating to Los Angeles. The Internet TLOC should be backup TLOC.

## Task 1 - Configure Groups of Interests/List that will be used for Traffic Engineering Policy for Rome

- In vManage, Navigate to **Configuration -> Policies -> Custom Options -> Centralized Policy -> Lists**.
- Click **TLOCs** and select **New TLOC list**. Create a policy based on the following:
  - Name : **LA-TLOC-MPLS-INT**
  - TLOC#1:
    - IP Address: 10.2.2.202
    - Color: MPLS
    - Encapsulation: IPSec
    - Preference: 300
  - TLOC#2:

- IP Address: 10.2.2.202
- Color: Biz-internet
- Encapsulation: IPSec
- Preference: 200

## **Task 2 – Configure Control/Topology policy based on the Requirements**

- In vManage, Navigate to **Configuration -> Policies -> Custom Options -> Centralized Policy -> Topology**.
- Configure 1 Route Policy based on the following:

- Policy Name : **LA-MPLS-INT**
- Description : **LA-MPLS-INT**

### **Route Sequence**

#### **Match Conditions:**

- Site List: **LosAngeles**
- VPN List: **VPN1**

#### **Action**

- TLOC/TLOC List: **LA-MPLS-INT**
- Click **Save Match and Actions** to save the Sequence.

### **Default Sequence**

#### **Action**

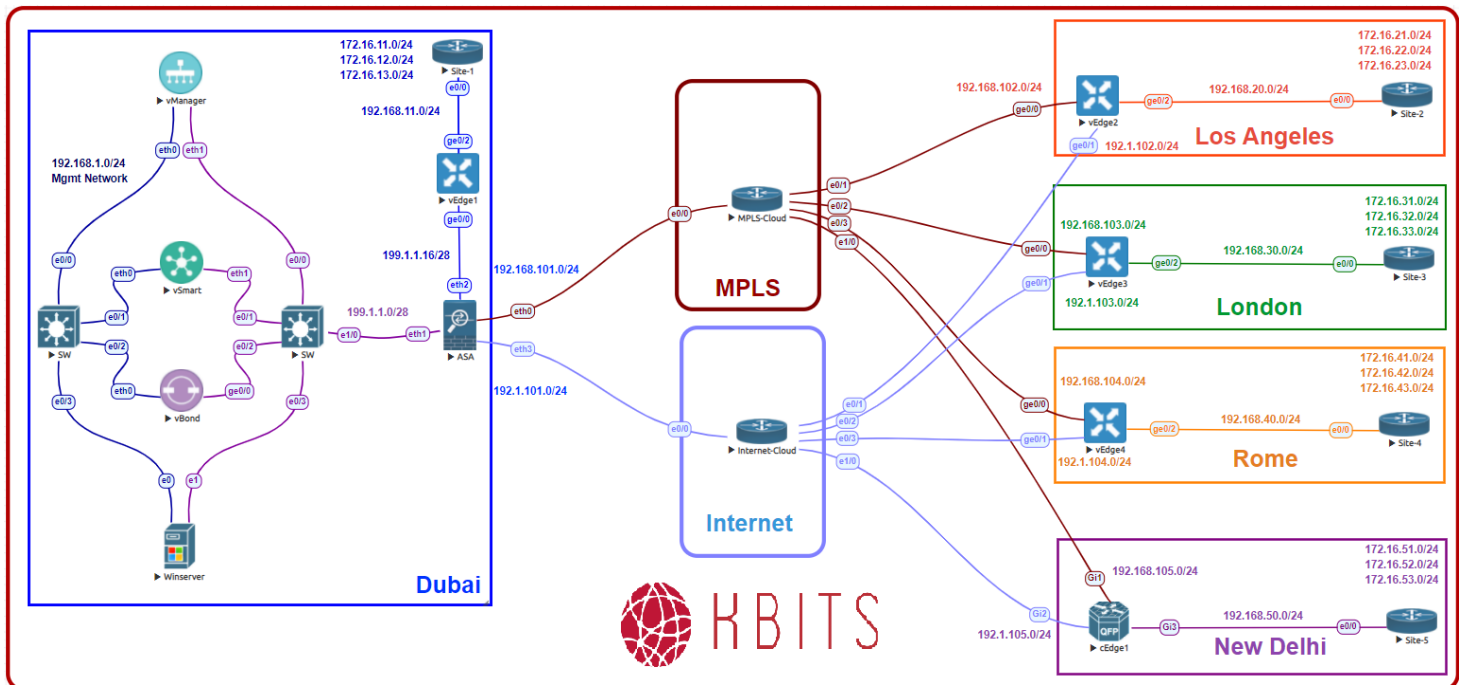
- Accept
- Click **Save Match and Actions** to save the Sequence.
- Save the Policy

## **Task 3 – Modify the existing Centralized Policy “Main-Central-Policy” and call the Topology Policy**

- In vManage, Navigate to **Configuration -> Policies -> Custom Options -> Centralized Policy -> Main-Central-Policy -> Click “...” -> Edit**.
- Click **Topology** on the **Top** of the page.
- Click **Add Topology**.
- Click **“Import Existing”** and select the **LA-MPLS-INT** from the drop-down list and click **Import**.

- Click **Policy Application** on the **Top** of the page.
- Click the “**Topology**” tab.
- The **LA-MPLS-INT-Policy** will be there. Click “**New Site**” button.
- Select **Rome** in the Outbound Site List.
- Click **Add**.
- Click the **Save Policy** button towards the button.
- **Activate** the policy.
- Wait for it to push the policy to the reachable vSmart Controller(s).
- Verify by using the **Show IP route vpn 1** command on the Rome vEdge (vEdge4).
- It should only have 1 TLOC for Los Angeles routes (10.2.2.202 – MPLS), whereas it will have 2 TLOCs for London (10.2.2.203-MPLS, 10.2.2.203-Biz-Internet).

# Lab 30 - Configuring Route Filtering



## Requirements:

- The 172.16.234.2/32, 172.16.234.3/24 & 172.16.234.4/24 should not be propagated to the Dubai Site.

## Task 1 - Configure Groups of Interests/List that will be used for Route Filtering Policy for Dubai

- In vManage, Navigate to **Configuration -> Policies -> Custom Options -> Centralized Policy -> Lists**.
- Click **Prefix** and select **New Prefix list**. Create a policy based on the following:
  - Name : **PL-234**
  - Prefix List Entry: **172.16.234.0/24 le 32**
- Click **Site** and select **New Site list**. Create a policy based on the following:
  - Name : **Dubai**
  - Site ID : **1**

## Task 2 – Configure Control/Topology policy based on the Requirements

- In vManage, Navigate to **Configuration -> Policies -> Custom Options -> Centralized Policy -> Topology**.
- Configure 1 Route Policy based on the following:
  - Policy Name : **PREF-234-NOT-2-DXB**
  - Description : **PREF-234-NOT-2-DXB**
  - Route Sequence**  
**Match Conditions:**
    - Prefix List: **PL-234**
    - Action: Reject**
  - Click **Save Match and Actions** to save the Sequence.
  - Default Sequence**  
**Action**
    - Accept
    - Click **Save Match and Actions** to save the Sequence.
  - Save the Policy

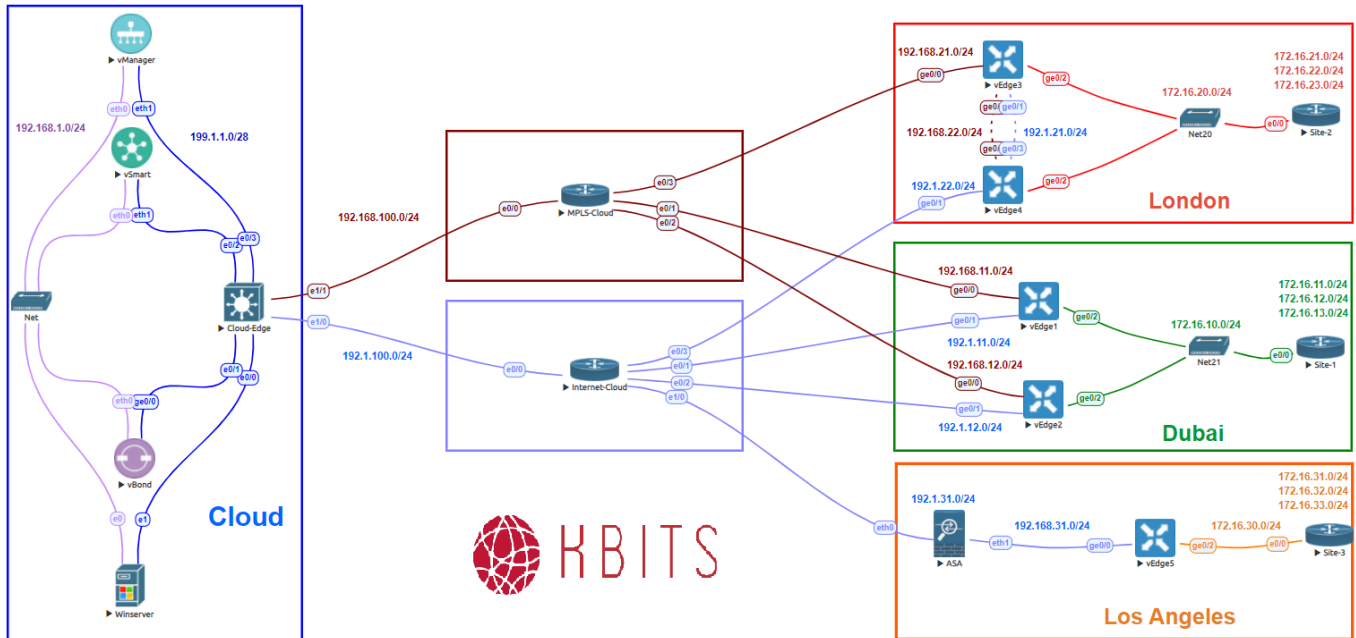
## Task 3 – Modify the existing Centralized Policy “Main-Central-Policy” and call the Topology Policy

- In vManage, Navigate to **Configuration -> Policies -> Custom Options -> Centralized Policy -> Main-Central-Policy -> Click “...” -> Edit**.
- Click **Topology** on the **Top** of the page.
- Click **Add Topology**.
- Click “**Import Existing**” and select the **PREF-234-NOT-2-DXB** from the drop-down list and click **Import**.
- Click **Policy Application** on the **Top** of the page.
- Click the “**Topology**” tab.
- The **PREF-234-NOT-2-DXB** will be there. Click “**New Site**” button.



- Select **Dubai** in the Outbound Site List.
- Click **Add**.
- Click the **Save Policy** button towards the button.
- **Activate** the policy.
- Wait for it to push the policy to the reachable vSmart Controller(s).
- Verify by using the **Show IP route vpn 1** command on the Dubai vEdge (vEdge1).
- It should all the routes from the Branches except the 172.16.234.X/32 routes.
- These routes should be present in the vEdge2, vEdge3 and vEdge4 routers. You can use the **Show IP route vpn 1** command to verify.

# Lab 31 – Configuring the WAN Components



## Interface Configuration

### Cloud Edge

Interface	IP Address	Subnet Mask
VLAN 199	199.1.1.14	255.255.255.240
E 1/0	192.1.100.1	255.255.255.0
E 1/1	192.168.100.1	255.255.255.0

### MPLS Cloud

Interface	IP Address	Subnet Mask
E 0/0	192.168.100.254	255.255.255.0
E 0/1	192.168.11.254	255.255.255.0
E 0/2	192.168.12.254	255.255.255.0
E 0/3	192.168.21.254	255.255.255.0

## Internet Cloud

Interface	IP Address	Subnet Mask
E 0/0	192.1.100.254	255.255.255.0
E 0/1	192.1.11.254	255.255.255.0
E 0/2	192.1.12.254	255.255.255.0
E 0/3	192.1.22.254	255.255.255.0
E 1/0	192.1.31.254	255.255.255.0

## WAN Setup

### Task 1 – Cloud Edge Router Configuration

- Configure the Interfaces based on the Logical Diagram
- Configure OSPF as the IGP to communicate with the MPLS Cloud. Enable all the interfaces.
- Make sure OSPF only sends and receives OSPF packets on the link towards the MPLS Cloud using the Passive-interface command.
- Configure a default route on the router towards the Internet. The IP Address of the Internet Router is 192.1.100.254

## Cloud Edge Router

```
no ip domain-lookup
line con 0
  logging synchronous
  exec-timeout 0 0
!
hostname Cloud Edge
!
vlan 199
!
interface range E 0/0-3
  switchport mode access
  switchport access vlan 199
!
interface VLAN 199
  ip address 199.1.1.14 255.255.255.240
  no shutdown
!
interface E 1/0
  ip address 192.1.100.1 255.255.255.240
  no shutdown
!
interface E 1/1
  ip address 192.168.100.1 255.255.255.0
  no shutdown
!
router ospf 1
  network 192.168.100.0 0.0.0.255 area 0
  network 199.1.1.0 0.0.0.255 area 0
  passive-interface default
  no passive-interface E 1/1
!
ip route 0.0.0.0 0.0.0.0 192.1.100.254
```

## Task 2 – MPLS Cloud Router Configuration

- Configure the Interfaces based on the Logical Diagram.
- Configure OSPF as the IGP on all the interfaces.

### MPLS Cloud Router

```
no ip domain-lookup
!
line con 0
  exec-timeout 0 0
  logging synchronous
!
hostname MPLS
!
interface Ethernet0/0
  ip address 192.168.100.254 255.255.255.0
  no shut
!
interface Ethernet0/1
  ip address 192.168.11.254 255.255.255.0
  no shut
!
interface Ethernet0/2
  ip address 192.168.12.254 255.255.255.0
  no shut
!
interface Ethernet0/3
  ip address 192.168.21.254 255.255.255.0
  no shut
!
router ospf 1
  network 192.168.100.0 0.0.0.255 area 0
  network 192.168.11.0 0.0.0.255 area 0
  network 192.168.12.0 0.0.0.255 area 0
  network 192.168.21.0 0.0.0.255 area 0
```

### Task 3 – Internet Cloud Router Configuration

- Configure the Interfaces based on the Logical Diagram
- Configure a Static Route on the Router for the 199.1.1.0/24 network. The Next Hop should point towards the Internet IP of the HQ Router.

#### Internet Cloud Router

```
no ip domain lookup
!
line con 0
  exec-timeout 0 0
  logging synchronous
!
hostname Internet
!
interface Ethernet0/0
  ip address 192.1.100.254 255.255.255.0
  no shut
!
interface Ethernet0/1
  ip address 192.1.11.254 255.255.255.0
  no shut
!
interface Ethernet0/2
  ip address 192.1.12.254 255.255.255.0
  no shut
!
interface Ethernet0/3
  ip address 192.1.22.254 255.255.255.0
  no shut
!
interface Ethernet1/0
  ip address 192.1.31.254 255.255.255.0
  no shut
!
ip route 199.1.1.0 255.255.255.0 192.1.100.1
```

# Server Setup

## Task 1 – Configure the Interfaces

### First Ethernet Interface:

IP Address: 192.168.1.5  
Subnet Mask: 255.255.255.0

### Third Ethernet Interface:

IP Address: 199.1.1.5  
Subnet Mask: 255.255.255.240  
Default Gateway: 199.1.1.14

## Task 2 – Configure the Timezone and Time

Configure the appropriate Timezone and Time on the Windows Server.

## Task 3 – Installing the Enterprise Root Certificate Server

- Open **Server Manager**
- Click **Roles**
- Click **Add Roles**
- Click **Next**
- Select the "**Active Directory Certificate Services**" and click **Next**
- Click **Next**
- Select "**Certification Authority Web Enrollment**" and click **Next**
- Leave it as Standalone and click **Next**
- Leave it as Root CA and click **Next**
- Leave "Create a new private key" and click **Next**
- Leave the default for the Cryptography for CA and click **Next**
- Set the Common name as **KBITS-CA** and click **Next**
- Leave the default for the Validity Period and click **Next**
- Click **Next**
- Click **Install**

## Task 4 – Install WinSCP

- **Double-click** the WinSCP Installation file.
- Do a Default Installation.

# Controller Setup – vManage

## Task 1 – Configuring the System Component

- Configure the System parameters based on the following:
  - Host-name : vManage1
  - Organization: KBITS
  - System-IP: 10.1.1.101
  - Site ID: 1
  - vbond Address: 199.1.1.3
  - Timezone: Based on the appropriate Timezone

**Note:** Default username: admin Default password: admin

### vManage

```
config
!  
system  
host-name vManage1  
system-ip 10.1.1.101  
site-id 1  
organization-name KBITS  
clock timezone Asia/Muscat  
vbond 199.1.1.3  
!
```

### Commit

## Task 2 – Configured the vpn parameters

- Configure the VPN parameters based on the following:
  - vpn 0
    - Interface eth1
    - IP Address: 199.1.1.1/28
    - Tunnel Interface
    - Tunnel Services (All, NetConf, SSHD)
    - Default Route: 199.1.1.14



- vpn 512
  - Interface eth0
  - IP Address: 192.168.1.1/24

### **vManage**

```
config
!
vpn 0
no interface eth0
interface eth1
ip address 199.1.1.1/28
tunnel-interface
allow-service all
allow-service netconf
allow-service sshd
no shut
ip route 0.0.0.0/0 199.1.1.14
!
vpn 512
interface eth0
ip address 192.168.1.1/24
no shut
!
commit
```

### **Task 3 – Organization name & vBond Address**

- Log into the vManage from the Server by browsing to <https://192.168.1.1:8443> using a username of **admin** and a password of **admin**.
- Navigate to **Administration -> Settings**
- Click **Edit** on the Organization name and set it to **KBITS**. Confirm the Organization name. Click **OK**.
- Click **Edit** on the **vBond** address and change it to 199.1.1.3. Confirm and click **OK**.

#### **Task 4 – Configure Controller Authorization as Enterprise Root and Download the Root Certificate.**

- Browse to <http://192.168.1.5/certsrv>
- Click **“Download Root Certificate”**.
- Select **“Base 64”**.
- Click **“Download CA Certificate”**.
- Open Explorer and navigate to the downloads folder.
- Change the name of the Downloaded file **“Certnew”** to **“RootCert”**.
- Open the **“RootCert.cer”** file using Notepad.
- Copy using **CTRL-A** and **CTRL-C**.
- In vManage, Navigate to **Administration -> Settings -> Controller Certificate Authorization**.
- Change the **“Certificate Signing by:”** to **“Enterprise Root Certificate”**.
- Paste the RootCert.cer that you had copied by using **CTRL-V**.
- Set the CSR Parameters with the Organization name, City, State, Country. Set the Time to 3 Years and save.

#### **Task 5 – Generate a CSR for vManage**

- Navigate to **Configuration -> Certificates -> Controllers -> vManage -> Generate CSR**.
- It will open a window with the CSR. Copy by using **CTRL-A** and **CTRL-C**.

#### **Task 6 – Request a Certificate from the CA Server**

- Browse to <http://192.168.1.5/certsrv>
- Click **“Request a Certificate”**.

- Select **“Advanced”**.
- Paste the CSR in the box by using **CTRL-V** and click **Submit**.

### **Task 7 – Issue the Certificate from the CA Server**

- Open Server Manager and navigate to **Active Directory Certificate Server -> KBITS-CA -> Pending Requests**.
- Right-Click the request and click **“Issue”**.

### **Task 8 – Downloading the Issued Certificate**

- Browse to <http://192.168.1.5/certsrv>
- Click **“Check on Pending request”**.
- The issued certificate link will show up. Click on the link.
- Select **“Base 64”** and click **“Download”**
- Open Explorer and navigate to the downloads folder.
- Change the name of the Downloaded file **“Certnew”** to **“vManage”**.
- Open the **“vManage.cer”** file using Notepad.
- Copy using **CTRL-A** and **CTRL-C**.

### **Task 9 – Installing the Identity Certificate for vManage**

- In vManage, Navigate to **Configuration -> Certificates -> Controllers**
- Click on the **“Install”** button at the top right corner
- Paste the Certificate (CTRL-V).
- The Identity certificate should be installed on vManage.

# Controller Setup – vBond

## Task 1 – Configuring the System Component

- Configure the System parameters based on the following:
  - Host-name : vBond1
  - Organization: KBITS
  - System-IP: 10.1.1.103
  - Site ID: 1
  - vbond Address: 199.1.1.3
  - Timezone: Based on the appropriate Timezone

**Note: Default username:** admin **Default password:** admin

### vBond

```
config
!  
system
host-name vBond1
system-ip 10.1.1.103
site-id 1
organization-name KBITS
clock timezone Asia/Muscat
vbond 199.1.1.3 local
!
```

### Commit

## Task 2 – Configured the vpn parameters

- Configure the VPN parameters based on the following:
  - vpn 0
    - Interface ge0/0
    - IP Address: 199.1.1.3/28
    - Tunnel Interface
    - Tunnel Services (All, NetConf, SSHD)
    - Encapsulation: IPsec
    - Default Route: 199.1.1.14
  - vpn 512
    - Interface eth0
    - IP Address: 192.168.1.3/24

## **vBond**

```
config
!  
vpn 0  
no interface eth0  
interface ge0/0  
ip address 199.1.1.3/28  
tunnel-interface  
encapsulation ipsec  
allow-service all  
allow-service netconf  
allow-service sshd  
no shut  
ip route 0.0.0.0/0 199.1.1.14  
!  
vpn 512  
interface eth0  
ip address 192.168.1.3/24  
no shut  
!  
Commit
```

### **Task 3 – Add vBond to vManage**

- Navigate to **Configuration -> Devices -> Controllers -> Add Controllers -> vBond** and specify the following to add the vBond in vManage.
  - IP Address: **199.1.1.3**
  - Username: **Admin**
  - Password: **Admin**
  - Check Generate CSR
  - Click **OK**

### **Task 4 – View the generated CSR for vBond and Copy it**

- Navigate to **Configuration -> Certificates -> Controllers -> vBond -> View CSR.**
- It will open a window with the CSR. Copy by using **CTRL-A** and **CTRL-C.**

### **Task 5 – Request a Certificate from the CA Server**

- Browse to <http://192.168.1.5/certsrv>
- Click **“Request a Certificate”**.
- Select **“Advanced”**.
- Paste the CSR in the box by using **CTRL-V** and click **Submit**.

### **Task 6 – Issue the Certificate from the CA Server**

- Open Server Manager and navigate to **Active Directory Certificate Server -> KBITS-CA -> Pending Requests**.
- Right-Click the request and click **“Issue”**.

### **Task 7 – Downloading the Issued Certificate**

- Browse to <http://192.168.1.5/certsrv>
- Click **“Check on Pending request”**.
- The issued certificate link will show up. Click on the link.
- Select **“Base 64”** and click **“Download”**
- Open Explorer and navigate to the downloads folder.
- Change the name of the Downloaded file **“Certnew”** to **“vBond”**.
- Open the **“vBond.cer”** file using Notepad.
- Copy using **CTRL-A** and **CTRL-C**.

### **Task 8 – Installing the Identity Certificate for vManage**

- In vManage, Navigate to **Configuration -> Certificates -> Controllers**
- Click on the **“Install”** button at the top right corner
- Paste the Certificate (CTRL-V).

- The Identity certificate should be installed for vBond and pushed to it.

## Controller Setup – vSmart

### Task 1 – Configuring the System Component

- Configure the System parameters based on the following:
  - Host-name : vSmart1
  - Organization: KBITS
  - System-IP: 10.1.1.102
  - Site ID: 1
  - vbond Address: 199.1.1.3
  - Timezone: Based on the appropriate Timezone

**Note:** Default username: admin Default password: admin

#### vSmart

```
config
!  
system
host-name vSmart1
system-ip 10.1.1.102
site-id 1
organization-name KBITS
clock timezone Asia/Muscat
vbond 199.1.1.3
!  
Commit
```

### Task 2 – Configured the vpn parameters

- Configure the VPN parameters based on the following:
  - vpn 0
    - Interface Eth1
    - IP Address: 199.1.1.2/28
    - Tunnel Interface
    - Tunnel Services (All, NetConf, SSHD)
    - Default Route: 199.1.1.14
  - vpn 512
    - Interface eth0

- IP Address: 192.168.1.2/24

### **vSmart**

```
config
!  
vpn 0  
no interface eth0  
interface eth1  
ip address 199.1.1.2/28  
tunnel-interface  
allow-service all  
allow-service netconf  
allow-service sshd  
no shut  
ip route 0.0.0.0/0 199.1.1.14  
!  
vpn 512  
interface eth0  
ip address 192.168.1.2/24  
no shut  
!  
Commit
```

### **Task 3 – Add vSmart to vManage**

- Navigate to **Configuration -> Devices -> Controllers -> Add Controllers -> vSmart** and specify the following to add the vBond in vManage.
  - IP Address: **199.1.1.2**
  - Username: **Admin**
  - Password: **Admin**
  - Check Generate CSR
  - Click **OK**

### **Task 4 – View the generated CSR for vBond and Copy it**

- Navigate to **Configuration -> Certificates -> Controllers -> vSmart -> View CSR.**
- It will open a window with the CSR. Copy by using **CTRL-A** and **CTRL-C.**

### **Task 5 – Request a Certificate from the CA Server**



- Browse to <http://192.168.1.5/certsrv>
- Click **“Request a Certificate”**.
- Select **“Advanced”**.
- Paste the CSR in the box by using **CTRL-V** and click **Submit**.

### **Task 6 – Issue the Certificate from the CA Server**

- Open Server Manager and navigate to **Active Directory Certificate Server -> KBITS-CA -> Pending Requests**.
- Right-Click the request and click **“Issue”**.

### **Task 7 – Downloading the Issued Certificate**

- Browse to <http://192.168.1.5/certsrv>
- Click **“Check on Pending request”**.
- The issued certificate link will show up. Click on the link.
- Select **“Base 64”** and click **“Download”**
- Open Explorer and navigate to the downloads folder.
- Change the name of the Downloaded file **“Certnew”** to **“vSmart”**.
- Open the **“vSmart.cer”** file using Notepad.
- Copy using **CTRL-A** and **CTRL-C**.

### **Task 8 – Installing the Identity Certificate for vManage**

- In vManage, Navigate to **Configuration -> Certificates -> Controllers**
- Click on the **“Install”** button at the top right corner
- Paste the Certificate (CTRL-V).

- The Identity certificate should be installed for vSmart and pushed to it.

## WAN Edge Setup – (CLI)

### Task 1 – Upload the WAN Edge List

- On the vManage Main windows, Naviagte to **Configuration -> Devices**. Click on “**Upload WAN Edge List**”.
- Select the file you downloaded from the PNP Portal. Upload it and check the **Validate** option.

## vEDGE-1

### Task 1 – Configuring the System Component

- Configure the System parameters based on the following:
  - Host-name : vEdge1
  - Organization: KBITS
  - System-IP: 10.2.2.201
  - Site ID: 1
  - vbond Address: 199.1.1.3
  - Timezone: Based on the appropriate Timezone

**Note: Default username:** admin **Default password:** admin

#### vEdge1

```
config
!  
system  
host-name vEdge1  
system-ip 10.2.2.201  
site-id 1  
organization-name KBITS  
clock timezone Asia/Muscat  
vbond 199.1.1.3  
!  
commit
```

## Task 2 – Configure the vpn parameters

- Configure the VPN parameters based on the following:
  - vpn 0
    - Interface ge0/0
    - IP Address: 192.168.11.1/24
    - Tunnel Interface
    - Encapsulation IPSec
    - Tunnel Services (All, NetConf, SSHD)
    - Default Route: 192.168.11.254
  - vpn 512
    - Interface eth0
    - IP Address: DHCP Client

### vEdge1

```
config
!
vpn 0
no interface eth0
interface ge0/0
ip address 192.168.11.1/24
tunnel-interface
encapsulation ipsec
allow-service all
allow-service netconf
allow-service sshd
no shut
ip route 0.0.0.0/0 192.168.11.254
!
vpn 512
interface eth0
ip dhcp-client
no shutdown
commit
```

## vEDGE-2

### Task 1 – Configuring the System Component

- Configure the System parameters based on the following:
  - Host-name : vEdge2
  - Organization: KBITS
  - System-IP: 10.2.2.202
  - Site ID: 1
  - vbond Address: 199.1.1.3
  - Timezone: Based on the appropriate Timezone

**Note: Default username:** admin **Default password:** admin

#### vEdge-2

```
config
!  
system
  host-name vEdge2
  system-ip 10.2.2.202
  site-id 1
  organization-name KBITS
  clock timezone Asia/Muscat
  vbond 199.1.1.3
!  
commit
```

### Task 2 – Configure the vpn parameters

- Configure the VPN parameters based on the following:
  - vpn 0
    - Interface ge0/0
    - IP Address: 192.168.12.2/24
    - Tunnel Interface
    - Encapsulation IPSec
    - Tunnel Services (All, NetConf, SSHD)
    - Default Route: 192.168.12.254
  - vpn 512
    - Interface eth0
    - IP Address: DHCP Client

## vEdge2

```
config
!
vpn 0
no interface eth0
interface ge0/0
ip address 192.168.12.2/24
tunnel-interface
encapsulation ipsec
allow-service all
allow-service netconf
allow-service sshd
no shut
ip route 0.0.0.0/0 192.168.102.254
!
vpn 512
interface eth0
ip dhcp-client
no shutdown
!
commit
```

## vEDGE-3

### Task 1 – Configuring the System Component

- Configure the System parameters based on the following:
  - Host-name : vEdge3
  - Organization: KBITS
  - System-IP: 10.2.2.203
  - Site ID: 2
  - vbond Address: 199.1.1.3
  - Timezone: Based on the appropriate Timezone

**Note: Default username:** admin **Default password:** admin

## vEdge-3

```
config
!
system
```

```
host-name vEdge3
system-ip 10.2.2.203
site-id 2
organization-name KBITS
clock timezone Asia/Muscat
vbond 199.1.1.3
!
Commit
```

## Task 2 – Configure the vpn parameters

- Configure the VPN parameters based on the following:
  - vpn 0
    - Interface ge0/0
    - IP Address: 192.168.21.3/24
    - Tunnel Interface
    - Encapsulation IPSec
    - Tunnel Services (All, NetConf, SSHD)
    - Default Route: 192.168.21.254
  - vpn 512
    - Interface eth0
    - IP Address: DHCP Client

### vEdge3

```
config
!
vpn 0
no interface eth0
interface ge0/0
ip address 192.168.21.3/24
tunnel-interface
encapsulation ipsec
allow-service all
allow-service netconf
allow-service sshd
no shut
ip route 0.0.0.0/0 192.168.21.254
!
vpn 512
interface eth0
ip dhcp-client
no shutdown
!
```

**Commit**

## vEDGE-4

### Task 1 – Configuring the System Component

- Configure the System parameters based on the following:
  - Host-name : vEdge4
  - Organization: KBITS
  - System-IP: 10.2.2.204
  - Site ID: 2
  - vbond Address: 199.1.1.3
  - Timezone: Based on the appropriate Timezone

**Note: Default username:** admin **Default password:** admin

### vEdge-4

```
config
!
system
 host-name vEdge4
 system-ip 10.2.2.204
 site-id 2
 organization-name KBITS
 clock timezone Asia/Muscat
 vbond 199.1.1.3
!
Commit
```

### Task 2 – Configure the vpn parameters

- Configure the VPN parameters based on the following:
  - vpn 0
    - Interface Ge0/1
    - IP Address: 192.1.22.4/24
    - Tunnel Interface
    - Encapsulation IPSec
    - Tunnel Services (All, NetConf, SSHD)
    - Default Route: 192.168.22.254
  - vpn 512

- Interface eth0
- IP Address: DHCP Client

#### **vEdge4**

```
config
!  
vpn 0  
no interface eth0  
interface ge0/1  
ip address 192.1.22.4/24  
tunnel-interface  
encapsulation ipsec  
allow-service all  
allow-service netconf  
allow-service sshd  
no shut  
ip route 0.0.0.0/0 192.168.22.254  
!  
vpn 512  
interface eth0  
ip dhcp-client  
no shutdown  
!  
Commit
```



# WAN Edge Setup – vManage (GUI)

## vEDGE-1

### Task 1 – Upload the Root Certificate to the vEdge

- On the Windows Server, open **WINSCP** application.
- **Connect** to vEdge1 using the following information:
  - IP Address : 192.168.11.1
  - Protocol - SFTP
  - Username : admin
  - Password : admin
- Copy the **RootCert.cer** file from the Downloads folder to the **/home/admin** folder on the vEdge1

### Task 2 – Install the Root Certificate on vEdge1

- Connect to the console of vEdge1 and issue the following command:

```
request root-cert-chain install /home/admin/RootCert.cer
```

### Task 3 - Activate vEdge on vManage

- Navigate to **Configuration -> Devices**
- Note and use the **Chassis Number** and **Token number** for the 1st vEdge from vManage.
- Use the information from the previous step in the following command on the vEdge1 console.

```
request vedge-cloud activate chassis-number XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX token XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

- You should see the vEdge in the vManage console with a Certificate issued.

## vEDGE-2

### Task 1 – Upload the Root Certificate to the vEdge

- On the Windows Server, open **WINSCP** application.
- **Connect** to vEdge1 using the following information:
  - IP Address : 192.168.12.2
  - Protocol - SFTP
  - Username : admin
  - Password : admin
- Copy the **RootCert.cer** file from the Downloads folder to the **/home/admin** folder on the vEdge2

### Task 2 – Install the Root Certificate on vEdge2

- Connect to the console of vEdge2 and issue the following command:

```
request root-cert-chain install /home/admin/RootCert.cer
```

### Task 3 - Activate vEdge on vManage

- Navigate to **Configuration -> Devices**
- Note and use the **Chassis Number** and **Token number** for the 2<sup>nd</sup> vEdge from vManage.
- Use the information from the previous step in the following command on the vEdge2 console.

```
request vedge-cloud activate chassis-number XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX token  
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

- You should see the vEdge in the vManage console with a Certificate issued.



## vEDGE-4

### Task 1 – Upload the Root Certificate to the vEdge

- On the Windows Server, open **WINSCP** application.
- **Connect** to vEdge1 using the following information:
  - IP Address : 192.1.22.4
  - Protocol - SFTP
  - Username : admin
  - Password : admin
- Copy the **RootCert.cer** file from the Downloads folder to the **/home/admin** folder on the vEdge4

### Task 2 – Install the Root Certificate on vEdge4

- Connect to the console of vEdge4 and issue the following command:

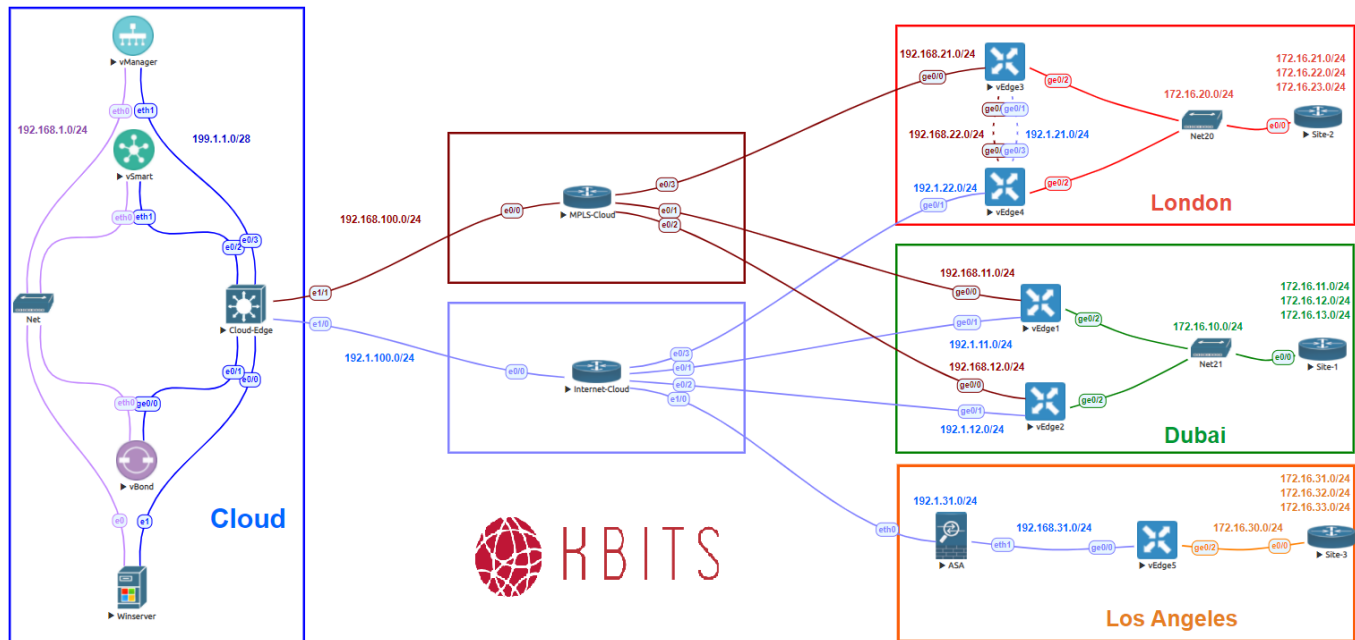
```
request root-cert-chain install /home/admin/RootCert.cer
```

### Task 3 - Activate vEdge on vManage

- Navigate to **Configuration -> Devices**
- Note and use the **Chassis Number** and **Token number** for the 4<sup>th</sup> vEdge from vManage.
- Use the information from the previous step in the following command on the vEdge4 console.

```
request vedge-cloud activate chassis-number XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX token  
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```
- You should see the vEdge in the vManage console with a Certificate issued.

# Lab 32 – Configuring Firewalls to supports SD-WAN



## Interface Configuration

### ASA v1

Interface	IP Address	Subnet Mask
Gig 0/0	192.1.31.10	255.255.255.0
Gig 0/1	192.168.31.10	255.255.255.0

## Firewall Configuration

### Task 1 – Interface Configuration and Default Routing on ASA in Los Angeles

- Configure the Interfaces based on the Logical Diagram.
- Configure Gig 0/0 with a Name of **“Outside”** using the default Security Level.
- Configure Gig 0/1 with a Name of **“Inside”** using the default Security Level.
- Configure a default route on the ASA pointing towards the Internet Cloud thru the Outside Interface.

### **ASAv Firewall**

```
Hostname ASAv1
!  
Interface Gig 0/0  
  Nameif Outside  
  ip address 192.1.31.10 255.255.255.0  
  no shut  
!  
Interface Gig 0/1  
  Nameif Inside  
  ip address 192.168.31.10 255.255.255.240  
  no shut  
!  
Route Outside 0.0.0.0 0.0.0.0 192.1.31.254
```

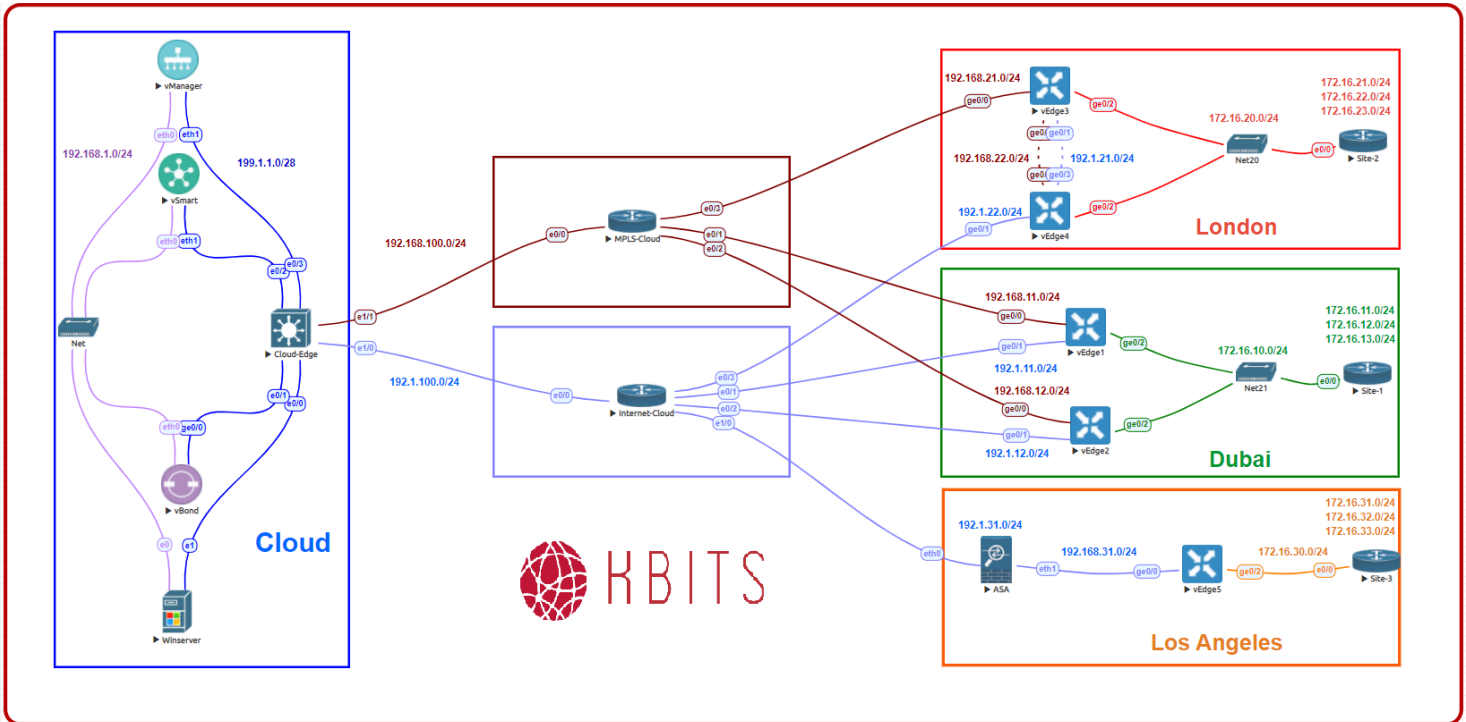
### **Task 2 – Translate vEdge5 on the Outside.**

- Statically Translate vEdge5 as 192.1.31.5 on the Outside.
- The Private address that will be assigned to vEdge5 is 192.168.31.5.

### **ASAv Firewall**

```
Object network vEdge5  
  Host 192.168.31.5  
  Nat (Inside,Outside) static 192.1.31.5
```

# Lab 33 – Configuring vEdges with NAT thru Firewalls



## vEDGE-5 Initialization – (CLI)

### Task 1 – Configuring the System Component on vEdge5

- Configure the System parameters based on the following:
  - Host-name : vEdge5
  - Organization: KBITS
  - System-IP: 10.2.2.205
  - Site ID: 3
  - vbond Address: 199.1.1.3
  - Timezone: Based on the appropriate Timezone

**Note: Default username:** admin **Default password:** admin

#### vEdge-5

```
config
!
```

```
system
host-name vEdge5
system-ip 10.2.2.205
site-id 3
organization-name KBITS
clock timezone Asia/Muscat
vbond 199.1.1.3
!
Commit
```

## **Task 2 – Configure the vpn parameters on vEdge5**

- Configure the VPN parameters based on the following:
  - vpn 0
    - Interface ge0/0
    - IP Address: 192.168.31.5/24
    - Tunnel Interface
    - Encapsulation IPsec
    - Tunnel Services (All, NetConf, SSHD)
    - Default Route: 192.168.31.10
  - vpn 512
    - Interface eth0
    - IP Address: DHCP Client

### **vEdge5**

```
config
!
vpn 0
no interface eth0
interface ge0/0
ip address 192.168.31.5/24
tunnel-interface
encapsulation ipsec
allow-service all
allow-service netconf
allow-service sshd
no shut
ip route 0.0.0.0/0 192.168.31.10
!
vpn 512
interface eth0
ip dhcp-client
no shutdown
```



!  
**Commit**

## vEDGE-5 Initialization – vManage(GUI)

### Task 1 – Configure an ACL on ASAv to allow the Certificate server to SFTP to vEdge5 to upload the Root Certificate

#### ASAv

```
Access-list OUTSIDE permit tcp host 199.1.1.5 host 192.168.31.5 eq 22  
!  
Access-group OUTSIDE in interface Outside
```

### Task 2 – Upload the Root Certificate to the vEdge

- On the Windows Server, open **WINSCP** application.
- **Connect** to vEdge5 using the following information:
  - IP Address : 192.1.31.5
  - Protocol - SFTP
  - Username : admin
  - Password : admin
- Copy the **RootCert.cer** file from the Downloads folder to the **/home/admin** folder on the vEdge3

### Task 3 – Install the Root Certificate on vEdge3

- Connect to the console of vEdge5 and issue the following command:

**request root-cert-chain install /home/admin/RootCert.cer**

### Task 4 - Activate vEdge on vManage

- Navigate to **Configuration -> Devices**
- Note and use the **Chassis Number** and **Token number** for the 5<sup>th</sup> vEdge from vManage.
- Use the information from the previous step in the following command on the vEdge5 console.



- VPN -> Global : **0**
- Name -> Global : **Transport VPN**

#### IPv4 Route

- Prefix -> Global : 0.0.0.0/0
- Next Hop -> Device Specific

- Click **Save** to save the Template.

### Task 2 – Configure a VPN Interface Template to be used by all BR3 vEdge-Cloud Devices for VPN 0 for Interface G0/0

- In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> VPN -> VPN Interface Ethernet**

- Configure the VPN parameters based on the following:

- Template Name : **BR3-VE-VPNINT-VPN0-GO**
- Description : **BR3-VE-VPNINT-VPN0-GO**

#### Basic Configuration

- Shutdown -> Global : **No**
- Interface Name -> Global : **ge0/0**
- IPv4 Address -> Static -> Device Specific

#### Tunnel

- Tunnel Interface -> Global : **On**
- Color -> Global : **Biz-internet**

#### Allow Service

- All -> Global : **On**
- NETCONF -> Global : **On**
- SSH -> Global : **On**

- Click **Save** to save the Template.

## VPN512

### Task 1 – Configure a VPN Template to be used by all Branch vEdge-Cloud Devices for VPN 512

- In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> VPN -> VPN**

- Configure the VPN parameters based on the following:

- Template Name : **BR-VE-VPN-VPN512**
- Description : **BR-VE-VPN-VPN512**

#### **Basic Configuration**

- VPN -> Global : **512**
- Name -> Global : **MGMT VPN**

- Click **Save** to save the Template.

### **Task 2 – Configure a VPN Interface Template to be used by all Branch vEdge-Cloud Devices for VPN 512 for Interface Eth0**

- In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> VPN -> VPN Interface Ethernet**

- Configure the VPN parameters based on the following:

- Template Name : **BR-VE-VPNINT-VPN512-E0**
- Description : **BR-VE-VPNINT-VPN512-E0**

#### **Basic Configuration**

- Shutdown -> Global : **No**
- Interface Name -> Global : **eth0**
- IPv4 Address -> Dynamic

- Click **Save** to save the Template

## **VPN 1**

### **Task 1 – Configure a VPN Template to be used by all Branch vEdge-Cloud Devices for VPN 1**

- In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> VPN -> VPN**

- Configure the VPN parameters based on the following:

- Template Name : **BR-VE-VPN-VPN1**
- Description : **BR-VE-VPN-VPN1**

#### **Basic Configuration**

- VPN -> Global : **1**
- Name -> Global : **Data VPN**

- Click **Save** to save the Template.

## Task 2 – Configure a VPN Interface Template to be used by all Branch vEdge-Cloud Devices for VPN 1 for Interface G0/2

- In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> VPN -> VPN Interface Ethernet**
- Configure the VPN parameters based on the following:
  - Template Name : **BR-VE-VPNINT-VPN1-G2**
  - Description : **BR-VE-VPNINT-VPN1-G2**
- Basic Configuration**
  - Shutdown -> Global : **No**
  - Interface Name -> Global : **ge0/2**
  - IPv4 Address -> Static -> Device Specific
- Click **Save** to save the Template.

## Task 3 – Configure a OSPF Template to be used by all Branch vEdge-Cloud Devices for VPN 1

- In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> Other Templates -> OSPF**
- Configure the OSPF parameters based on the following:
  - Template Name : **BR-VE-OSPF-VPN1**
  - Description : **BR-VE-OSPF-VPN1**
- Redistribution**
  - Protocol : **OMP**
- Area Configuration**
  - Area Number -> Global : **0**
  - Area Type -> Default
- Interface Configuration**
  - Interface Name: **ge0/2**
- Click **Add** to add the Interface and Click **Add** to add OSPF.
- Click **Save** to save the Template.

# vEdge5 Templates Deployment

## Task 1 – Configure a Device Template for BR3 vEdge Devices.

- In vManage, Navigate to Configuration -> **Templates -> Device -> Create Template -> vEdge Cloud**
- Configure the Device Template based on the following:
  - Template Name : **BR3-VE-TEMP**
  - Description : **BR3-VE-TEMP**
  - Basic Information**
    - System -> **VE-System**
  - Transport & Management**
    - VPN 0 : **BR3-VE-VPN-VPN0**
    - VPN Interface : **BR3-VE-VPNINT-VPN0-G0**
  
    - VPN 512 : **BR-VE-VPN-VPN512**
    - VPN Interface : **BR-VE-VPNINT-VPN512-E0**
  - Service VPN**
    - VPN 1 : **BR-VE-VPN-VPN1**
    - VPN Interface : **BR-VE-VPNINT-VPN1-G2**
    - OSPF: **BR-VE-OSPF-VPN1**
- Click **Save** to save the Template.

## Task 2 – Attach vEdge5 to the Device Template

- In vManage, Navigate to Configuration -> **Templates -> Device -> BR3-VE-TEMP.**
- Click on “...” towards the right-hand side.
- Click **Attach Devices.**
- Select **vEdge5** and click the “->” button.
- Click **Attach.**

### Task 3 – Configure the Variable Parameters for the Feature Templates

- **vEdge5** will appear in the window.
- Click on “...” towards the right-hand side.
- Click **Edit Device Template**.
- Configure the variables based on the following:
  - Default Gateway for VPN0 : **192.168.31.10**
  - Interface IP for ge0/0 :**192.168.31.5/24**
  - Interface IP for ge0/2 :**172.16.30.5/24**
  - Timezone: **America/Los\_Angeles**
  - Hostname : **vEdge-5**
  - System IP : **10.2.2.205**
  - Site ID : **3**
- Click **Update**.
- Verify the Configuration & Click **Configure Devices**.
- Wait for it to update the device. It should come back with Status of **Success**.

### Site-3 Internal Router Configuration

#### Site-3 Internal Router

```
No ip domain-lookup
!  
Hostname R3
!  
Interface E 0/0
Ip address 172.16.30.33 255.255.255.0
No shut
!  
Interface loopback1
Ip address 172.16.31.1 255.255.255.0
Ip ospf network point-to-point
!  
Interface loopback2
Ip address 172.16.32.1 255.255.255.0
Ip ospf network point-to-point
```

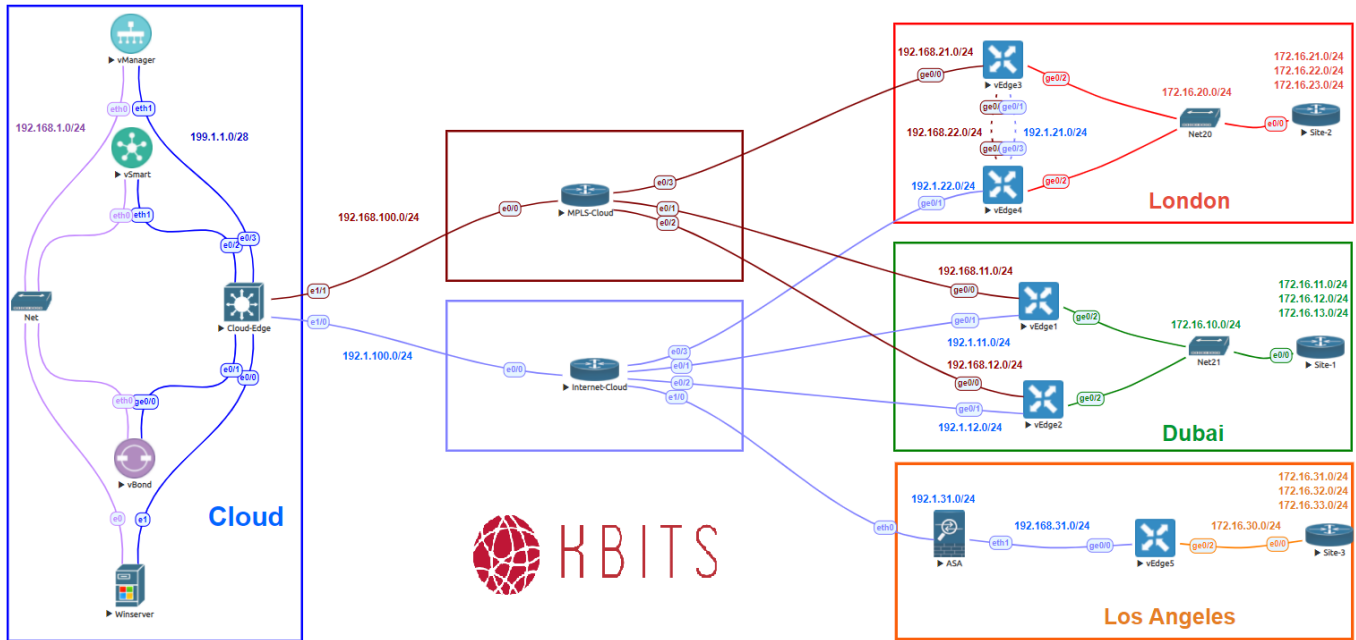
```
!  
Interface loopback3  
Ip address 172.16.33.1 255.255.255.0  
Ip ospf network point-to-point  
!  
Router ospf 1  
Network 172.16.0.0 0.0.255.255 area 0
```

## Verification

- Verify the configuration on **vEdge5**. You can do that by verify OSPF Neighbor relationship with the MPLS Router by issuing the **Show ospf neighbor** command on **vEdge5**.



# Lab 34 – Configuring TLOC Extensions



## vEdge3 Templates Creation

### VPN 0

#### Task 1 – Configure a VPN Template to be used by BR2 vEdges for VPN0

- In vManage, Navigate to Configuration -> **Templates** -> **Feature** -> **vEdge Cloud** -> **VPN** -> **VPN**
- Configure the VPN parameters based on the following:

- Template Name : **BR2-VE-VPN-VPN0**
- Description : **BR2-VE-VPN-VPN0**

#### Basic Configuration

- VPN -> Global : **0**
- Name -> Global : **Transport VPN**

#### IPv4 Route

- Prefix -> Global : 0.0.0.0/0
- Next Hop -> Device Specific

➤ Click **Save** to save the Template.

### **Task 2 – Configure a VPN Interface Template to be used by all BR2 vEdge-Cloud Devices for VPN 0 for Interface G0/0**

➤ In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> VPN -> VPN Interface Ethernet**

➤ Configure the VPN parameters based on the following:

- Template Name : **BR2-VE-VPNINT-VPN0-G0**
- Description : **BR2-VE-VPNINT-VPN0-G0**

#### **Basic Configuration**

- Shutdown -> Global : **No**
- Interface Name -> Global : **ge0/0**
- IPv4 Address -> Static -> Device Specific

#### **Tunnel**

- Tunnel Interface -> Global : **On**
- Color -> Global : **Mpls**

#### **Allow Service**

- All -> Global : **On**
- NETCONF -> Global : **On**
- SSH -> Global : **On**

➤ Click **Save** to save the Template.

### **Task 3 – Configure a VPN Interface Template to be used by all BR2 vEdge-Cloud Devices for VPN 0 for Interface G0/1**

➤ In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> VPN -> VPN Interface Ethernet**

➤ Configure the VPN parameters based on the following:

- Template Name : **BR2-VE-VPNINT-VPN0-G1**
- Description : **BR2-VE-VPNINT-VPN0-G1**

#### **Basic Configuration**

- Shutdown -> Global : **No**
- Interface Name -> Global : **ge0/1**

- IPv4 Address -> Static -> Device Specific

### **Tunnel**

- Tunnel Interface -> Global : **On**
- Color -> Global : **Biz-Internet**
- **Allow Service**
- All -> Global : **On**
- NETCONF -> Global : **On**
- SSH -> Global : **On**

- Click **Save** to save the Template.

## **Task 4 – Configure a Template that will be used for TLOC-Extension on BR2 vEdges**

- In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> VPN -> VPN Interface Ethernet**
- Configure the VPN parameters based on the following:
  - Template Name : **BR2-VE-VPNINT-VPNO-TLOC-G3**
  - Description : **BR2-VE-VPNINT-VPNO-TLOC-G3**

### **Basic Configuration**

- Shutdown -> Global : **No**
- Interface Name -> Global : **ge0/3**
- IPv4 Address -> Static -> Device Specific

### **Advanced**

- TLOC Extension: Device Specific

- Click **Save** to save the Template.

## **Task 5 – Configure a OSPF Template to be used by vEdge3 for VPN 0**

- In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> Other Templates -> OSPF**
- Configure the OSPF parameters based on the following:
  - Template Name : **BR2-VE-vEdge3-OSPF-VPNO**
  - Description : **BR-VE-vEdge3-OSPF-VPNO**

### **Area Configuration**

- Area Number -> Global : **0**

- Area Type -> Default
- **Interface Configuration**
- Interface Name: **ge0/0**
- Interface Name: **ge0/3**

➤ Click **Save** to save the Template.

## vEdge3 Templates Deployment

### Task 1 – Configure a Device Template for BR2 vEdge3.

➤ In vManage, Navigate to Configuration -> **Templates -> Device -> Create Template -> vEdge Cloud**

➤ Configure the Device Template based on the following:

- Template Name : **BR2-VE-vEdge3-TEMP**
- Description : **BR2-VE-vEdge3-TEMP**

#### Basic Information

- System -> **VE-System**

#### Transport & Management

- VPN 0 : **BR2-VE-VPN-VPN0**
- VPN Interface : **BR2-VE-VPNINT-VPN0-G0**
- VPN Interface : **BR2-VE-VPNINT-VPN0-G1**
- VPN Interface : **BR2-VE-VPNINT-VPN0-TLOC-G3**
- OSPF: **BR2-VE-vEdge3-OSPF-VPN0**

- VPN 512 : **BR-VE-VPN-VPN512**
- VPN Interface : **BR-VE-VPNINT-VPN512-E0**

#### Service VPN

- VPN 1 : **BR-VE-VPN-VPN1**
- VPN Interface : **BR-VE-VPNINT-VPN1-G2**
- OSPF: **BR-VE-OSPF-VPN1**

➤ Click **Save** to save the Template.

### Task 2 – Attach vEdge3 to the Device Template

➤ In vManage, Navigate to Configuration -> **Templates -> Device -> BR2-VE-vEdge3-TEMP**

➤ Click on “...” towards the right-hand side.

- Click **Attach Devices**.
- Select **vEdge3** and click the “ -> “ button.
- Click **Attach**.

### **Task 3 – Configure the Variable Parameters for the Feature Templates**

- **vEdge3** will appear in the window.
- Click on “...” towards the right-hand side.
- Click **Edit Device Template**.
- Configure the variables based on the following:
  - Default Gateway for VPN0 : **192.1.21.4**
  - Interface IP for ge0/0 : **192.168.21.3/24**
  - Interface IP for ge0/1 : **192.1.21.3/24**
  - Interface IP for ge0/2 : **172.16.20.3/24**
  - Interface IP for ge0/3 : **192.168.22.3/24**
  - TLOC Extension: **ge0/0**
  - Timezone: **Europe/London**
  - Hostname : **vEdge-3**
  - System IP : **10.2.2.203**
  - Site ID : **2**
- Click **Update**.
- Verify the Configuration & Click **Configure Devices**.
- Wait for it to update the device. It should come back with Status of **Success**.

# vEdge4 Templates Creation

## VPN 0

### Task 1 – Configure a OSPF Template to be used by vEdge4 for VPN 0

- In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> Other Templates -> OSPF**
- Configure the OSPF parameters based on the following:
  - Template Name : **BR2-VE-vEdge4-OSPF-VPN0**
  - Description : **BR-VE-vEdge4-OSPF-VPN0**
  - Area Configuration**
    - Area Number -> Global : **0**
    - Area Type -> Default
  - Interface Configuration**
    - Interface Name: **ge0/0**
- Click **Save** to save the Template.

# vEdge4 Templates Deployment

### Task 1 – Configure a Device Template for BR2 vEdge4.

- In vManage, Navigate to Configuration -> **Templates -> Device -> Create Template -> vEdge Cloud**
- Configure the Device Template based on the following:
  - Template Name : **BR2-VE-vEdge4-TEMP**
  - Description : **BR2-VE-vEdge4-TEMP**
  - Basic Information**
    - System -> **VE-System**
  - Transport & Management**
    - VPN 0 : **BR2-VE-VPN-VPN0**
    - VPN Interface : **BR2-VE-VPNINT-VPN0-G0**
    - VPN Interface : **BR2-VE-VPNINT-VPN0-G1**
    - VPN Interface : **BR2-VE-VPNINT-VPN0-TLOC-G3**
    - OSPF: **BR2-VE-vEdge4-OSPF-VPN0**

- VPN 512 : **BR-VE-VPN-VPN512**
- VPN Interface : **BR-VE-VPNINT-VPN512-E0**

#### **Service VPN**

- VPN 1 : **BR-VE-VPN-VPN1**
- VPN Interface : **BR-VE-VPNINT-VPN1-G2**
- OSPF: **BR-VE-OSPF-VPN1**

- Click **Save** to save the Template.

### **Task 2 – Attach vEdge4 to the Device Template**

- In vManage, Navigate to Configuration -> **Templates -> Device -> BR2-VE-vEdge4-TEMP**
- Click on “...” towards the right-hand side.
- Click **Attach Devices**.
- Select **vEdge4** and click the “->” button.
- Click **Attach**.

### **Task 3 – Configure the Variable Parameters for the Feature Templates**

- **vEdge4** will appear in the window.
- Click on “...” towards the right-hand side.
- Click **Edit Device Template**.
- Configure the variables based on the following:
  - Default Gateway for VPN0 : **192.1.22.254**
  - Interface IP for ge0/0 : **192.168.22.4/24**
  - Interface IP for ge0/1 : **192.1.22.4/24**
  - Interface IP for ge0/2 : **172.16.20.4/24**
  - Interface IP for ge0/3 : **192.1.21.4/24**
  - TLOC Extension: **ge0/1**
  - Timezone: **Europe/London**
  - Hostname : **vEdge-4**
  - System IP : **10.2.2.204**
  - Site ID : **2**

- Click **Update**.
- Verify the Configuration & Click **Configure Devices**.
- Wait for it to update the device. It should come back with Status of **Success**.

## Site-2 Internal Router Configuration

### Site-2 Internal Router

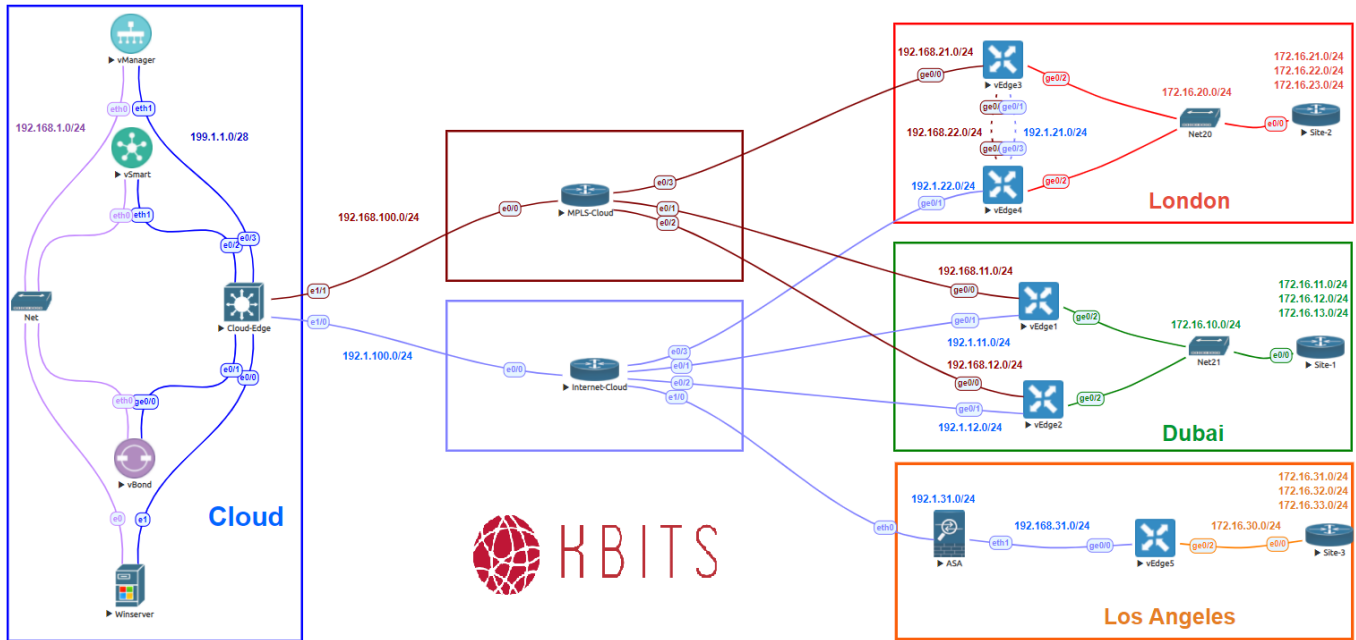
```
No ip domain-lookup
!  
Hostname R2
!  
Interface E 0/0
Ip address 172.16.20.22 255.255.255.0
No shut
!  
Interface loopback1
Ip address 172.16.21.1 255.255.255.0
Ip ospf network point-to-point
!  
Interface loopback2
Ip address 172.16.22.1 255.255.255.0
Ip ospf network point-to-point
!  
Interface loopback3
Ip address 172.16.23.1 255.255.255.0
Ip ospf network point-to-point
!  
Router ospf 1
Network 172.16.0.0 0.0.255.255 area 0
```

## Verification

- Verify the configuration on **vSmart**. You can do that by making sure that you are receiving 2 TLOCS for vEdge3 and 2 TLOCS for vEdge4. The command to verify is **show omp tlocs**.



# Lab 35 – Load Balancing using Multiple vEdges



## vEdge1 Templates Creation

### VPN 0

#### Task 1 – Configure a VPN Template to be used by BR2 vEdges for VPN0

- In vManage, Navigate to Configuration -> **Templates** -> **Feature** -> **vEdge Cloud** -> **VPN** -> **VPN**
- Configure the VPN parameters based on the following:

- Template Name : **BR1-VE-VPN-VPN0**
- Description : **BR1-VE-VPN-VPN0**

#### Basic Configuration

- VPN -> Global : **0**
- Name -> Global : **Transport VPN**

#### IPv4 Route

- Prefix -> Global : 0.0.0.0/0
- Next Hop -> Device Specific

➤ Click **Save** to save the Template.

### **Task 2 – Configure a VPN Interface Template to be used by all BR2 vEdge-Cloud Devices for VPN 0 for Interface G0/0**

➤ In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> VPN -> VPN Interface Ethernet**

➤ Configure the VPN parameters based on the following:

- Template Name : **BR1-VE-VPNINT-VPNO-GO**
- Description : **BR1-VE-VPNINT-VPNO-GO**

#### **Basic Configuration**

- Shutdown -> Global : **No**
- Interface Name -> Global : **ge0/0**
- IPv4 Address -> Static -> Device Specific

#### **Tunnel**

- Tunnel Interface -> Global : **On**
- Color -> Global : **Mpls**

#### **Allow Service**

- All -> Global : **On**
- NETCONF -> Global : **On**
- SSH -> Global : **On**

➤ Click **Save** to save the Template.

### **Task 3 – Configure a VPN Interface Template to be used by all BR2 vEdge-Cloud Devices for VPN 0 for Interface G0/1**

➤ In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> VPN -> VPN Interface Ethernet**

➤ Configure the VPN parameters based on the following:

- Template Name : **BR1-VE-VPNINT-VPNO-G1**
- Description : **BR1-VE-VPNINT-VPNO-G1**

#### **Basic Configuration**

- Shutdown -> Global : **No**
- Interface Name -> Global : **ge0/1**

- IPv4 Address -> Static -> Device Specific

### **Tunnel**

- Tunnel Interface -> Global : **On**
- Color -> Global : **Biz-Internet**
- Allow Service**
- All -> Global : **On**
- NETCONF -> Global : **On**
- SSH -> Global : **On**

- Click **Save** to save the Template.

## **Task 4 – Configure a OSPF Template to be used by all BR2 vEdge-Cloud Devices for VPN 0**

- In vManage, Navigate to Configuration -> **Templates -> Feature -> vEdge Cloud -> Other Templates -> OSPF**
- Configure the OSPF parameters based on the following:
  - Template Name : **BR1-VE-OSPF-VPN0**
  - Description : **BR1-VE-OSPF-VPN0**

### **Area Configuration**

- Area Number -> Global : **0**
- Area Type -> Default
- Interface Configuration**
- Interface Name: **ge0/0**

- Click **Save** to save the Template.

## **vEdge 1&2 Templates Deployment**

### **Task 1 – Configure a Device Template for BR2 vEdges.**

- In vManage, Navigate to Configuration -> **Templates -> Device -> Create Template -> vEdge Cloud**
- Configure the Device Template based on the following:
  - Template Name : **BR1-VE-TEMP**
  - Description : **BR1-VE-TEMP**

### **Basic Information**

- System -> **VE-System**

### Transport & Management

- VPN 0 : **BR1-VE-VPN-VPN0**
- VPN Interface : **BR1-VE-VPNINT-VPN0-G0**
- VPN Interface : **BR1-VE-VPNINT-VPN0-G1**
- OSPF: **BR1-VE-OSPF-VPN0**
  
- VPN 512 : **BR-VE-VPN-VPN512**
- VPN Interface : **BR-VE-VPNINT-VPN512-E0**

### Service VPN

- VPN 1 : **BR-VE-VPN-VPN1**
- VPN Interface : **BR-VE-VPNINT-VPN1-G2**
- OSPF: **BR-VE-OSPF-VPN1**

- Click **Save** to save the Template.

### Task 2 – Attach vEdge1 & 2 to the Device Template

- In vManage, Navigate to Configuration -> **Templates** -> **Device** -> **BR2-VE-TEMP**
- Click on “...” towards the right-hand side.
- Click **Attach Devices**.
- Select **vEdge1 & vEdge2** and click the “->” button.
- Click **Attach**.

### Task 3 – Configure the Variable Parameters for the Feature Templates

- **vEdge1 & vEdge2** will appear in the window.
- Click on “...” towards the right-hand side.
- Click **Edit Device Template**.
- Configure the variables based on the following:

### vEdge1

- Default Gateway for VPN0 : **192.1.11.254**
- Interface IP for ge0/0 : **192.168.11.1/24**

- Interface IP for ge0/1 :**192.1.11.1/24**
- Interface IP for ge0/2 :**172.16.10.1/24**
- Timezone: **Asia/Dubai**
- Hostname : **vEdge-1**
- System IP : **10.2.2.201**
- Site ID : **1**

## vEdge2

- Default Gateway for VPN0 : **192.1.12.254**
- Interface IP for ge0/0 :**192.168.12.2/24**
- Interface IP for ge0/1 :**192.1.12.2/24**
- Interface IP for ge0/2 :**172.16.10.2/24**
- Timezone: **Asia/Dubai**
- Hostname : **vEdge-2**
- System IP : **10.2.2.202**
- Site ID : **1**

- Click **Update**.
- Verify the Configuration & Click **Configure Devices**.
- Wait for it to update the device. It should come back with Status of **Success**.

## Site-1 Internal Router Configuration

### Site-1 Internal Router

```

No ip domain-lookup
!
Hostname R1
!
Interface E 0/0
Ip address 172.16.10.11 255.255.255.0
No shut
!
Interface loopback1
Ip address 172.16.11.1 255.255.255.0
Ip ospf network point-to-point
!
Interface loopback2
Ip address 172.16.12.1 255.255.255.0
Ip ospf network point-to-point

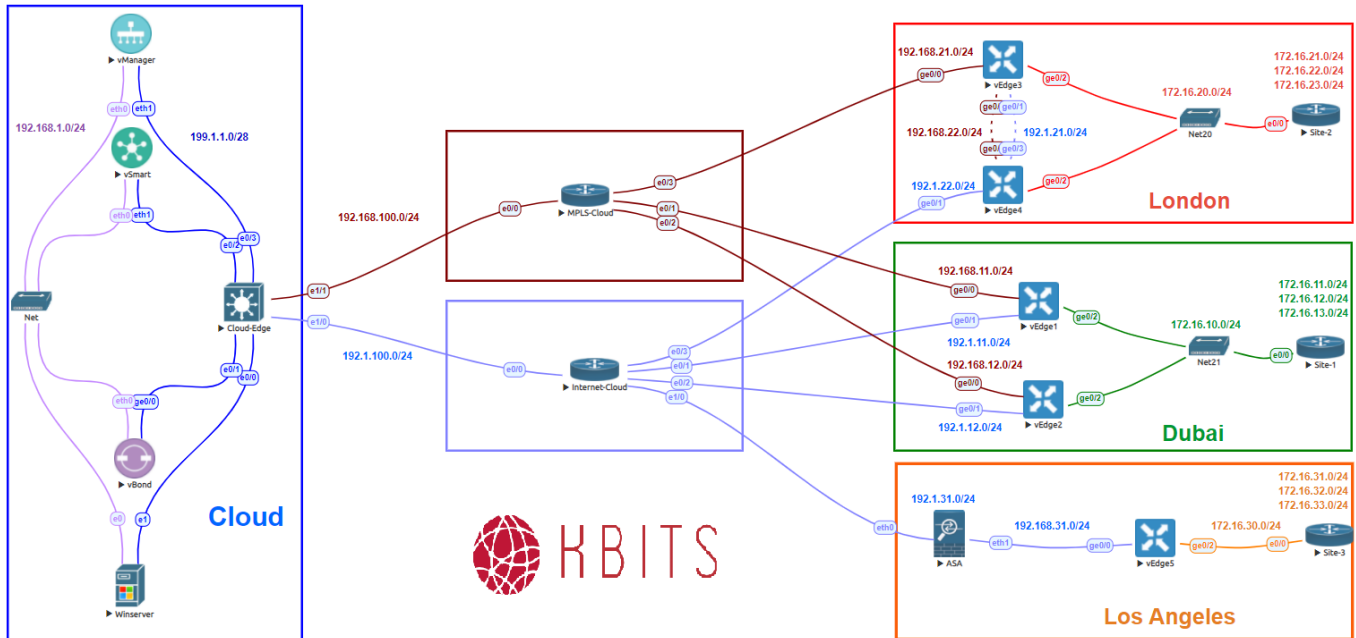
```

```
!  
Interface loopback3  
Ip address 172.16.13.1 255.255.255.0  
Ip ospf network point-to-point  
!  
Router ospf 1  
Network 172.16.0.0 0.0.255.255 area 0
```

## Verification

- Verify the configuration on **vEdge1 & vEdge2**. You can do that by verify OSPF Neighbor relationship with the MPLS Router by issuing the **Show ospf neighbor** command on **vEdge1 & vEdge2**.

# Lab 36 – Configuring A Hub-n-Spoke Topology using a TLOC



## Task 1 – Configure a VPN Template to be used by vSmart Controllers for VPN 0

➤ In vManage, Navigate to Configuration -> **Templates** -> **Feature** -> **vSmart** -> **VPN** -> **VPN**

➤ Configure the VPN parameters based on the following:

- Template Name : **vSmart-VPN-VPNO**
- Description : **vSmart-VPN-VPNO**

### Basic Configuration

- VPN -> Global : **0**
- Name -> Global : **Transport VPN**

### IPv4 Route

- Prefix -> Global : **0.0.0.0/0**
- Next Hop -> Global : **199.1.1.14**

➤ Click **Save** to save the Template.

## Task 2 – Configure a VPN Template to be used by vSmart Controllers for VPN 512

- In vManage, Navigate to Configuration -> **Templates -> Feature -> vSmart -> VPN -> VPN**
- Configure the VPN parameters based on the following:
  - Template Name : **vSmart-VPN-VPN512**
  - Description : **vSmart-VPN-VPN512**
  - Basic Configuration**
    - VPN -> Global : **512**
    - Name -> Global : **MGMT VPN**
- Click **Save** to save the Template.

## Task 3 – Configure a VPN Interface Template to be used by vSmart Controllers for VPN 0 for Interface Eth1

- In vManage, Navigate to Configuration -> **Templates -> Feature -> vSmart -> VPN -> VPN Interface Ethernet**
- Configure the VPN parameters based on the following:
  - Template Name : **vSmart-VPNINT-VPN0-E1**
  - Description : **vSmart-VPNINT-VPN0-E1**
  - Basic Configuration**
    - Shutdown -> Global : **No**
    - Interface Name -> Global : **eth1**
    - IPv4 Address -> Static -> Device Specific
  - Tunnel**
    - Tunnel Interface -> Global : **On**
    - Color -> default
  - Allow Service**
    - All -> Global : **On**
    - NETCONF -> Global : **On**
    - SSH -> Global : **On**
- Click **Save** to save the Template.



#### **Task 4 – Configure a VPN Interface Template to be used vSmart Controllers for VPN 512 for Interface Eth0**

- In vManage, Navigate to Configuration -> **Templates -> Feature -> vSmart -> VPN -> VPN Interface Ethernet**
- Configure the VPN parameters based on the following:
  - Template Name : **vSmart-VPNINT-VPN512-E0**
  - Description : **vSmart-VPNINT-VPN512-E0**
  - Basic Configuration**
    - Shutdown -> Global : **No**
    - Interface Name -> Global : **eth0**
    - IPv4 Address -> Static -> Device-Specific
- Click **Save** to save the Template

#### **Task 5 – Configure a Device Template for vSmart Controllers.**

- In vManage, Navigate to Configuration -> **Templates -> Device -> Create Template -> vSmart**
- Configure the Device Template based on the following:
  - Template Name : **vSmart-TEMP**
  - Description : **vSmart-TEMP**
  - Basic Information**
    - System -> **VE-System**
  - Transport & Management**
    - VPN 0 : **vSmart-VPN-VPNO**
    - VPN Interface : **vSmart-VPNINT-VPNO-E1**
  
    - VPN 512 : **vSmart-VPN-VPN512**
    - VPN Interface : **vSmart-VPNINT-VPN512-E0**
- Click **Save** to save the Template.

#### **Task 6 – Attach vSmart to the Device Template**

- In vManage, Navigate to Configuration -> **Templates -> Device -> vSmart-TEMP**

- Click on “...” towards the right-hand side.
- Click **Attach Devices**.
- Select **vSmart** and click the “ -> “ button.
- Click **Attach**.

### **Task 7 – Configure the Variable Parameters for the Feature Templates**

- **vSmart** will appear in the window.
- Click on “...” towards the right-hand side.
- Click **Edit Device Template**.
- Configure the variables based on the following:
  - Interface IP for Eth1 : **199.1.1.2/28**
  - Interface IP for Eth1 : **192.168.1.2/24**
  - Hostname : **vSmart-1**
  - System IP : **10.1.1.102**
  - Site ID : **1**
- Click **Update**.
- Verify the Configuration & Click **Configure Devices**.
- Wait for it to update the device. It should come back with Status of **Success**.

## Policy Requirements:

- Los Angeles & London Sites are communicating to each other directly. You can verify this by checking the routes. The routes should be pointing directly at the TLOCs of the Branch Sites directly.
- All traffic between the sites should be forwarded via the HQ Site Dubai. Use a TLOC list to accomplish this task.

### Task 1 – Configure Groups of Interests/List that will be used for Hub-n-Spoke

- In vManage, Navigate to **Configuration -> Policies -> Custom Options -> Centralized Policy -> Lists.**
- Click **VPN** and select **New VPN list.** Create 1 policy based on the following:
  - Name : **VPN1**
  - ID : **1**
- Click **Site** and select **New Site list.** Create 2 policies based on the following:
  - Name : **Dubai**
  - Site ID : **1**
  
  - Name : **London**
  - Site ID : **2**
  
  - Name : **Los Angeles**
  - Site ID : **3**
- Click **TLOC** and select **New TLOC list.** Create 1 policies based on the following:
  - Name : **TLOC-Dubai**
  - **TLOCs**
    - 10.2.2.201 – mpls – IPsec – 500
    - 10.2.2.202 – mpls – IPsec – 500
    - 10.2.2.201 – biz-internet – IPsec – 400
    - 10.2.2.202 – biz-internet – IPsec – 400

## Task 2 – Configure a Topology based on the Requirements

- In vManage, Navigate to **Configuration -> Policies -> Custom Options -> Centralized Policy -> Topology -> Add Topology -> Custom ->**

- Configure the topology based on the following:

- Policy Name : **Hub-n-Spoke**
- Description : **Hub-n-Spoke**

### Route Sequence- London

#### Match Conditions:

- Site: London
- **Action**
- TLOC: **TLOC-List = Dubai-TLOC**
- Click **Save Match and Actions** to save the Sequence.

### Route Sequence- Los Angeles

#### Match Conditions:

- Site: Los Angeles
- **Action**
- TLOC: **TLOC-List = Dubai-TLOC**
- Click **Save Match and Actions** to save the Sequence.

### Default

#### Action

- Accept
- Click **Save Match and Actions**
- Click **Save Match and Actions** to save the Sequence.
- **Save Control Policy.**

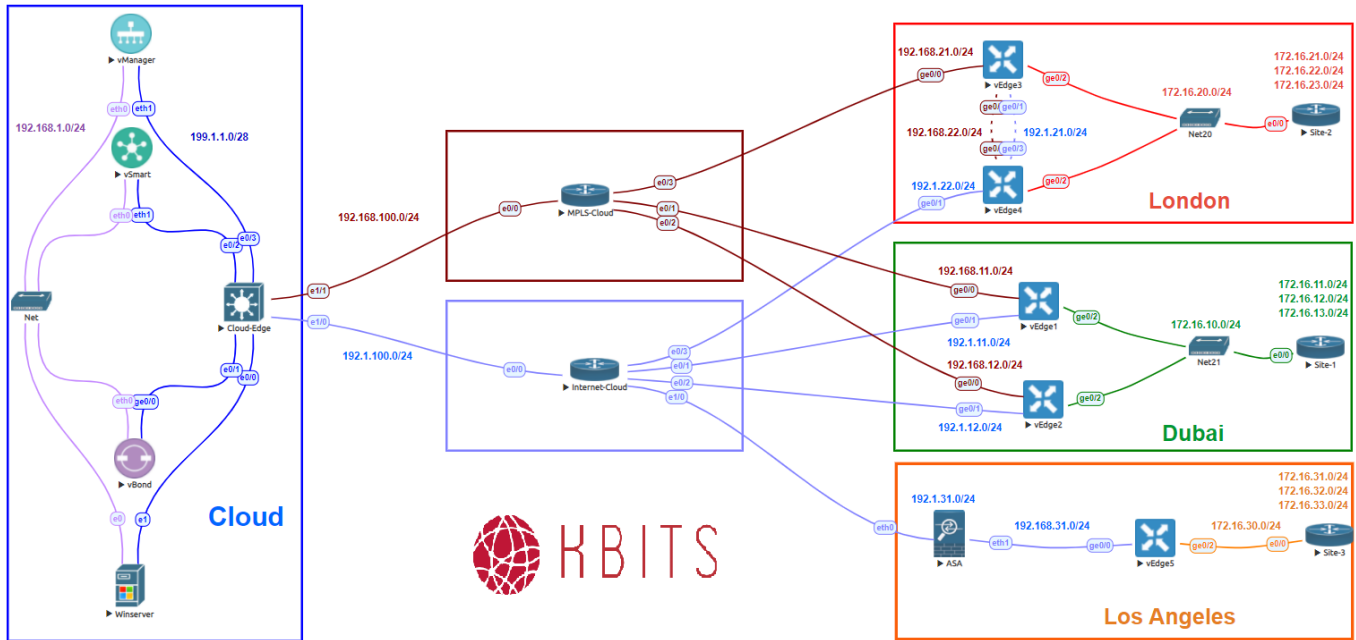
## Task 3 – Create a Centralized Policy and call the Traffic Policy

- In vManage, Navigate to **Configuration -> Policies -> Custom Options -> Centralized Policy -> Add Centralized Policy**

- Click **Next** on the “**Group of Interests**” page as we have already created the required lists.

- Click **Add Policy** on the “**Topology and VPN Membership**” page.
- Click “**Import Existing**”, Select **Custom** and select the **Hub-n-Spoke** from the drop-down list and click **Import**. Click **Next**.
- Click **Next** on the “**Configure Traffic Rules**” page as we are not using any Control Policies. You will move to the “**Apply Policy to Sites and VPNs**” Page.
- The **Hub-n-Spoke** policy will be there. Click “**New Site**” button.
- Select **Los Angeles** and **London** in the Outbound Site List.
- Click **Add**.
- Assign the Policy a name and Description based on the following:
  - Policy Name : **Main-Central-Policy**
  - Description : **Main-Central-Policy**
- Click the **Save Policy** button towards the button.
- Activate the policy.
- Wait for it to push the policy to the reachable vSmart Controller(s).
- You can verify this by doing checking the routes. The routes should be pointing directly at the TLOCs of Dubai and all traffic will be forwarded thru Dubai.

# Lab 37 - Implementing QoS - Configuring Custom Options



## Requirements:

- Create 3 Class maps and assign them to queues based on the following:
  - Name: **CM-Priority** – Queue: 0
  - Name: **CM-Web** – Queue: 1
  - Name: **CM-Best-Effort** – Queue: 2

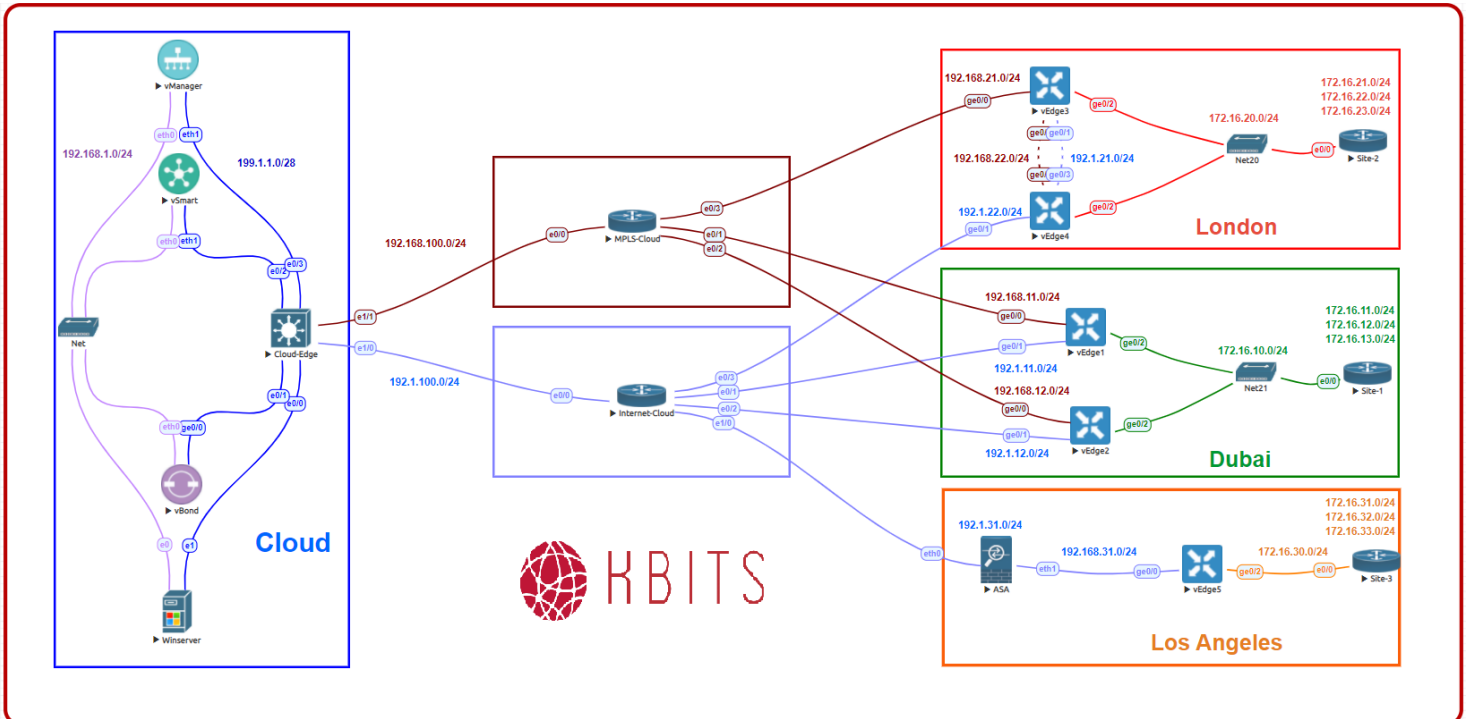
## Task 1 – Create the Class-maps and apply it to a Queue

- In vManage, Navigate to **Configuration -> Policies -> Custom Options -> Localized Policy -> Lists**
- Click **Class Map** and select **New Class Map**. Create 3 Class-Maps based on the following:
  - Name : **CM-Priority**
  - Queue : **0**
  
  - Name : **CM-Web**
  - Site ID : **1**
  
  - Name : **CM-Best-Effort**
  - Site ID : **2**

## Task 2 – Create a Classification ACL to link the Traffic to the appropriate Class Maps

- In vManage, Navigate to **Configuration -> Policies -> Custom Options -> Localized Policy -> Lists**
- Click **Access Lists** and select **Add IPv4 ACL**. Create an ACL with a name of QOS-ACL with the following “Traffic to Class Map” Mappings
  - Traffic: **DSCP – 46**
  - Class-Map: **CM-Priority**
  
  - Traffic: **TCP/22 & 23**
  - Class-Map: **CM-Priority**
  
  - Traffic: **TCP/80 & 443**
  - Class-Map: **CM-Web**
  
  - Traffic: **Rest**
  - Class-Map: **CM-Best-Effort**

# Lab 38 - Implementing QoS - Configuring the Scheduler



## Requirements:

- Configure the Queue characteristics in the Forwarding Class based on the following:
  - **Queue 1**
    - Bandwidth reservation - **30%**
    - Scheduling - **wrr**
    - Drop - **Random Early Detection**
  - **Queue 2**
    - Bandwidth reservation - **30%**
    - Scheduling - **wrr**
    - Drop - **Tail**

## Task 1 - Create the Forwarding Class based on the above requirements

- In vManage, Navigate to **Configuration -> Policies -> Custom Options -> Localized Policy -> Forwarding Class/Qos**



➤ Click **Add Qos Map** and select **Create New**. Create a new Qos Map called **QOS-MAP** based on the following:

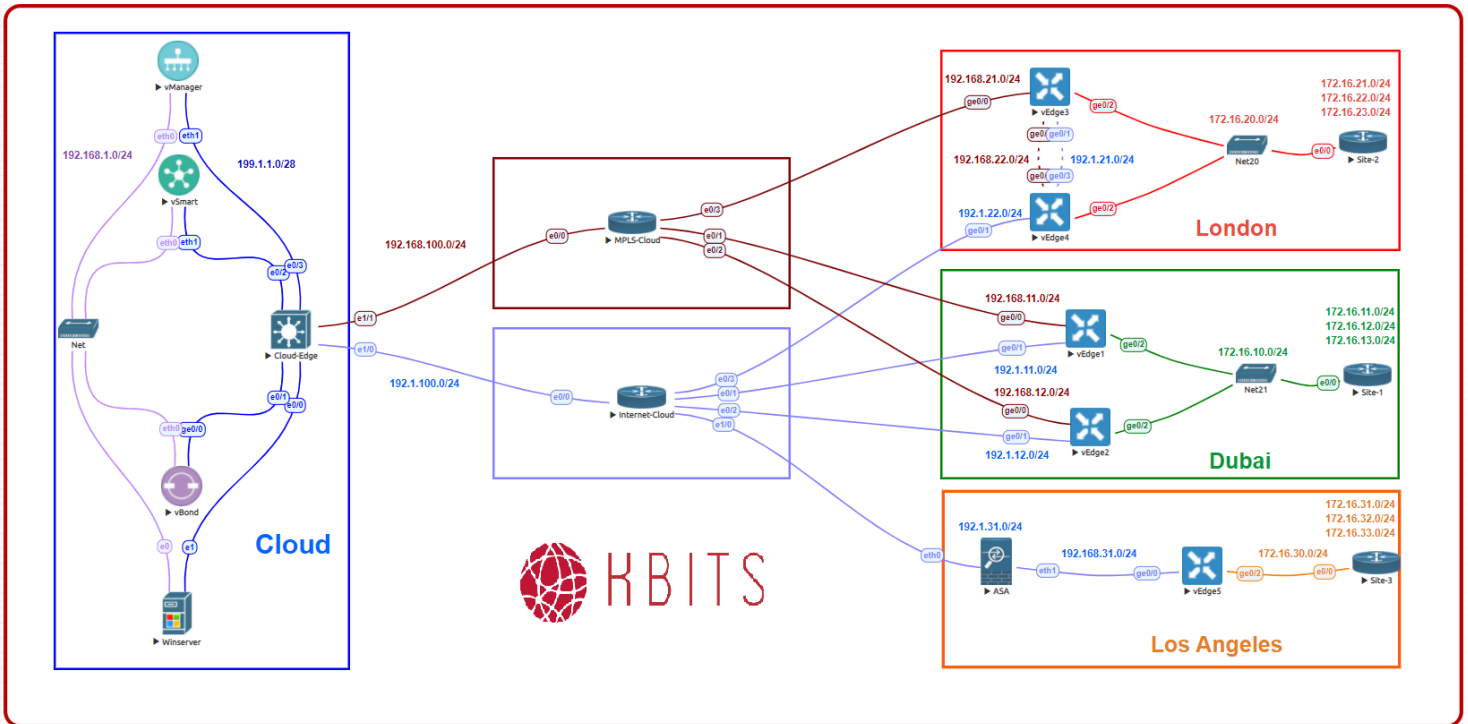
○ **Queue 1**

- Bandwidth reservation - **30%**
- Scheduling - **wrr**
- Drop - **Random Early Detection**

○ **Queue 2**

- Bandwidth reservation - **30%**
- Scheduling - **wrr**
- Drop - **Tail**

# Lab 39 - Implementing QoS - Configure & apply the Localized Policies



## Task 1 – Create a new Local Policy

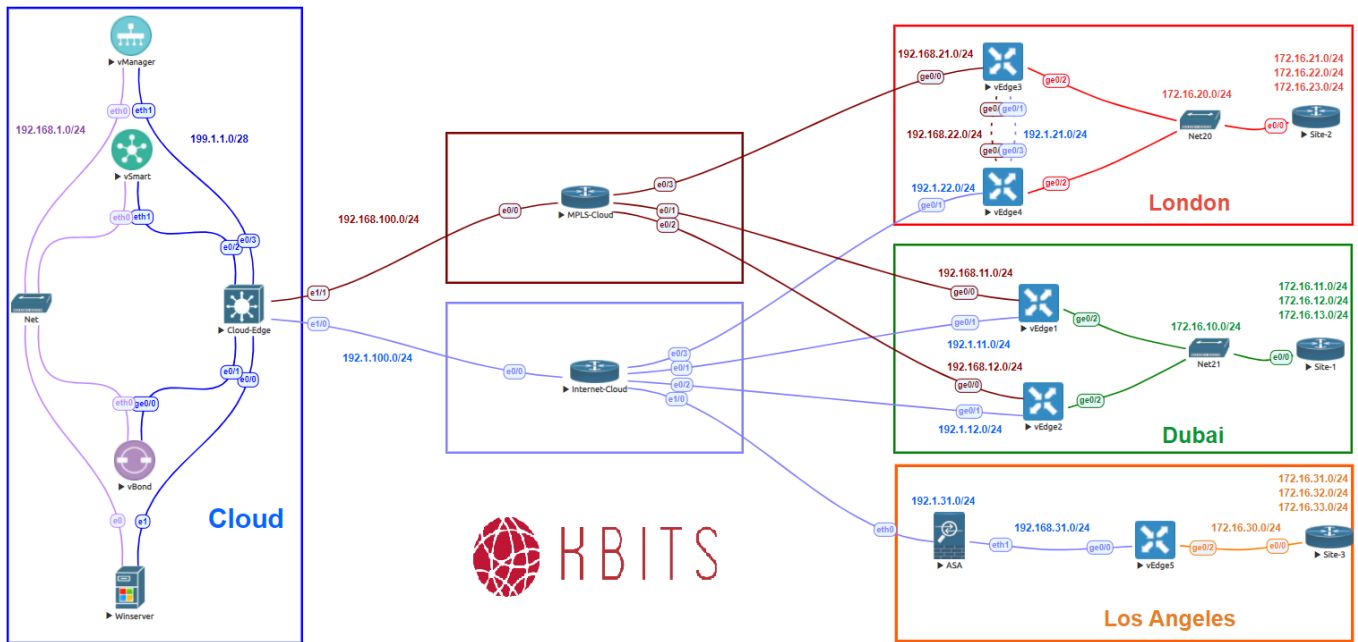
- In vManage, Navigate to **Configuration -> Policies -> Localized Policy**
- Click **Next** on the “**Group of Interests**” page as we have already created the required lists.
- Click **Import Qos Map** on the “**Forwarding Class**” page.
- Click “**Import QoS Map**”. Select **QOS-MAP** from the drop-down list and click **Import**. Click **Next**.
- Click **Add IPv4 Access List** on the “**ACL Page**” page.
- Select **QOS-ACL** from the drop-down list and click **Import**. Click **Next**.
- The **Next** on the “**Route Policy**” page.

- Assign the Policy a name and Description based on the following:
  - Policy Name : **QOS-POLICY**
  - Description : **QOS-POLICY**
- Click the **Save Policy** button towards the button.

### **Task 2 – Apply the new policy to the Dubai Device Template**

- In vManage, Navigate to **Configuration -> Templates -> Device Templates -> BR1-DEV-TEMP**
- Click **Edit**
- Configure the Localized Policy as **QOS-POLICY**
- Follow the prompts to update the 2 vEdges in BR1 (Dubai)

# Lab 40 - Implementing QoS - Configure the Interface parameters using Templates



## Task 1 – Apply the QOS-ACL to the “BR-VE-VPNINT-VPN1-G2” template

- In vManage, Navigate to **Configuration -> Templates -> Feature Template -> VE-VPNINT-G2 -> Edit -> ACL/QoS**
- Enable the “**Ingress ACL – Ipv4**” globally.
- Specify the IPv4 Ingress Access List as **QOS-ACL**.

## Task 2 – Apply the QOS-MAP & Shaper to the “BR1-VPNINT-G0” template

- In vManage, Navigate to **Configuration -> Templates -> Feature Template -> BR1-VE-VPNINT-VPN0-G0 -> Edit -> ACL/QoS**
- Specify the Shaping Rate (Kbps) as **500000**
- Specify the QoS MAP as **QOS-MAP**

### **Task 3 – Apply the QOS-MAP & Shaper to the “BR1-VPNINT-G1” template**

- In vManage, Navigate to **Configuration -> Templates -> Feature Template -> BR1-VE-VPNINT-VPN0-G1 -> Edit -> ACL/QoS**
- Specify the Shaping Rate (Kbps) as **200000**
- Specify the QoS MAP as **QOS-MAP**

# Implementing SDA

**Authored By:**

**Khawar Butt**

CCIE # 12353

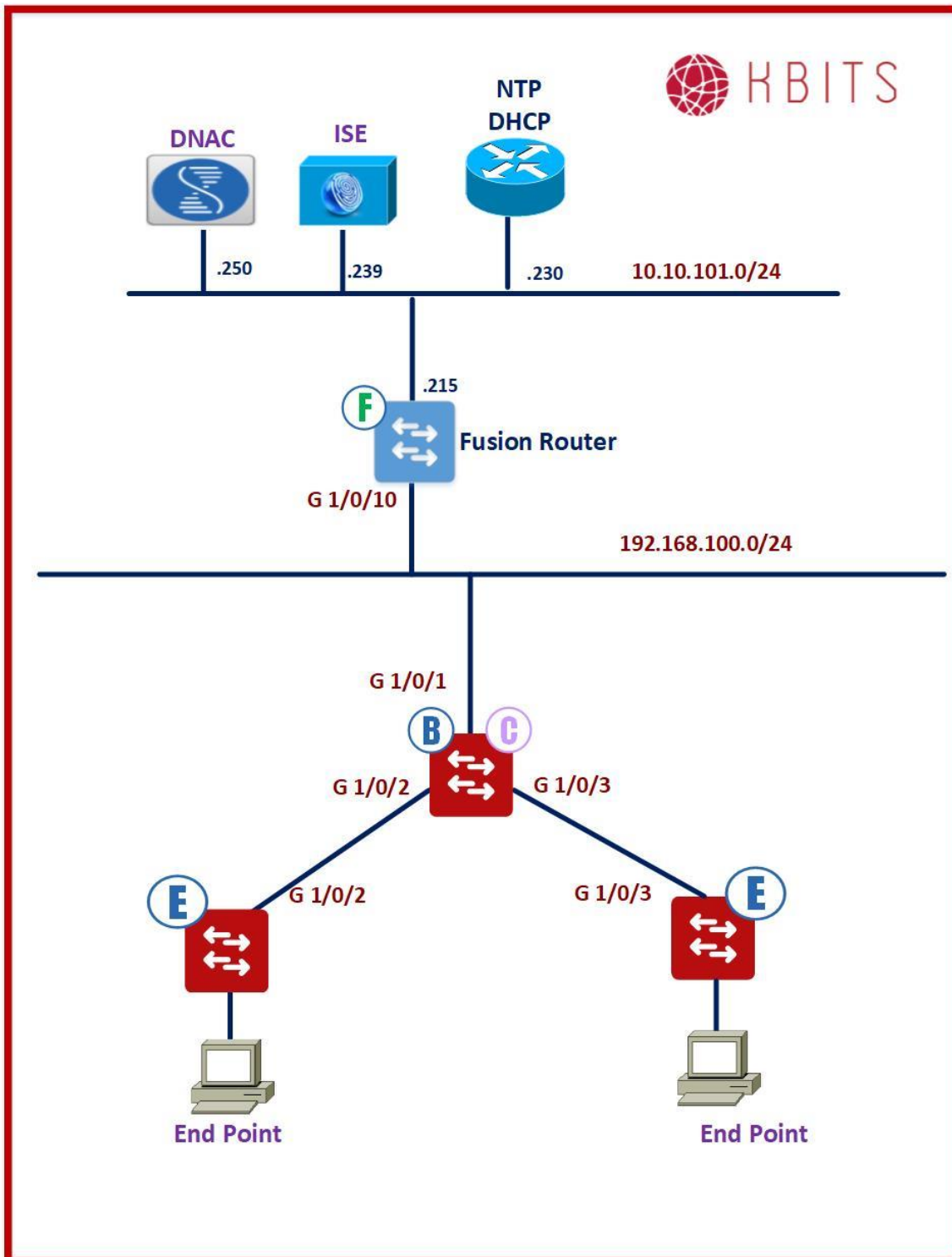
Hepta CCIE#12353

CCDE # 20110020

**Implementing SDA**



# Lab 1 – Configuring DNAC & ISE Integration



## **Task 1 – Turn on the Service on ISE**

### **RADIUS**

#### **Administration -> System -> Settings -> Protocols -> RADIUS**

- Uncheck “Reject RADIUS requests from clients with repeated failures”
- Uncheck “Suppress repeated failed clients”.
- Uncheck “Suppress repeated successful authentications”
- Click Save

### **ERS**

#### **Administration -> System -> Settings -> ERS Settings**

- Check Enable ERS for Read/Write
- Click Save

### **pxGrid**

#### **Administration -> System -> Deployment -> DNAC-ISE**

- Check to enable the following:
  - ➔ **pxGrid**
- **Save**



## **2. Configure DNAC to communicate to ISE**

**Settings (Icon) -> System Settings -> Settings -> Authentication & Policy Servers -> Add**

- Server IP Address: 10.10.101.239
- Shared Secret: Cisco@123
- Cisco ISE Server: Slide to Enable
- Username: admin
- Password: Cisco@123
- FQDN: dnac-ise.kbits.local
- Subscriber Name: DNAC-KBITS

**Click Apply**

## **3. Verify and Approve the Integration on ISE**

**Administration -> pxGrid Services**

- Click on Total Pending and Approve All

**Note: Need to see "Connected via XMPP dna-ise.kbits.local"**

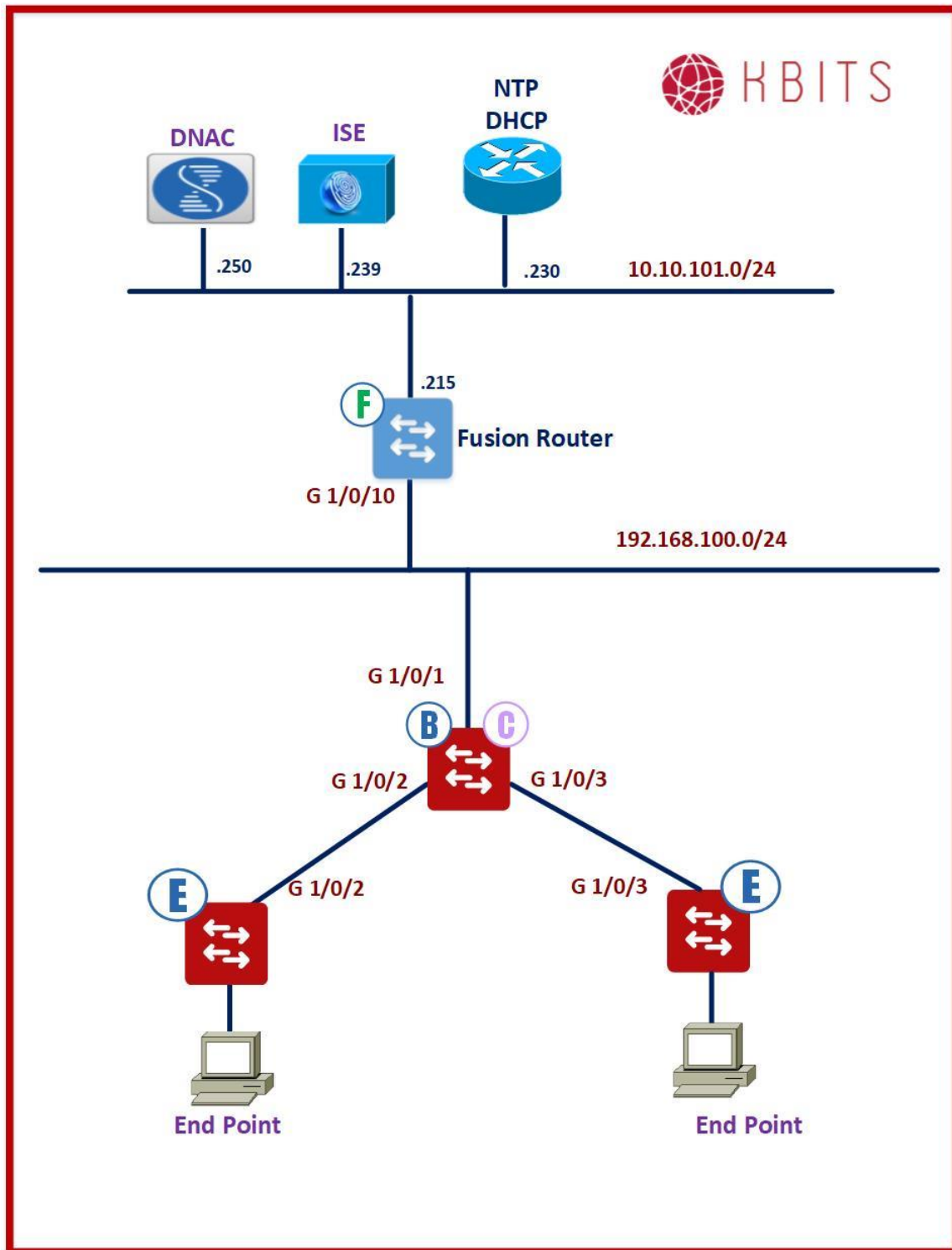
## **4. Migrate Policy Data from ISE into DNAC**

**Policy -> Group-Based Access Control -> Scalable Groups -> Start Migration (In Message)**

- **Yes to accept**

**Note: Wait for the Integration to complete**

# Lab 2 - Configuring Border Switch Initial Configuration



## Task 1 – Configure Connectivity towards Fusion Router

### 9300CB

```
no ip domain lookup
!  
line con 0
  logg sync
  no exec-timeout
!  
hostname 9300CB
!  
Interface Gig 1/0/1
  switchport mode trunk
!  
vlan 199
!  
ip routing
!  
interface VLAN 199
  ip address 192.168.100.2 255.255.255.0
  no shut
!  
ip route 0.0.0.0 0.0.0.0 192.168.100.1
```

## Task 2 – Configure Telnet/SSH Credentials

### 9300CB

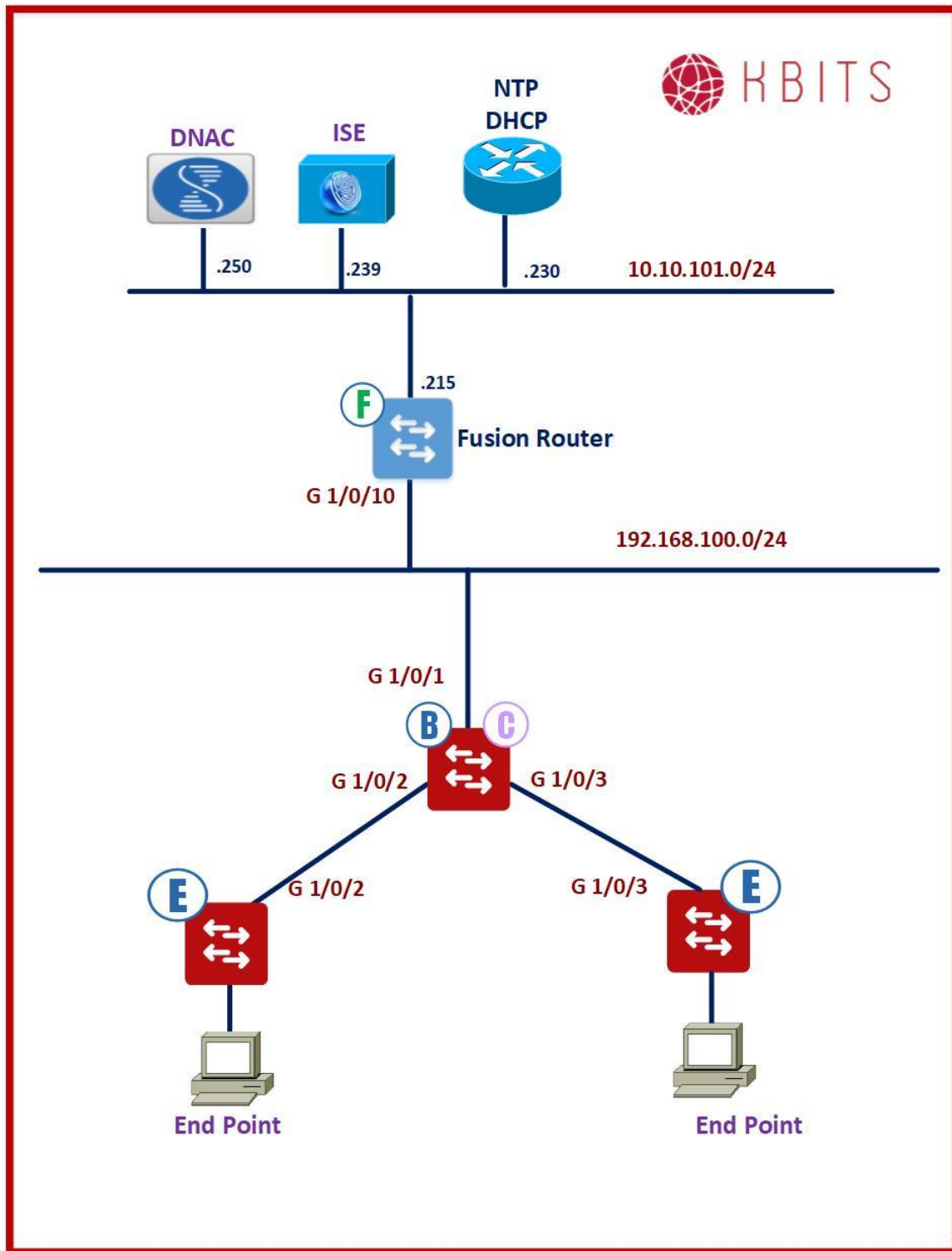
```
username kbits privilege 15 secret Cisco@123
!  
line vty 0 4
  login local
```

## Task 3 – Configure SNMP Parameters for RO & RW Communities

### 9300CB

```
snmp-server community RO ro public
snmp-server community RW rw private
```

# Lab 3 - Configuring Fusion Router Initial Configuration

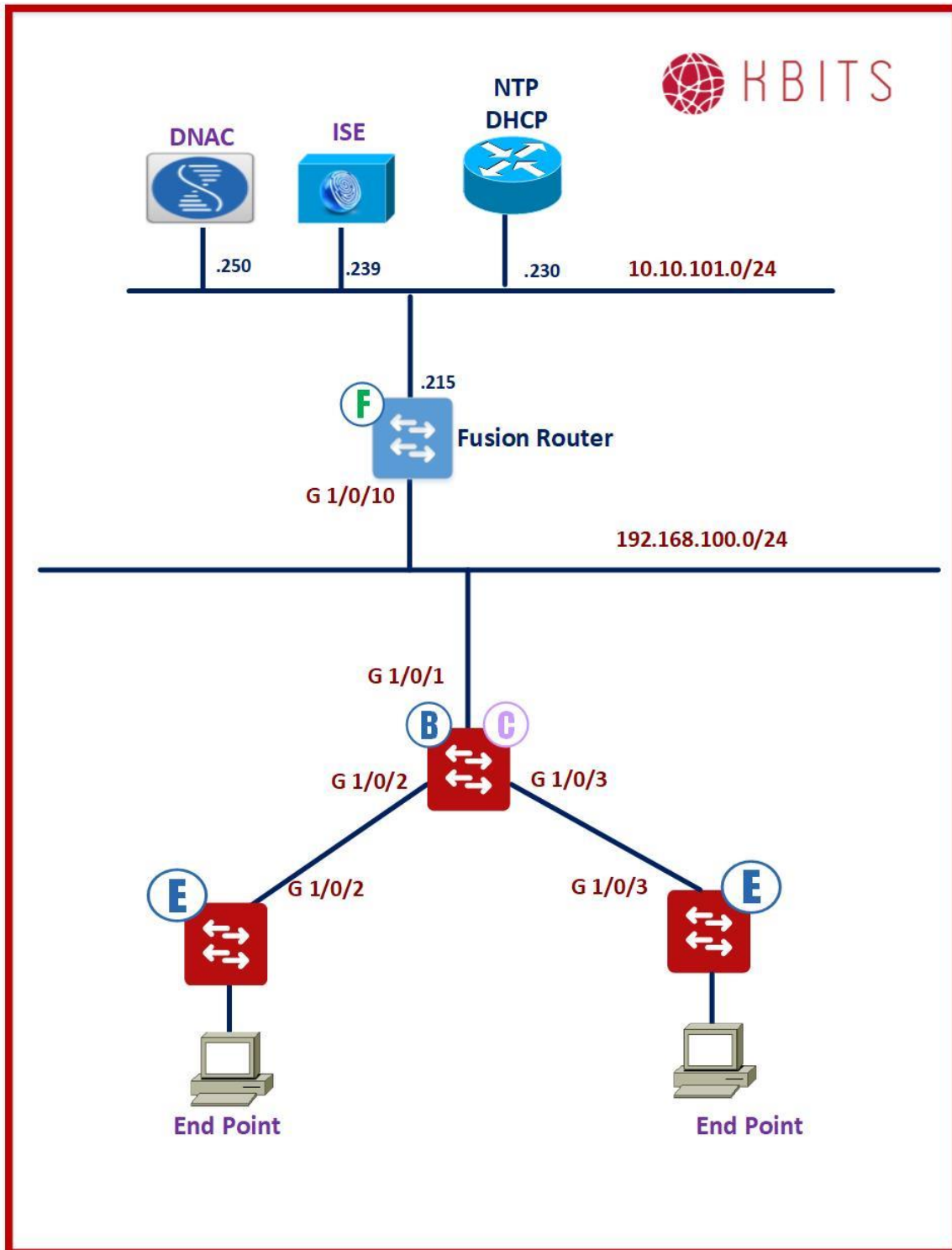


## **Task 1 – Configure Connectivity towards Fusion Router using VLAN 199**

### **Fusion Router**

```
hostname Fusion
!  
Interface Gig 1/0/10  
  switchport trunk encapsulation dot1q  
  switchport mode trunk  
!  
vlan 199  
!  
ip routing  
!  
interface VLAN 199  
  ip address 192.168.100.1 255.255.255.0  
  no shut
```

# Lab 4 – DNAC Design - Network Hierarchy – Site & Building



## Task 1 – Add an Area under Global

Desgin -> Network Hierarchy -> Add Site -> Add Area

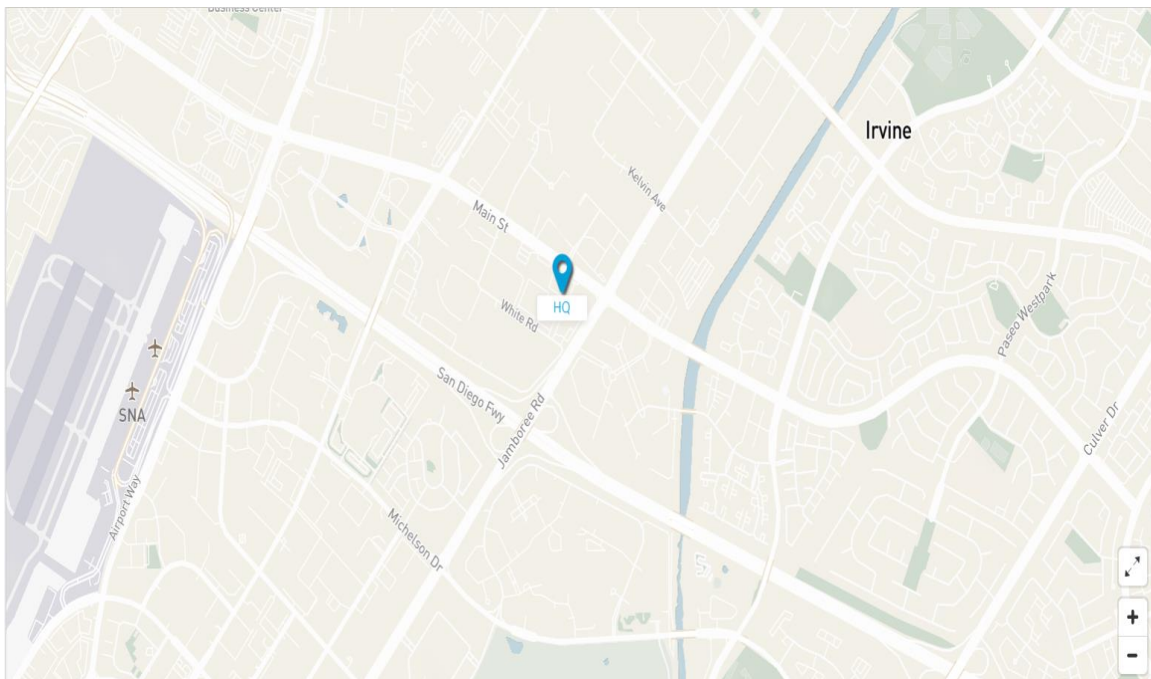
- Area Name: **Los Angeles**
- Parent: **Global**

## Task 2 – Add a Building under Los Angeles

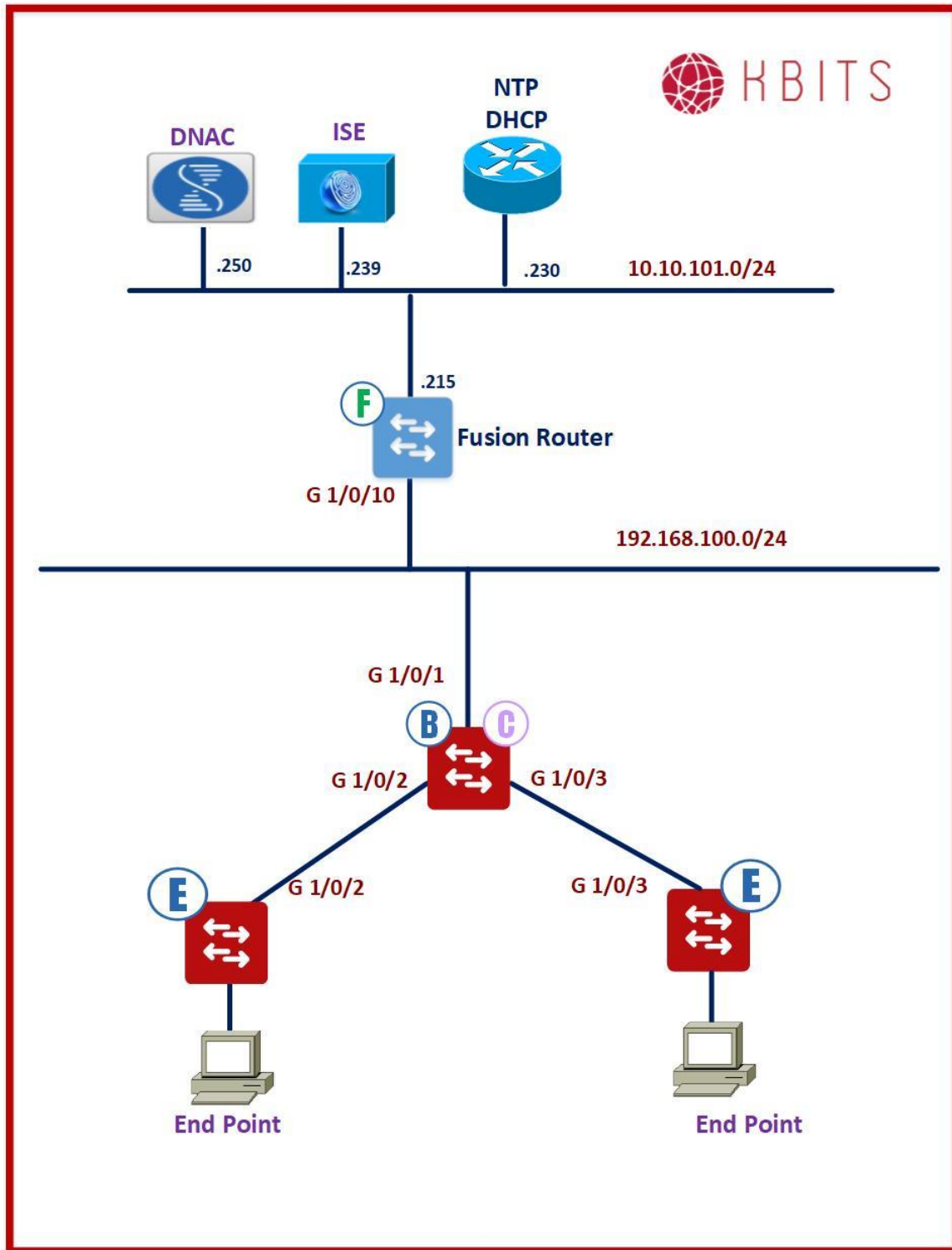
Desgin -> Network Hierarchy -> Add Site -> Add Building

- Building Name: **HQ**
- Parent: **Los Angeles**
- Address: **2640 Main Street, Irvine, California 92614, US**

→ **Click Save**



# Lab 5 – DNAC Design – Server Configuration – AAA, NTP etc





## **Task 1 – Add ISE & NTP Server to Network Settings**

**Desgin -> Network Settings -> Network -> Add**

- Click ISE
- Click NTP
- **Click OK**

## **Task 2 – Add ISE Parameters**

- Check Client/Endpoint

### **CLIENT/ENDPOINT**

- Servers: **ISE**
- Protocols: **RADIUS**
  
- Client/Endpoint: **10.10.101.239 (Select from Drop-Down)**
- IP Address (Pri.): **10.10.101.239 (Select from Drop-Down)**

## **3. Add DHCP Server**

- DHCP: **10.10.101.230**

## **4. Add NTP Server**

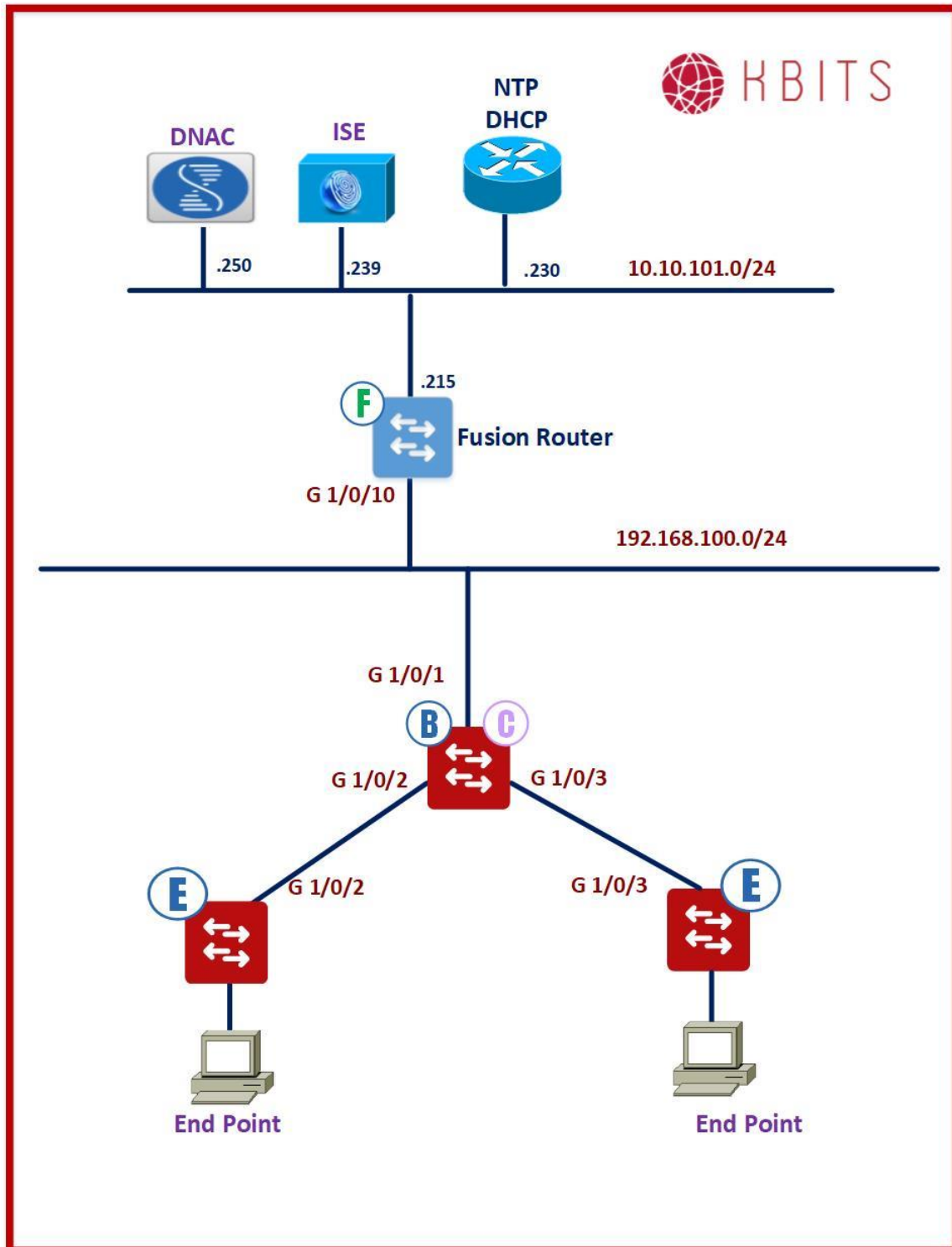
- NTP: **10.10.101.230**

## **5. Time Zone**

- Time Zone: **PST8PDT**

**→ Click Save**

# Lab 6 – DNAC Design - Device Credentials



## 1. Configuring CLI Credentials

Design -> Network Settings -> Device Credentials

**Note:** Click to make sure you are setting it at the Global Level

### CLI Credentials

- Name: **FabricAdmin**
- Username: **kbits**
- Password: **Cisco@123**
- Enable Password: **Cisco@123**

→ **Click Save**

## 2. Configuring SNMP Credentials

Select **SNMPV2C Read** -> **Click Add**

- Type: **SNMP v2c**
- Community Type: **Read**

- Name: **RO**
- Community: **public**

Select **SNMPV2C Write** -> **Click Add**

- Type: **SNMP v2c**
- Community Type: **Write**

- Name: **RW**
- Community: **private**

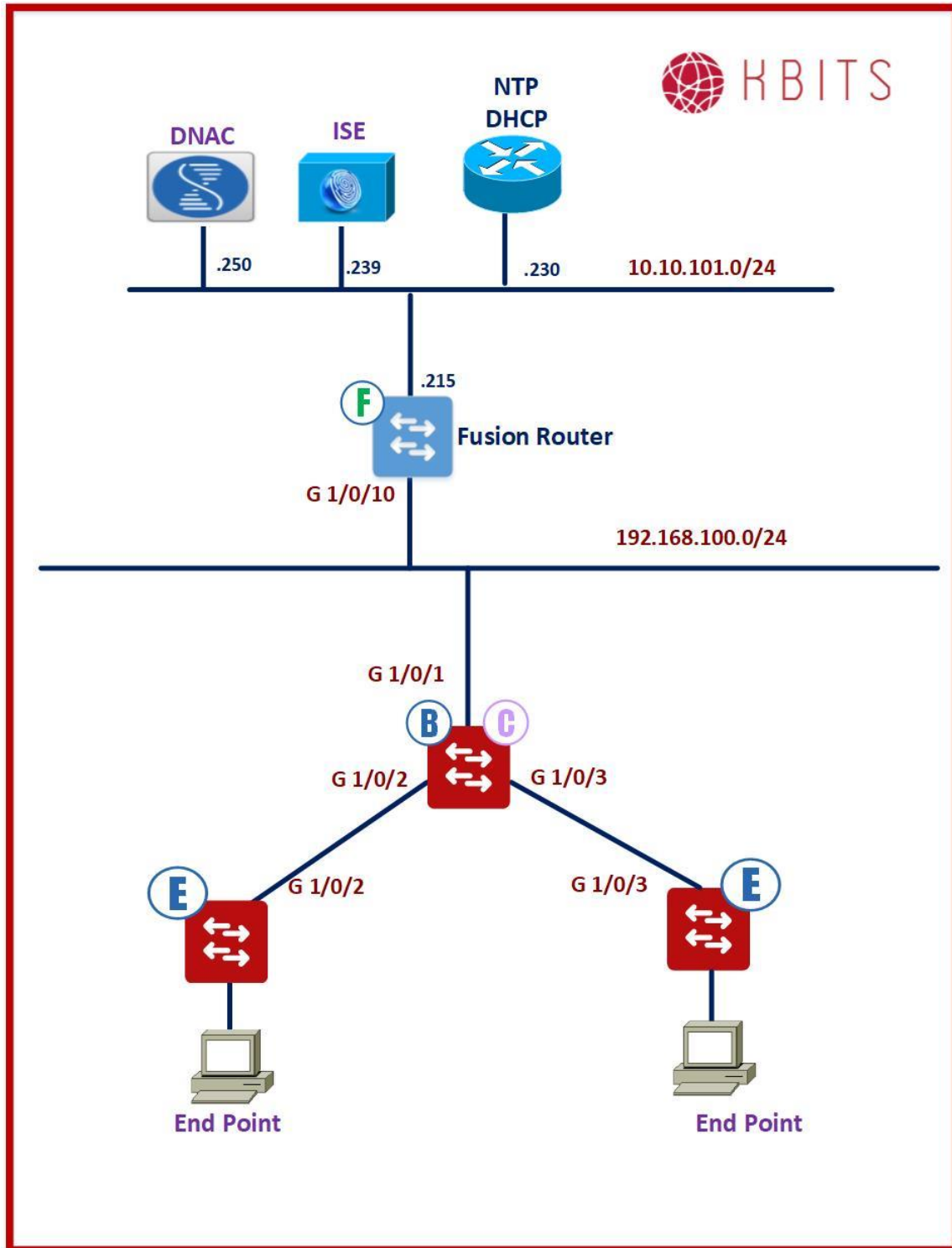
### **3. Delete existing CLI Admin**

→ **Click Save**

→ Delete the existing CLI Admin

→ **Click Save**

# Lab 7 – DNAC Design - IP Address Pools



## 1. Configuring Overlay Global Level Pool

Design -> Network Settings -> IP Address Pools

**Note:** Click to make sure you are setting it at the Global Level

→ Click on **Add**

→ Name: **LA\_OVERLAY\_POOL**

→ Type: **Generic**

→ IP Address Space: **(IPv4)**

→ Subnet: **172.16.0.0**

→ Prefix-length: **/16**

→ **Click Save**

→ Click on **Add**

→ Name: **LA\_UNDERLAY\_POOL**

→ Type: **Generic**

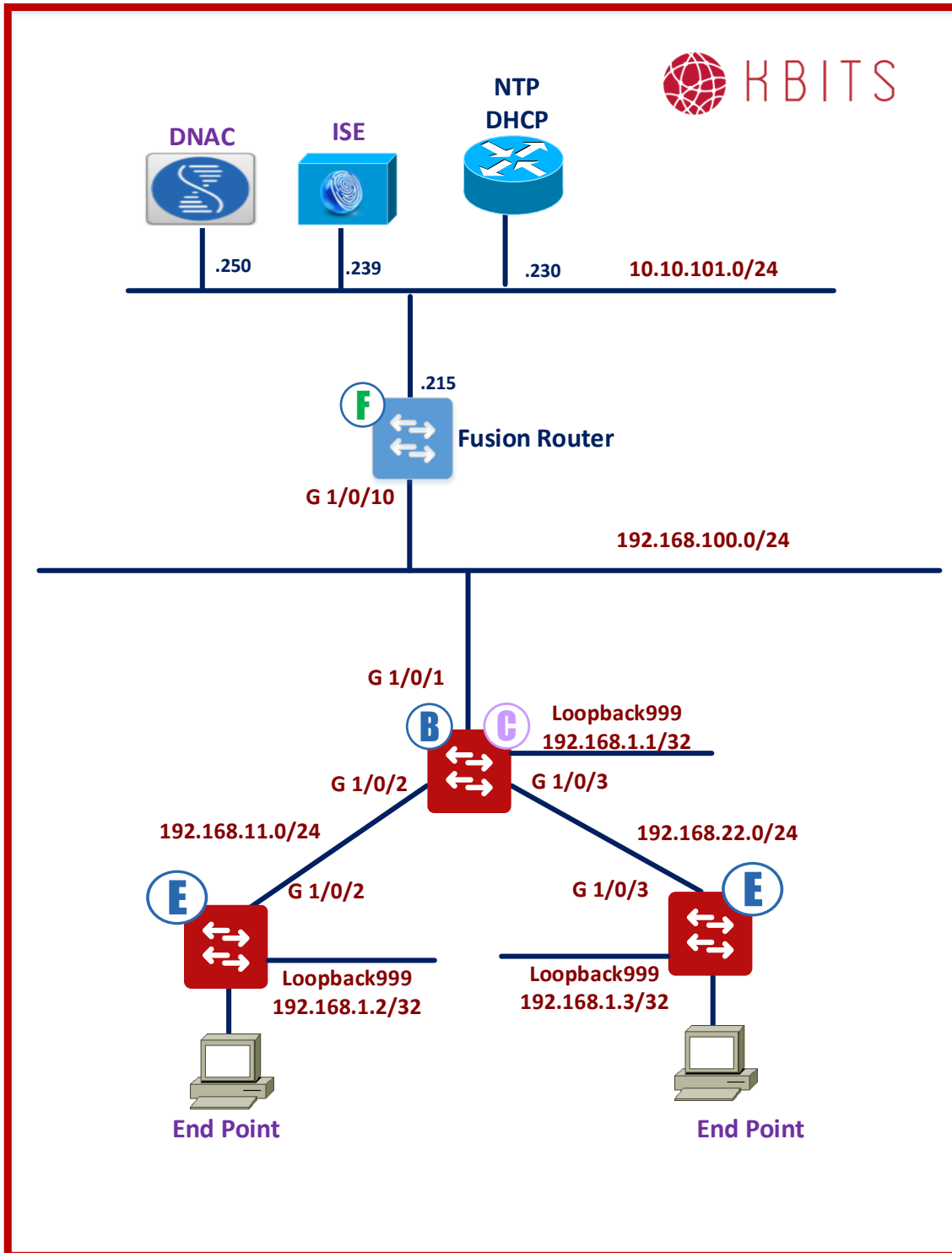
→ IP Address Space: **(IPv4)**

→ Subnet: **172.20.0.0**

→ Prefix-length: **/16**

→ **Click Save**

# Lab 8 – Manual Underlay Configuration – Fabric Skinny Configuration



## Task 1 – Configurig the Underlay Network for IP connectivity

### 9300CB

```
Interface Gig 1/0/2
no switchport
ip address 192.168.11.1 255.255.255.0
no shut
!
Interface Gig 1/0/3
no switchport
ip address 192.168.22.1 255.255.255.0
no shut
!
interface Loopback999
ip address 192.168.1.1 255.255.255.255
no shut
```

### 9300E1

```
Ip routing
!
Interface Gig 1/0/2
no switchport
ip address 192.168.11.2 255.255.255.0
no shut
!
interface Loopback999
ip address 192.168.1.2 255.255.255.255
no shut
```

### 9300E2

```
Ip routing
!
Interface Gig 1/0/3
no switchport
ip address 192.168.22.3 255.255.255.0
no shut
!
interface Loopback999
ip address 192.168.1.3 255.255.255.255
no shut
```



## Task 2 – Configure Telnet/SSH Credentials

### 9300E1

```
username kbits privilege 15 secret Cisco@123
!  
line vty 0 4  
login local
```

### 9300E2

```
username kbits privilege 15 secret Cisco@123
!  
line vty 0 4  
login local
```

## Task 3 – Configure SNMP Parameters for RO & RW Communities

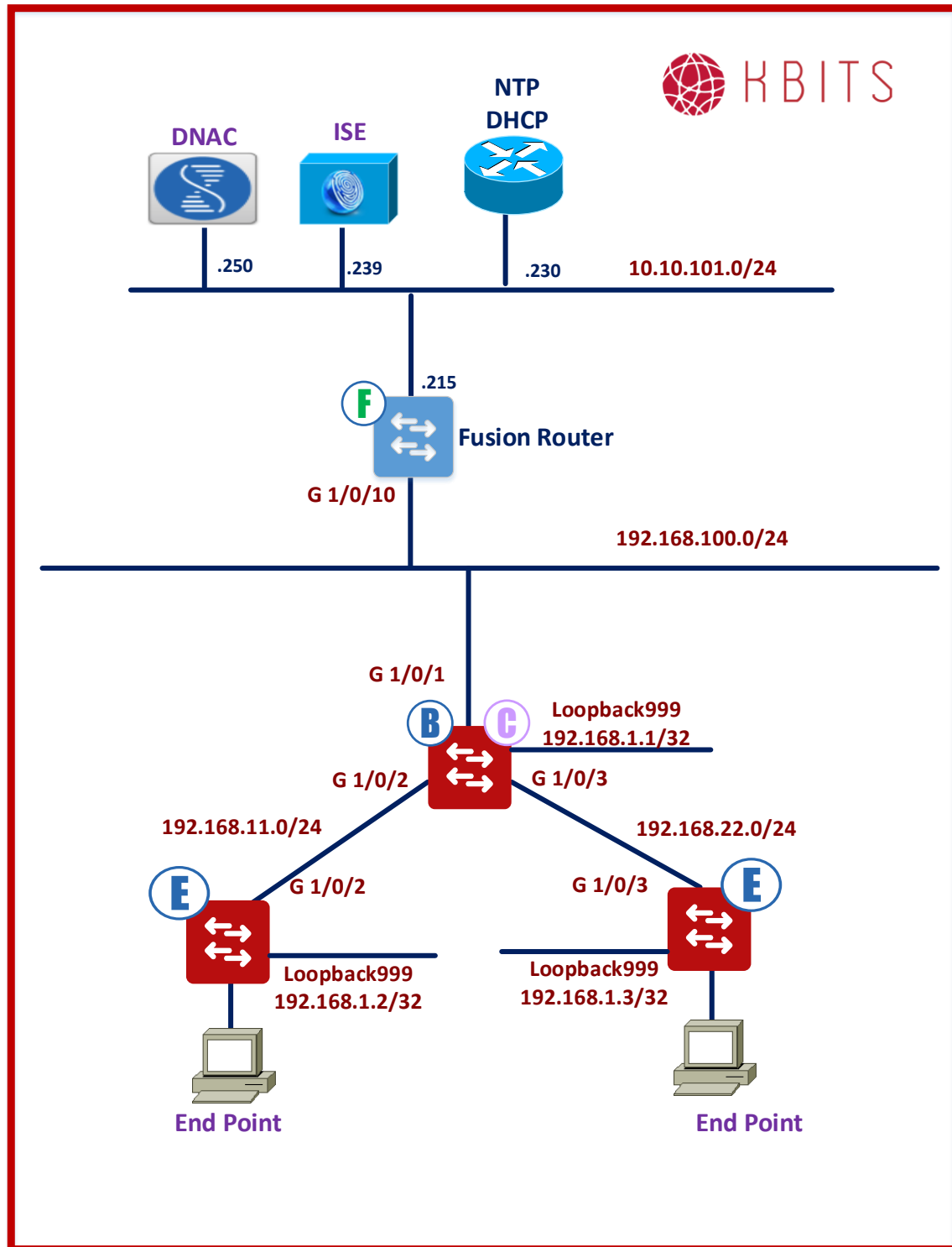
### 9300E1

```
snmp-server community RO ro public  
snmp-server community RW rw private
```

### 9300E2

```
snmp-server community RO ro public  
snmp-server community RW rw private
```

# Lab 9 – Manual Underlay Configuration – Configuring IGP - OSPF



## **Task 1 – Configurig the Underlay Network IGP as OSPF to route the Loopback Networks.**

### **Fusion Router**

```
Router ospf 1
Router-id 0.0.0.100
Network 192.168.100.0 0.0.0.255 area 0
Network 10.10.101.0 0.0.0.255 area 0
Passive-interface vlan 101
```

### **9300CB**

```
Router ospf 1
Router-id 0.0.0.1
Network 192.168.1.0 0.0.0.255 area 0
Network 192.168.11.0 0.0.0.255 area 0
Network 192.168.22.0 0.0.0.255 area 0
Network 192.168.100.0 0.0.0.255 area 0
```

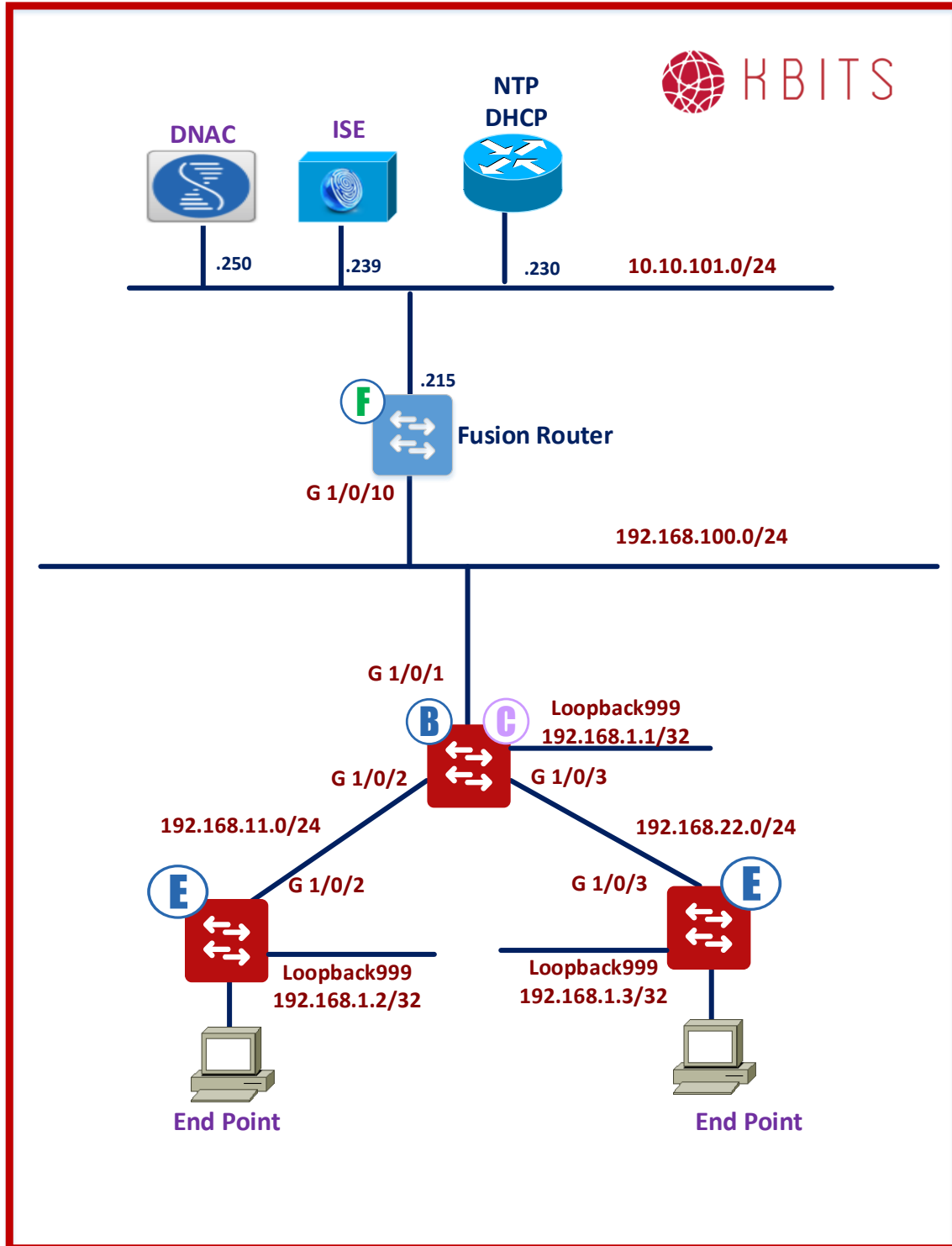
### **9300E1**

```
Router ospf 1
Router-id 0.0.0.2
Network 192.168.1.0 0.0.0.255 area 0
Network 192.168.11.0 0.0.0.255 area 0
```

### **9300E2**

```
Router ospf 1
Router-id 0.0.0.3
Network 192.168.1.0 0.0.0.255 area 0
Network 192.168.22.0 0.0.0.255 area 0
```

# Lab 10 – Manual Underlay Configuration – Device Discovery & Provisioning



## 1. Discover the Underlay

Tools -> Discovery -> Add Discovery

**Note:** Click to make sure you are setting it at the Global Level

→ Discovery Name: **UnderLay**

**IP Address/Range**

→ Discovery Type: **IP Address/Range**

→ IP Address Space: **192.168.1.1 – 192.168.1.3**

**Credentials**

→ CLI: **kbits/FabricAdmin**

→ SNMPv2c Read: **RO**

→ SNMPv2c Write: **RW**

→ Uncheck **SNMPv3**

→ **Click Discover** to Start Discovery

**Note:** Wait for the Fabric devices to be discovered

## 2. Assign the Underlay Devices to the Site

Provision -> Unassigned Devices -> Inventory

→ Select the **9300CB, 9300E1 & 9300E2**

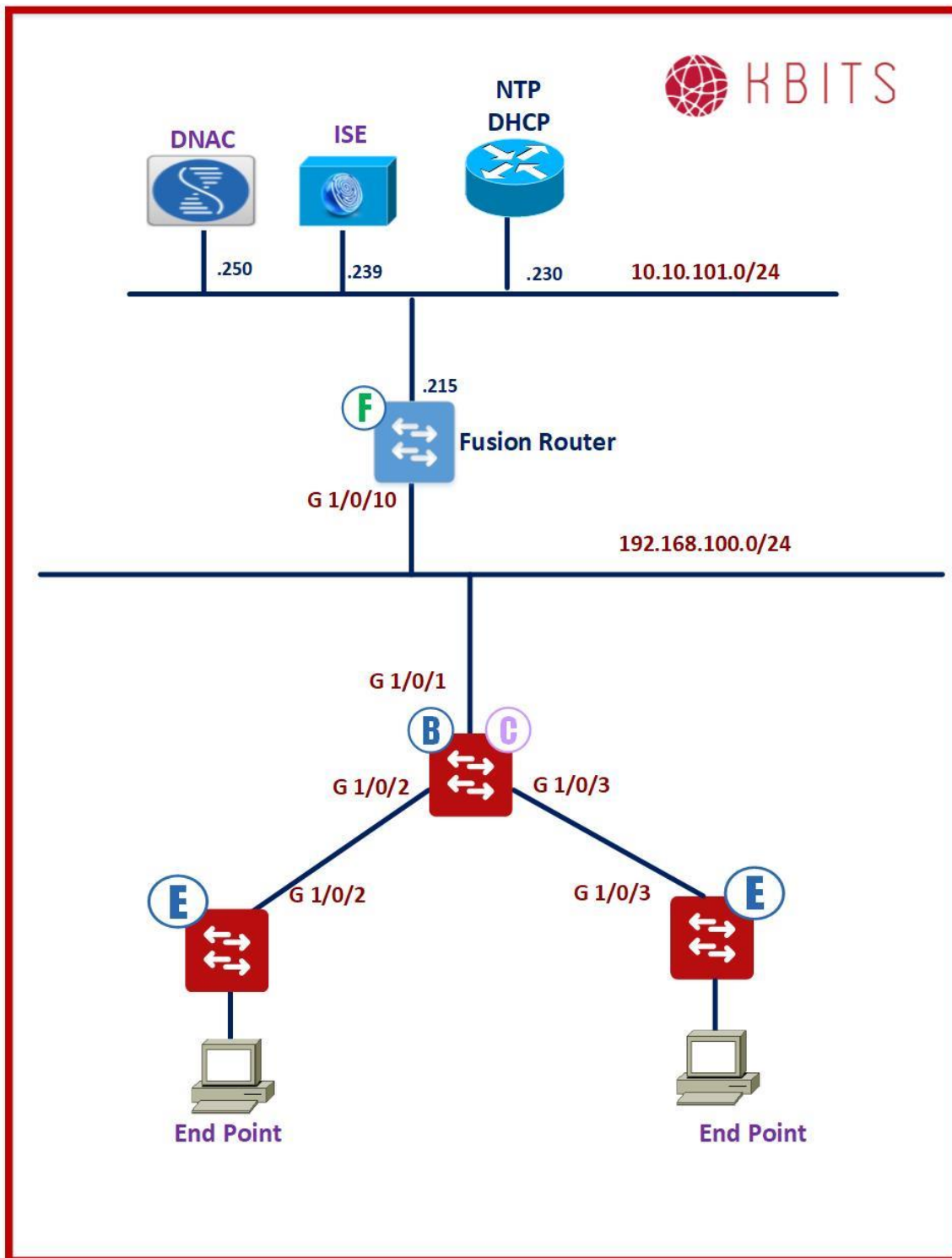
→ Click **Actions** -> **Provision** -> **Assign Device to Site**

→ **Select** -> **Global** -> **Los Angeles** -> **HQ**

→ **Click Assign**

**Note:** The Switches will move under HQ

# Lab 11 – LAN Automation – Seed Device Configuration & Discovery



## Task 1 – Configure Connectivity towards Fusion Router

### 9300CB

```
no ip domain lookup
!  
line con 0
  logg sync
  no exec-timeout
!  
hostname 9300CB
!  
Interface Gig 1/0/1
  switchport mode trunk
!  
vlan 199
!  
ip routing
!  
interface VLAN 199
  ip address 192.168.100.2 255.255.255.0
  no shut
!  
ip route 0.0.0.0 0.0.0.0 192.168.100.1
```

## Task 2 – Configure Telnet/SSH Credentials

### 9300CB

```
username kbits privilege 15 secret Cisco@123
!  
line vty 0 4
  login local
```

## Task 3 – Configure SNMP Parameters for RO & RW Communities

### 9300CB

```
snmp-server community RO ro public
snmp-server community RW rw private
```

## 2. Discover the Seed Device

Tools -> Discovery -> Add Discovery

**Note:** Click to make sure you are setting it at the Global Level

→ Discovery Name: **SEED-DEVICE**

### IP Address/Range

→ Discovery Type: **IP Address/Range**

→ IP Address Space: **192.168.100.2 – 192.168.100.2**

### Credentials

→ CLI: **kbits/FabricAdmin**

→ SNMPv2c Read: **RO**

→ SNMPv2c Write: **RW**

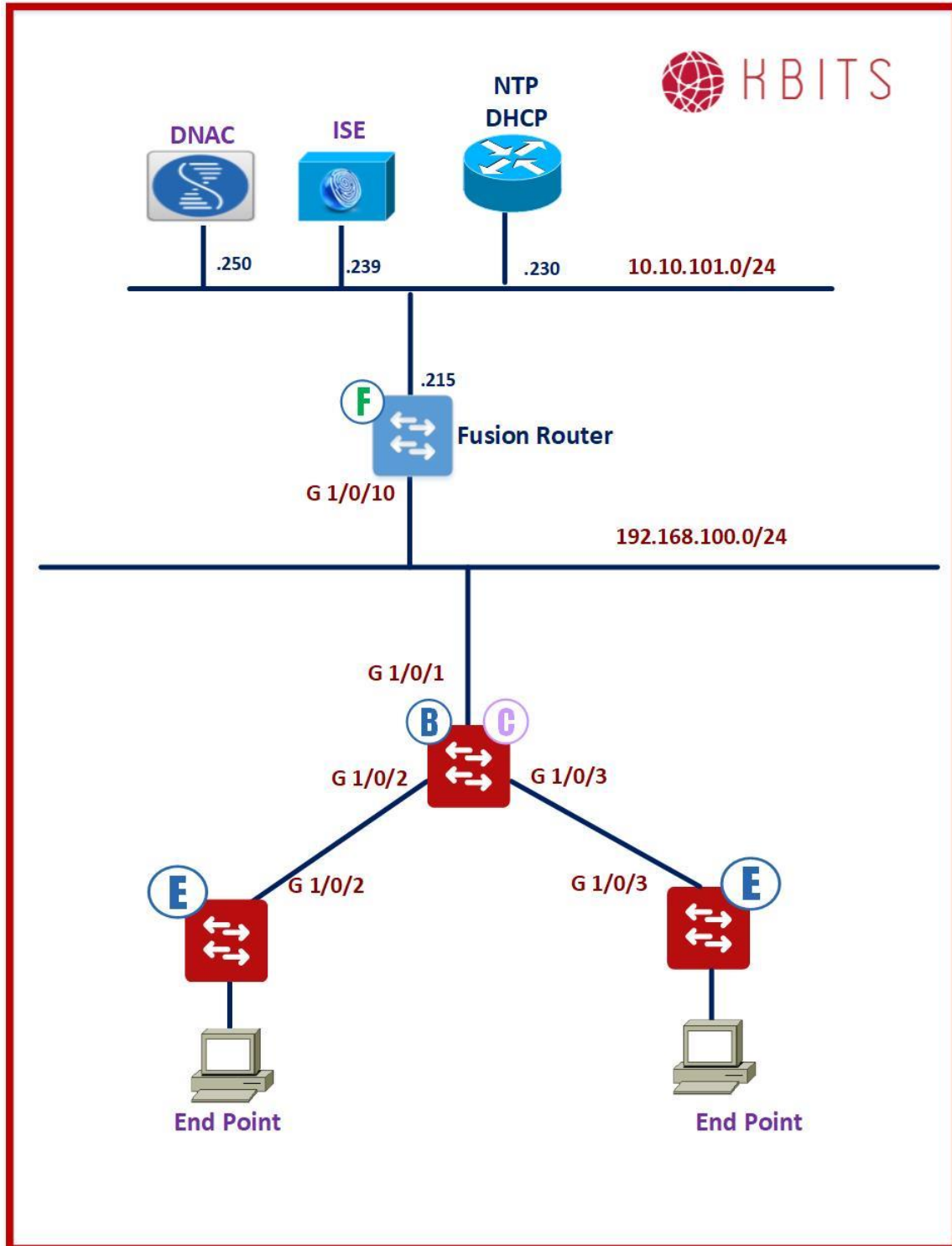
→ Uncheck **SNMPv3**

→ **Click Discover** to Start Discovery

**Note:** Wait for the 9300CB to be discovered



# Lab 12 – LAN Automation – Seed Device Assignment



## **1. Assign the Seed Device**

**Provision -> Unassigned Devices -> Inventory**

→ Select the **9300CB**

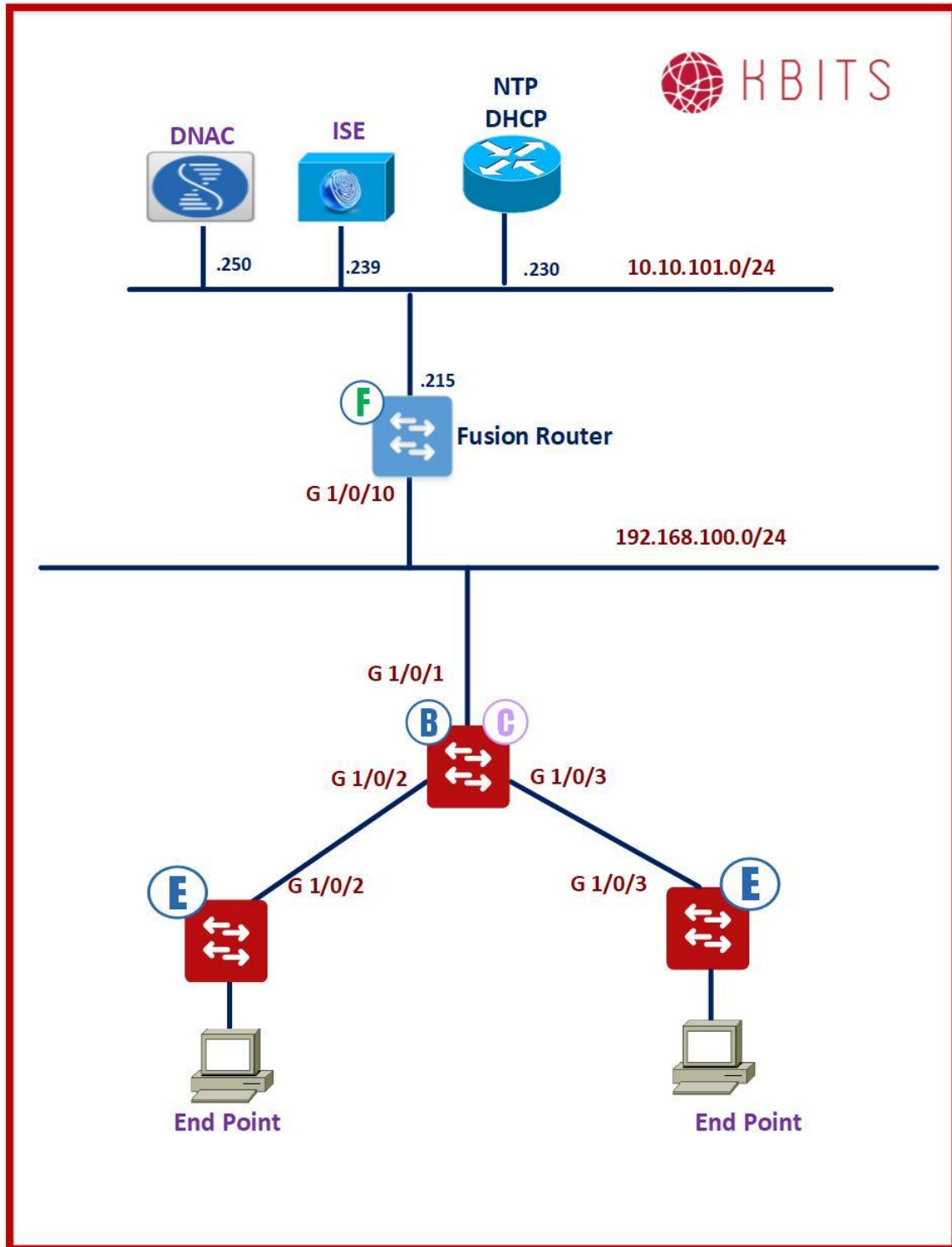
→ Click **Actions -> Provision -> Assign Device to Site**

→ **Select -> Global -> Los Angeles -> HQ**

→ **Click Assign**

**Note: The Switch will move under HQ**

# Lab 13 – LAN Automation – Implementing LAN Automation



## **Task1 - Reserve a Pool for LAN AUTOMATION for the HQ Site**

**Design -> Network Settings -> IP Address Pools -> Global -> Los Angeles -> HQ**

- Click **Reserve**
- IP Addrss Pool Name: **LAN\_AUTOMATION\_HQ**
- Type: **LAN**
- IPv4 Global Pool: **172.20.0.0/24**
- Select **UNDERLAY\_GLOBAL**
- Prefix Length: **/24**
- IPv4 Subnet: **172.20.1.0**
- **Click Save**

## **Task 2 – Configure the Device Credentials for HQ**

**Design -> Network Settings -> Device Credentials -> Global -> Los Angeles -> HQ**

- Select CLI Credentials: **FabricAdmin**
- Select SNMP Credentials: SNMPV2C Read : **RO**
- Select SNMP Credentials: SNMPV2C Write : **RW**
- **Click Save**

### **Task 3 – Configure the Fusion Router with a Static Route for the LAN Automation Pool pointing towards the Seed/Border Device**

#### **Fusion Router**

```
Ip route 172.20.0.0 255.255.0.0 192.168.100.2
```

### **Task 4 - Initiate LAN AUTOMATION**

**Provision -> Inventory -> Action -> Provision -> LAN AUTOMATION -> Start**

- Primary Site: **Global/Los Angeles/HQ**
- Primary Device: **9300CB**
- Selected Ports of Primary Device: **Gig1/0/2 & Gig1/0/3**
  
- Discoverd Device Site: **HQ**
- IP Pool: **LAN\_AUTOMATION\_HQ**
- IS-IS Domain Password: **Cisco@123**
  
- **Click Save to Initiate LAN Automation**
- 

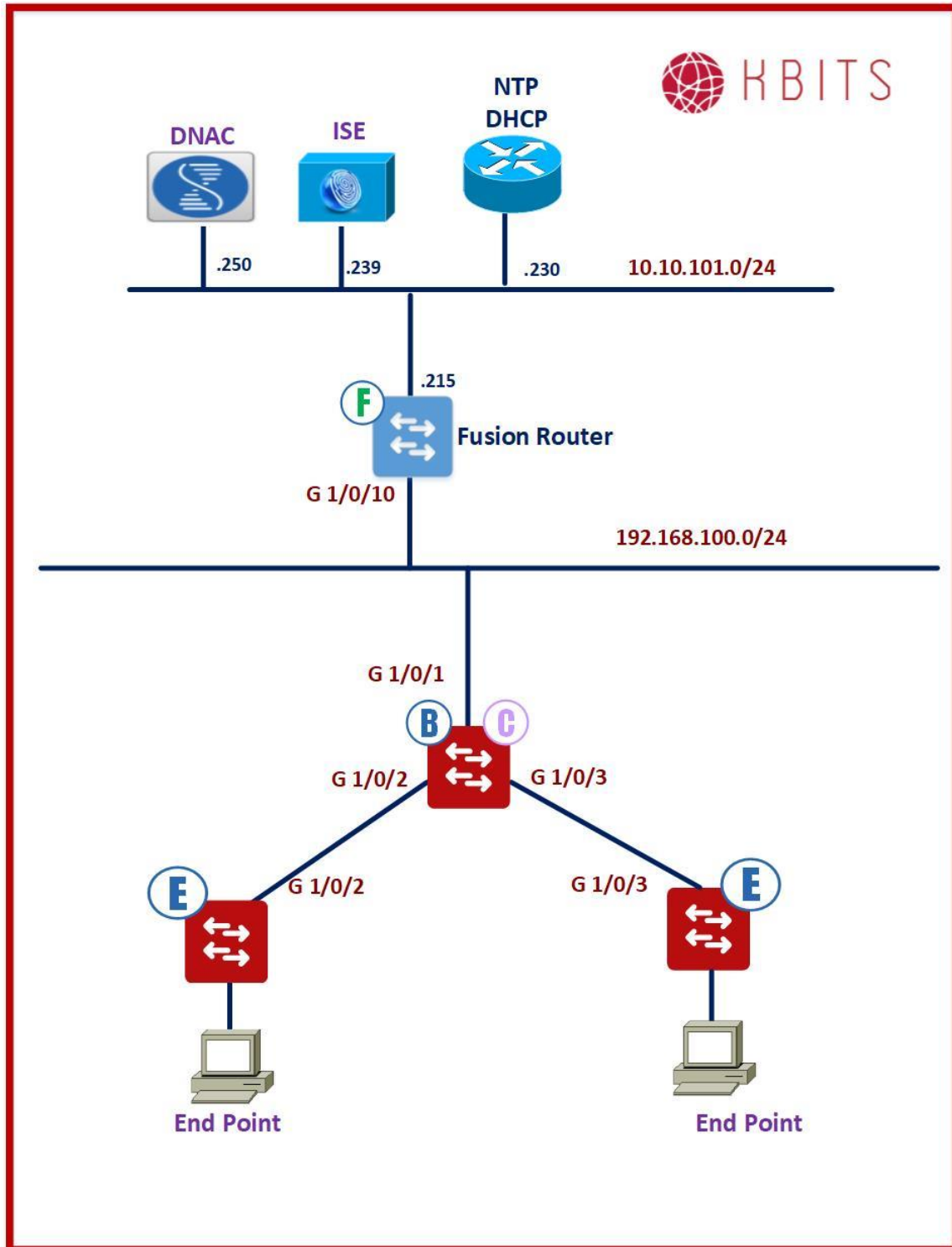
### **Task 5 - Discover the Devices**

**Provision -> Inventory -> Action -> Provision -> LAN AUTOMATION Status**

- Click on **Devices** to verify the devices
- Wait until the switches are **Managed**
- Click **Stop** to stop **LAN AUTOMATION**

**Note: The devices should show up as "Managed" & "Access"**

# Lab 14 – LAN Automation – Provisioning the devices to HQ Site



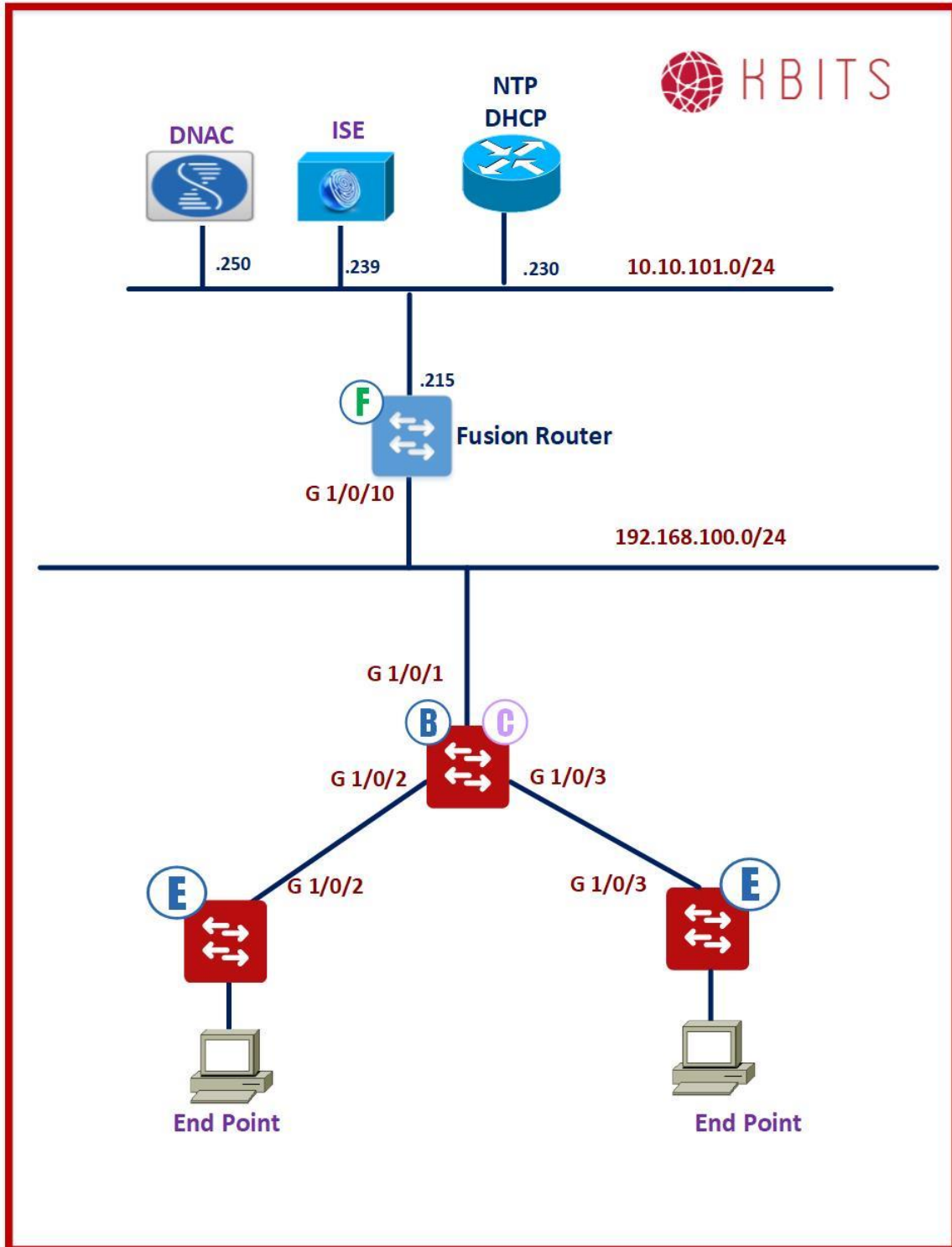
## **Task 1 – Provision the devices as HQ Devices**

**Provision -> Global -> Los Angeles -> HQ**

- Select **“All” the devices**
- Click **Action -> Provision -> Provision the device**
- **Select -> Global -> Los Angeles -> HQ & Check "Apply to all devices"**
- **Click Assign**

**Note: The devices are now available for Device Role Assignment under the HQ Fabric**

# Lab 15 – Reserve the IP Pools for HQ Site for Overlay & Underlay





## **Task 1 – Configure the pools for the IT Department/VN from the OVERLAY\_GLOBAL Pool**

**Design -> Network Settings -> IP Address Pools -> HQ**

- Name: **IT-DATA-1-POOL**
- Prefix Length: **/24**
- IPv4 Subnet: **172.16.1.0**
- Default GW: **172.16.1.254**
- DHCP Server: **10.10.101.230**

- Name: **IT-DATA-2-POOL**
- Prefix Length: **/24**
- IPv4 Subnet: **172.16.2.0**
- Default GW: **172.16.2.254**
- DHCP Server: **10.10.101.230**

- Name: **IT-VOICE-1-POOL**
- Prefix Length: **/24**
- IPv4 Subnet: **172.16.101.0**
- Default GW: **172.16.101.254**
- DHCP Server: **10.10.101.230**

## **Task 2 – Configure the pools for the SALES Department/VN from the OVERLAY\_GLOBAL Pool**

**Design -> Network Settings -> IP Address Pools -> HQ**

- Name: **SALES-DATA-1-POOL**
- Prefix Length: /24
- IPv4 Subnet: **172.16.3.0**
- Default GW: **172.16.3.254**
- DHCP Server: **10.10.101.230**

- Name: **SALES-DATA-2-POOL**
- Prefix Length: /24
- IPv4 Subnet: **172.16.4.0**
- Default GW: **172.16.4.254**
- DHCP Server: **10.10.101.230**

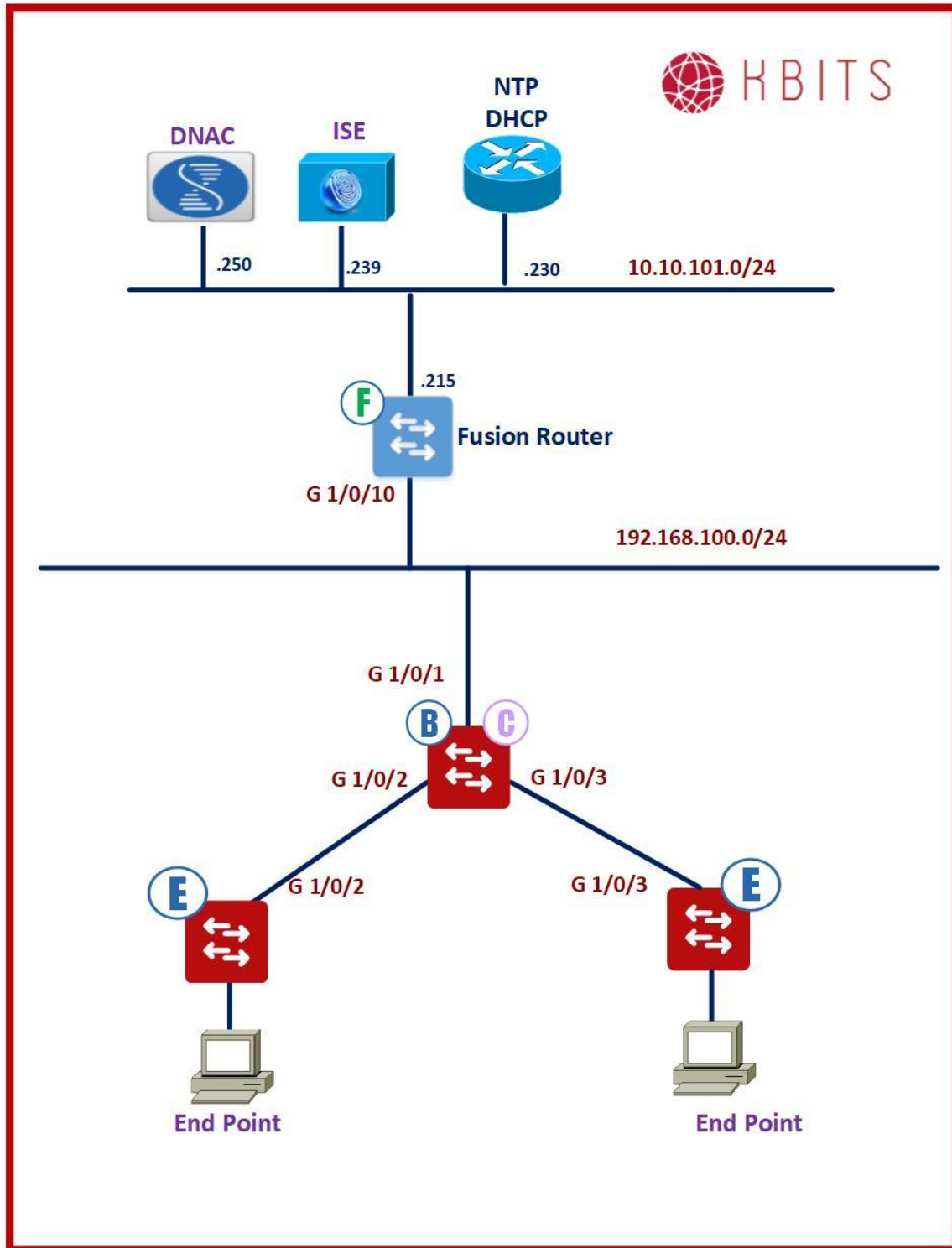
- Name: **SALES-VOICE-1-POOL**
- Prefix Length: /24
- IPv4 Subnet: **172.16.102.0**
- Default GW: **172.16.102.254**
- DHCP Server: **10.10.101.230**

## **Task 3 – Configure the pool for L3HANDOFF from the UNDERLAY\_GLOBAL Pool**

**Design -> Network Settings -> IP Address Pools -> HQ**

- Name: **L3HANDOFF\_POOL**
- Prefix Length: /24
- IPv4 Subnet: **172.20.2.0/24**

# Lab 16 – Create VNs for the Fabric



## **Task 1 – Create the VNs that wil be used in the Fabric**

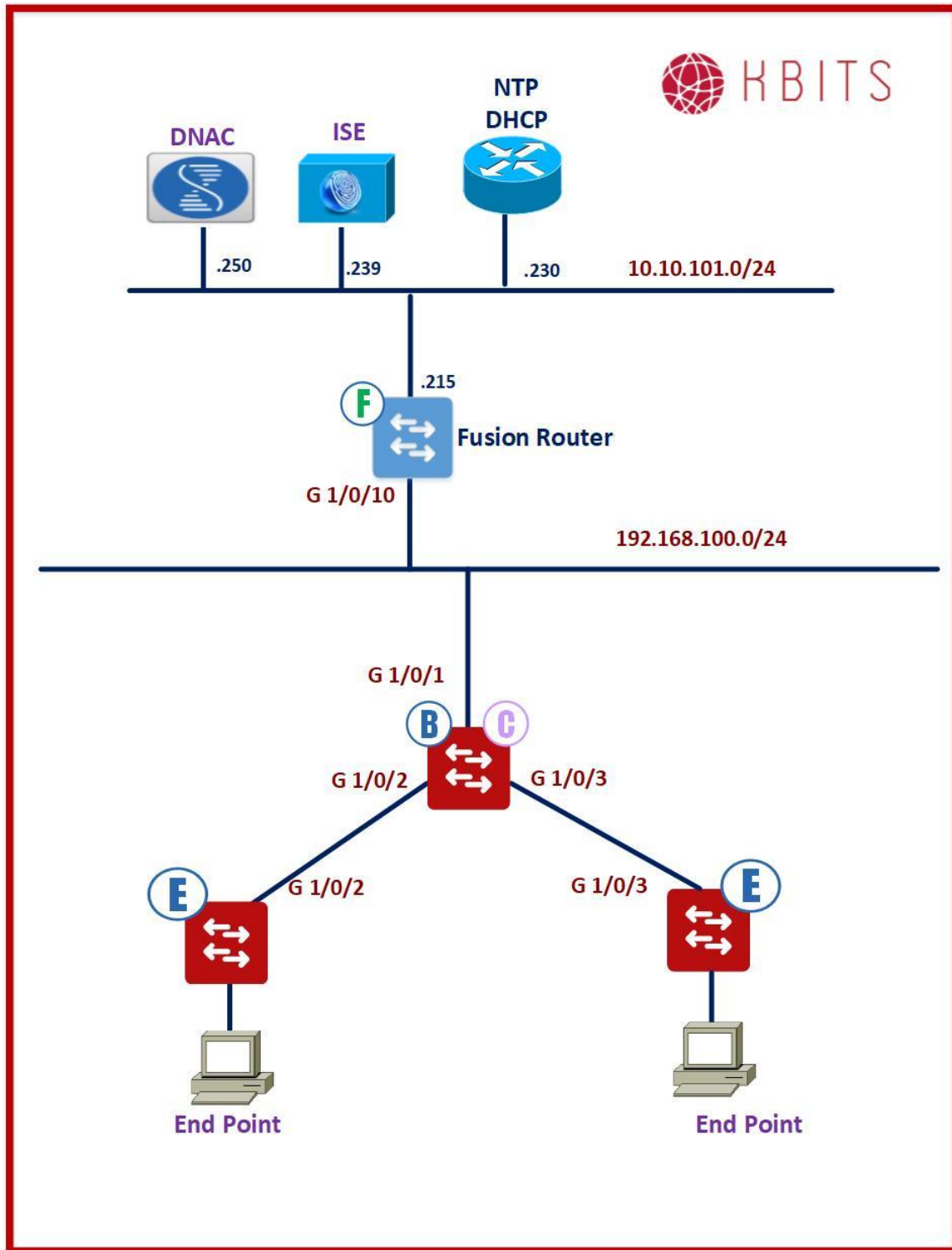
**Policy -> Virtual Network -> Add**

**Add the following VNs**

→ Name: **IT\_VN**

→ Name: **SALES\_VN.**

# Lab 17 - Create the Transit Network - (L3 Handoff)



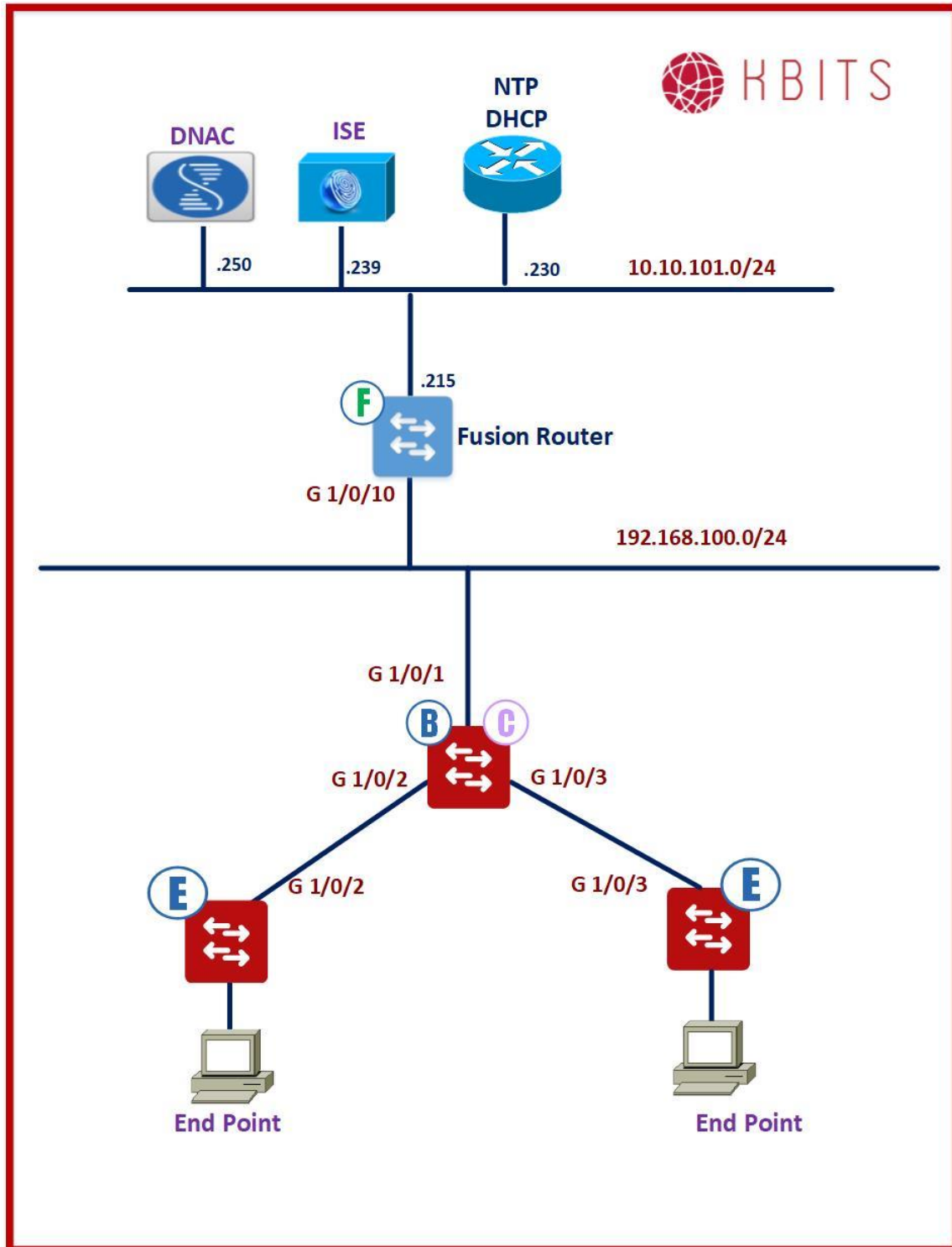
## **Task 1 – Configure the Transit Network (L3 Handoff)**

**Provision -> Fabric -> Add Transit/Peer Network**

- Name: **L3HANDOFF**
- Transit/Type: **IP-Based**
- Protocol: **BGP**
- Type: **ASPlain**
- AS #: **65001 (Fusion Router AS)**

→ **Click Save**

# Lab 18 - Configure Host Onboarding



## Task 1 – Create the Fabric

Provision -> Fabric -> Add Fabric

- Name: **HQ\_FABRIC**
- Add all except for Default
- Click **Add**

## Task 2 – Configuring Host Onboarding

Provision -> Fabric -> HQ-FABRIC -> Los Angeles -> HQ

### Authentication Template

- Select “**Closed Authentication**“
- Click to set it as the “**Default**”

### Virtual Network – IT\_VN

- Select: **IT\_VN**
- Add the following Pools:
  - IP Address Pool: **IT-DATA-1-POOL**
  - Authentication Policy: **IT-DATA-1**
  - Traffic Type: **Data**
  - IP Address Pool: **IT-DATA-2-POOL**
  - Authentication Policy: **IT-DATA-2**
  - Traffic Type: **Data**
  - IP Address Pool: **IT-VOICE-1-POOL**
  - Authentication Policy: **IT-VOICE-1**
  - Traffic Type: **Voice**
- **Click Save**

**Note:** The “**Authentication Policy**” is linked to the ISE **Authentication Profile**

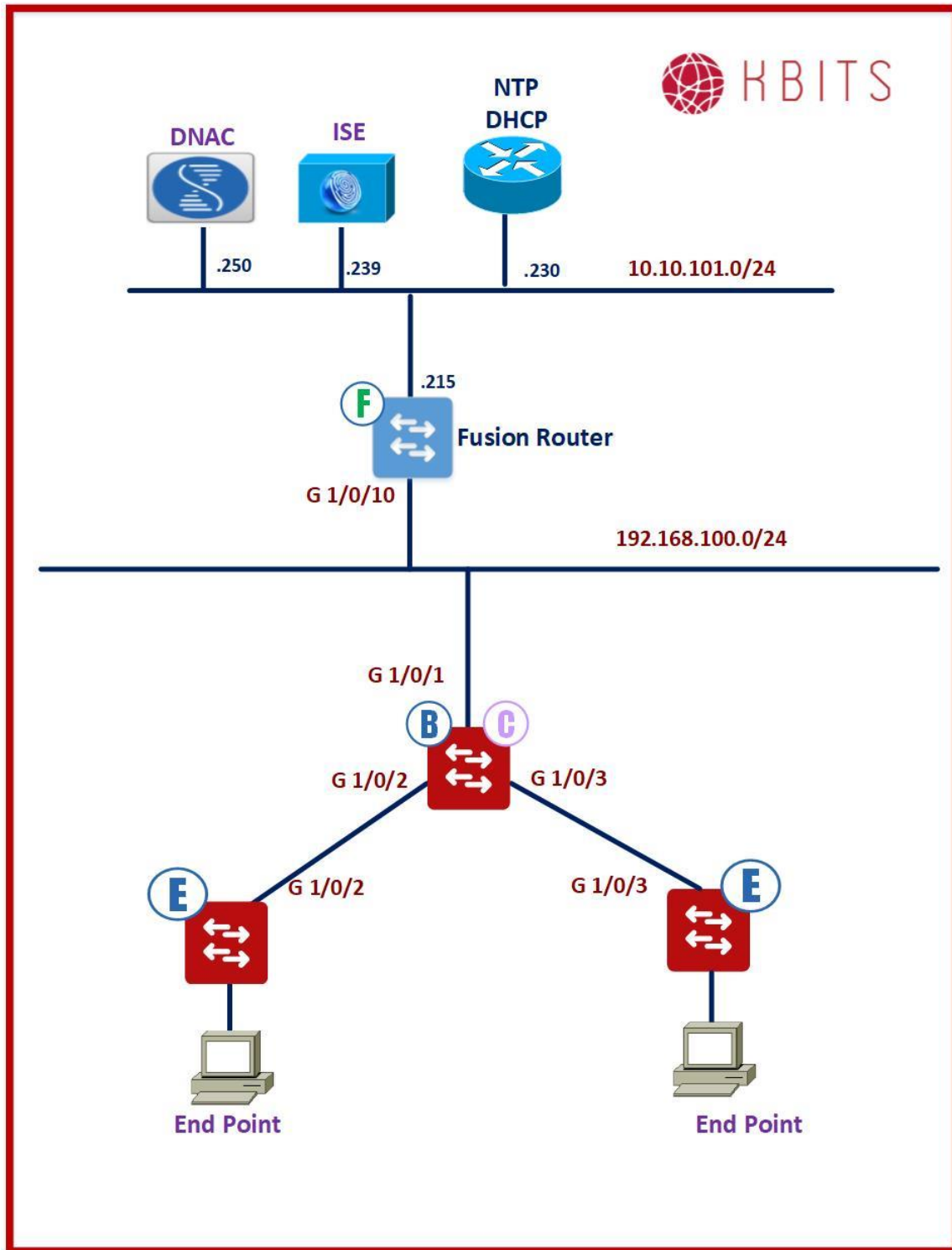


## Virtual Network – SALES\_VN

- Select: **SALES\_VN**
- Add the following Pools:
  - IP Address Pool: **SALES-DATA-1-POOL**
  - Authentication Policy: **SALES-DATA-1**
  - Traffic Type: **Data**
  
  - IP Address Pool: **SALES-DATA-2-POOL**
  - Authentication Policy: **SALES-DATA-2**
  - Traffic Type: **Data**
  
  - IP Address Pool: **SALES-VOICE-1-POOL**
  - Authentication Policy: **SALES-VOICE-1**
  - Traffic Type: **Voice**
  
- **Click Save**

**Note:** The “Authentication Policy” is links to the ISE **Authentication Profile**

# Lab 19 - Configuring & Provisioning the Control / Border Devices

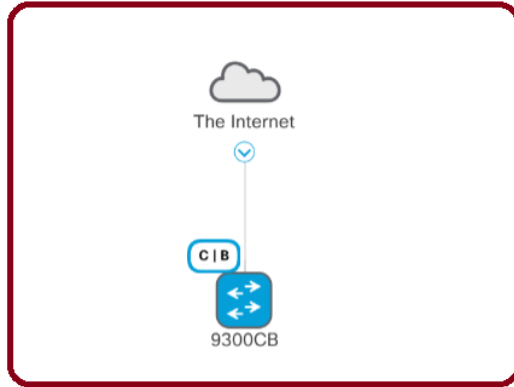


## **Task 1 – Provision the 9300CB as the Control and Border Device**

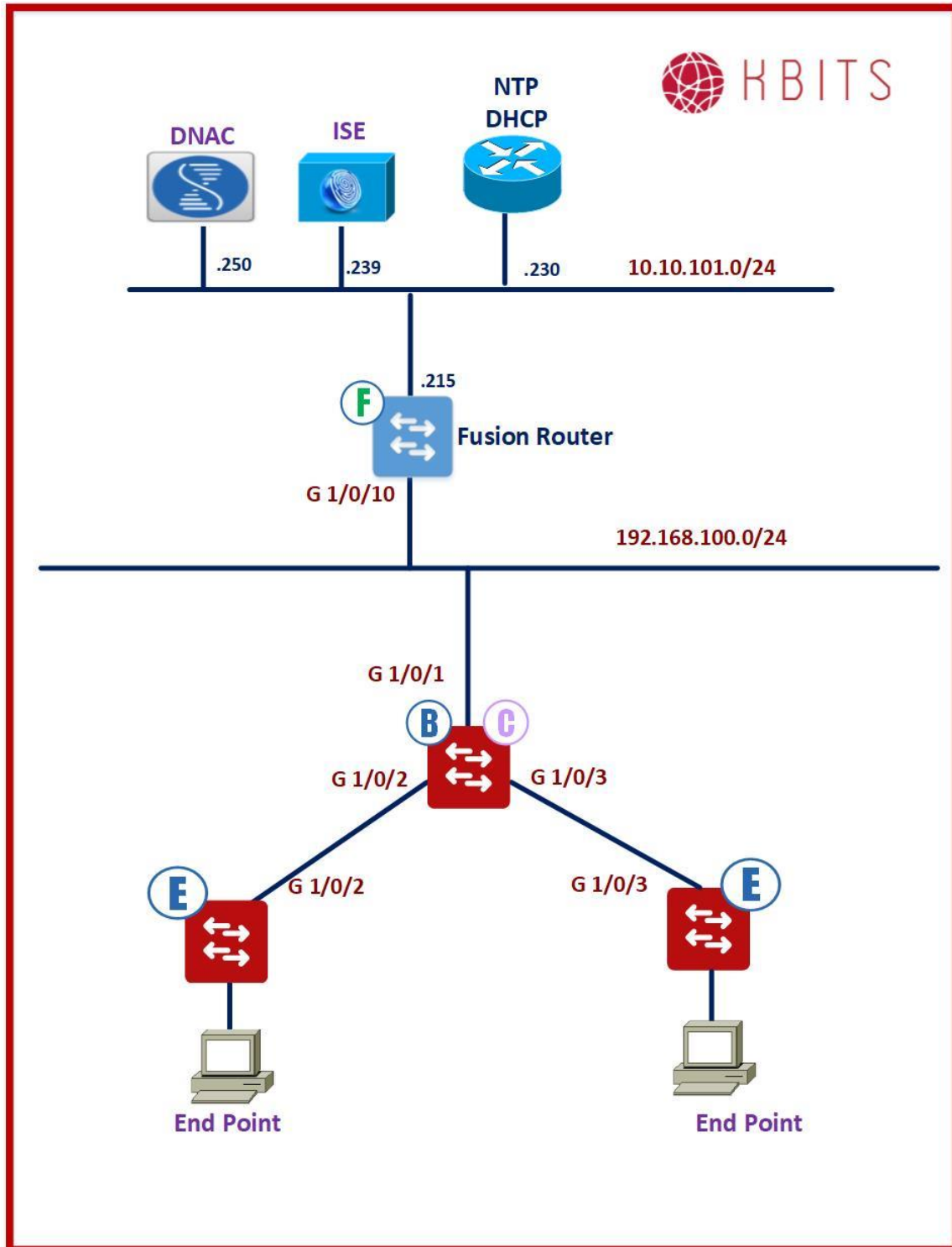
**Fabric -> HQ\_FABRIC -> Los Angeles -> HQ**

- Select **9300CB**
- Slide to select **Control**
- Slide to select **Border**
- **Border L3Handoff Configuration Parameters:**
- **Select Type ASPLAIN**
- Local AS: **65002**
- Default to all Virtual Networks = **Checked**
- **Uncheck** the Do not import External Routes
- Pool: **L3HANDOFF\_POOL**
- Click **Add**
- Select **L3HANDOFF**
- Click **“Add”** to add the Interface **G 1/0/1**
- Select all the **VNs**
- **Click Save & Add**
- **Click Add**
- **Click Save**

**Note: The Device should turn Blue indicating that it is in the Fabric**



# Lab 20 - Configuring & Provisioning the Fabric Edge Devices



## **Task 1 – Provision the HQ-1 as the Fabric Edge Node**

**Fabric -> HQ\_FABRIC -> Los Angeles -> HQ**

→ Select **HQ-1**

→ Slide to select **Edge**

→ **Click Add**

## **Task 2 – Provision the HQ-2 as the Fabric Edge Node**

**Fabric -> HQ\_FABRIC -> Los Angeles -> HQ**

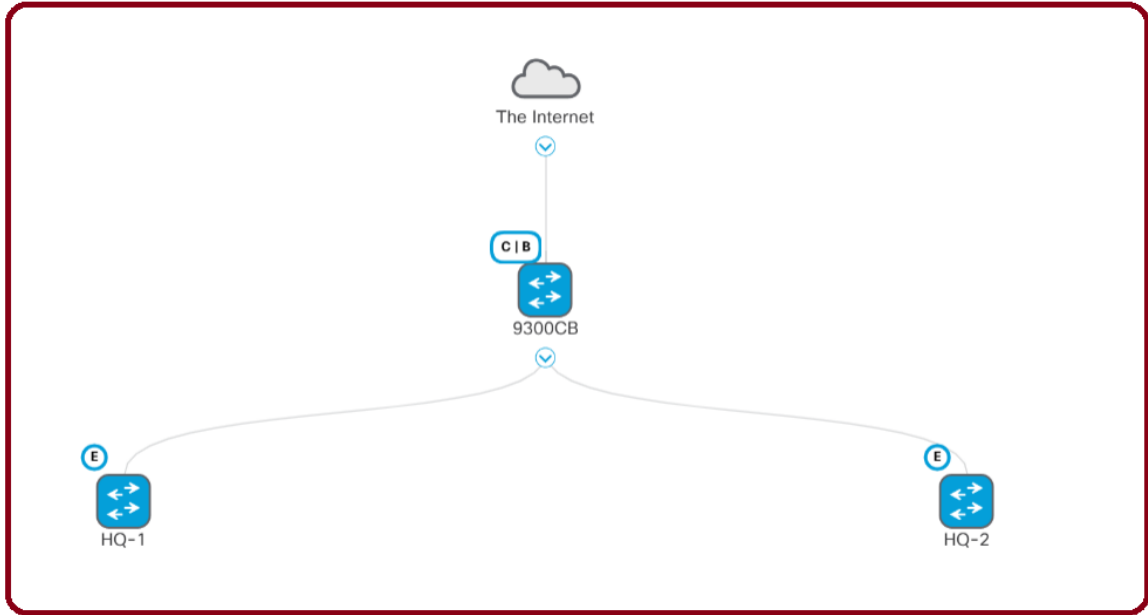
→ Select **HQ-2**

→ Slide to select **Edge**

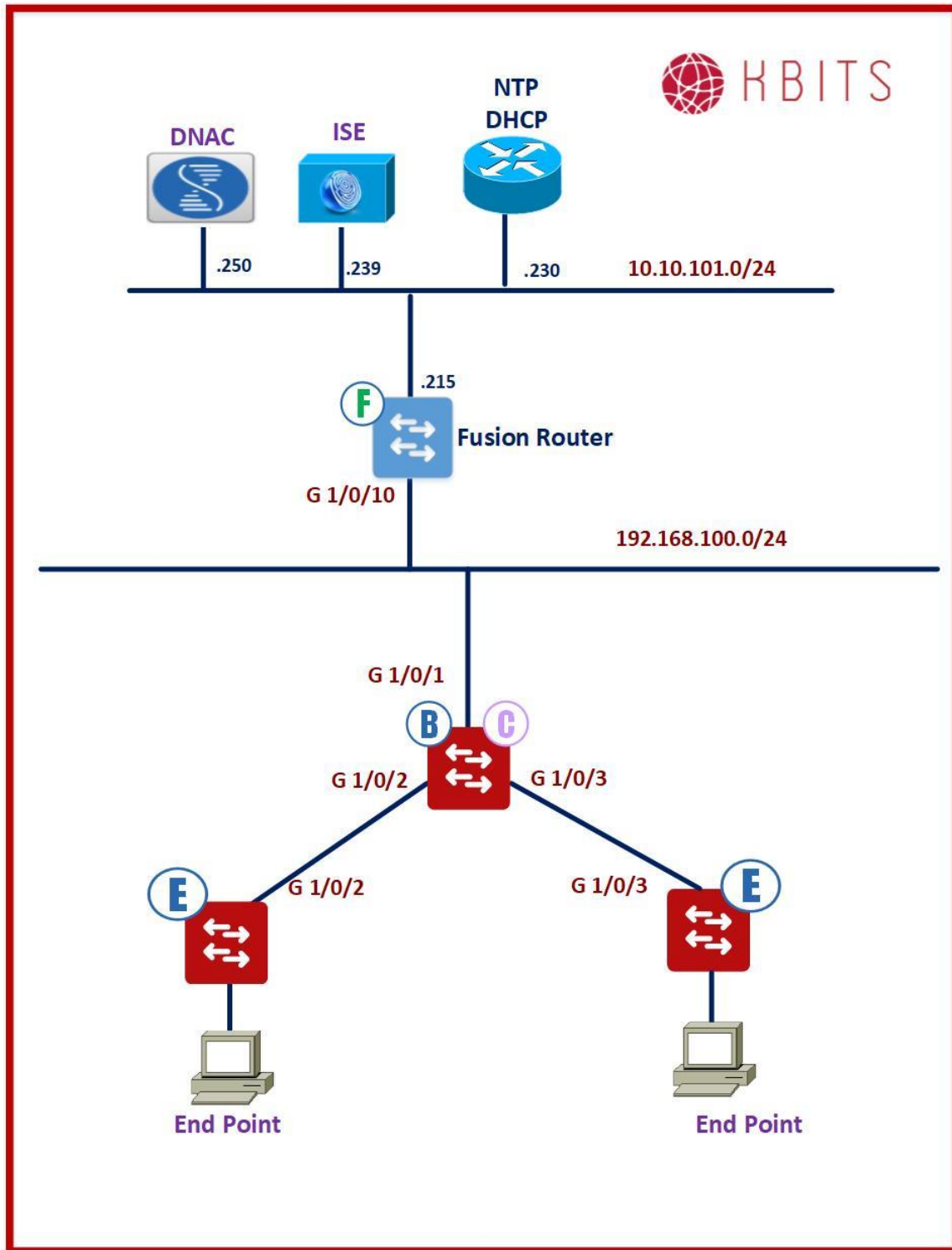
→ **Click Add**

→ **Click Save**

**Note: The Device should turn Blue indicating that it is in the Fabric**



# Lab 21 - Configure the Fusion Router - VRF, SVI, BGP & Route Leaking





## Task 1 – Configure the VRFs to match the Border Interfaces

### Fusion Router

```
vrf definition IT_VN
rd 1:4099
!
address-family ipv4
route-target export 1:4099
route-target import 1:4099
exit-address-family

!
vrf definition SALES_VN
rd 1:4100
!
address-family ipv4
route-target export 1:4100
route-target import 1:4100
exit-address-family
```

## 2. Configure the Interfaces to match the Border Interfaces

### Fusion Router

```
vlan 3001-3003
!!! You might need to enable VTP Transparent Mode to make it work
!
interface Vlan3001
description vrf interface to External router
ip address 172.20.2.2 255.255.255.252
no shut
!
interface Vlan3002
description vrf interface to External router
vrf forwarding IT_VN
ip address 172.20.2.6 255.255.255.252
no shut
!
interface Vlan3003
description vrf interface to External router
vrf forwarding SALES_VN
ip address 172.20.2.10 255.255.255.252
no shut
```

### 3. Configure BGP between Fusion & Border

#### Fusion Router

```
router bgp 65001
neighbor 172.20.2.1 remote-as 65002
neighbor 172.20.2.1 update-source Vlan3001
!
address-family ipv4
neighbor 172.20.2.1 activate
neighbor 172.20.2.1 default-originate
network 10.10.101.0 mask 255.255.255.0
!
address-family ipv4 vrf SALES_VN
neighbor 172.20.2.9 remote-as 65002
neighbor 172.20.2.9 update-source Vlan3003
neighbor 172.20.2.9 activate
neighbor 172.20.2.9 default-originate
network 10.10.101.0 mask 255.255.255.0
!
address-family ipv4 vrf IT_VN
neighbor 172.20.2.5 remote-as 65002
neighbor 172.20.2.5 update-source Vlan3002
neighbor 172.20.2.5 activate
neighbor 172.20.2.5 default-originate
network 10.10.101.0 mask 255.255.255.0
```

### 4. Configure Route Leaking from Global into VRFs

#### Fusion Router

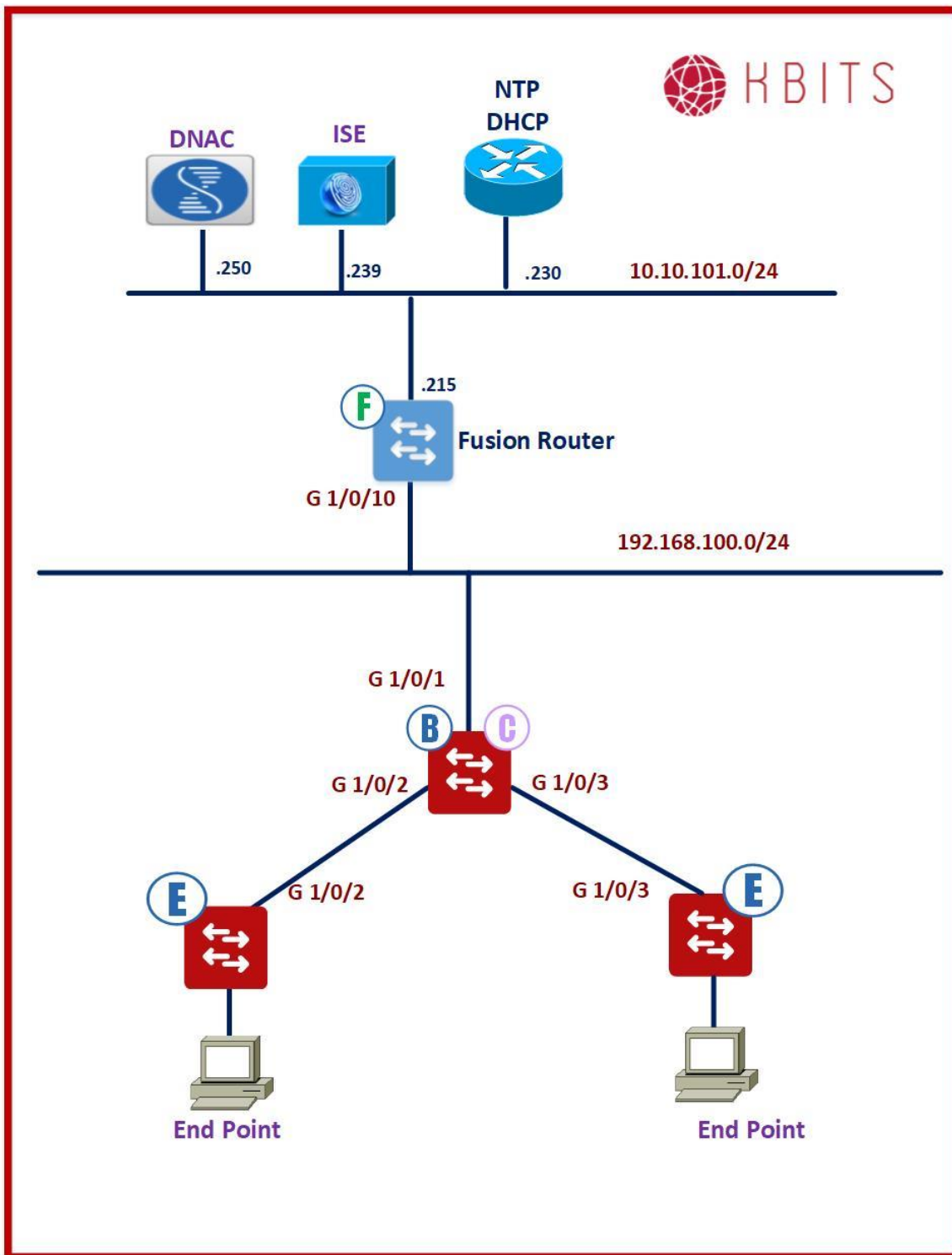
```
ip prefix-list GLOBAL seq 5 permit 10.10.101.0/24
!
route-map GLOBAL permit 10
match ip address prefix-list GLOBAL
!
vrf definition IT_VN
address-family ipv4
import ipv4 unicast map GLOBAL
!
vrf definition SALES_VN
address-family ipv4
import ipv4 unicast map GLOBAL
```

## 5. Configure Route Leaking from VRF into Global

### **Fusion Router**

```
ip route 172.16.1.0 255.255.255.0 Vlan3002
ip route 172.16.2.0 255.255.255.0 Vlan3002
ip route 172.16.3.0 255.255.255.0 Vlan3003
ip route 172.16.4.0 255.255.255.0 Vlan3003
ip route 172.20.2.4 255.255.255.252 Vlan3002
ip route 172.20.2.8 255.255.255.252 Vlan3003
ip route 172.16.101.0 255.255.255.0 Vlan3002
ip route 172.16.102.0 255.255.255.0 Vlan3003
```

# Lab 22 – Configure User & Groups on ISE



## **Task 1 – Configure User Identity Groups in ISE**

**Administration -> Identity Management -> Groups -> User Identity Groups -> Create**

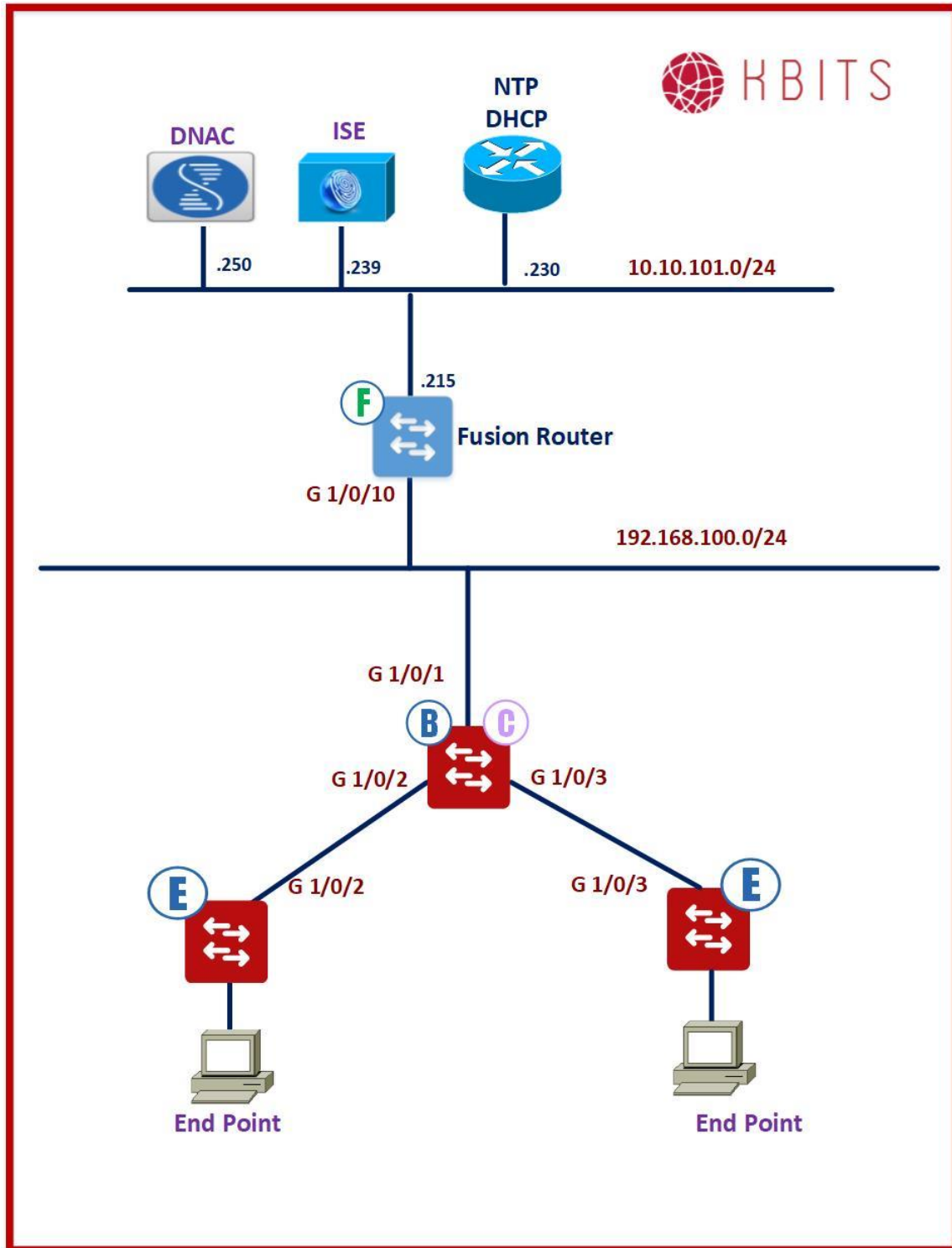
- Name: **IT-DATA-1**
- Name: **IT-DATA-2**
- Name: **IT-VOICE**
- Name: **SALES-DATA-1**
- Name: **SALES-DATA-2**
- Name: **SALES-VOICE**

## **Task 2 – Configure Users on ISE & Assign them to the appropriate Groups**

**Administration -> Identity Management -> Identities -> Create**

- Name: **IT1**
- Password: **Cisco@123**
- Group: **IT-DATA-1**
  
- Name: **IT2**
- Password: **Cisco@123**
- Group: **IT-DATA-2**
  
- Name: **SALES1**
- Password: **Cisco@123**
- Group: **SALES-DATA-1**
  
- Name: **SALES2**
- Password: **Cisco@123**
- Group: **SALES-DATA-2**

# Lab 23 - Configure Authorization Profiles for the DNAC VNs

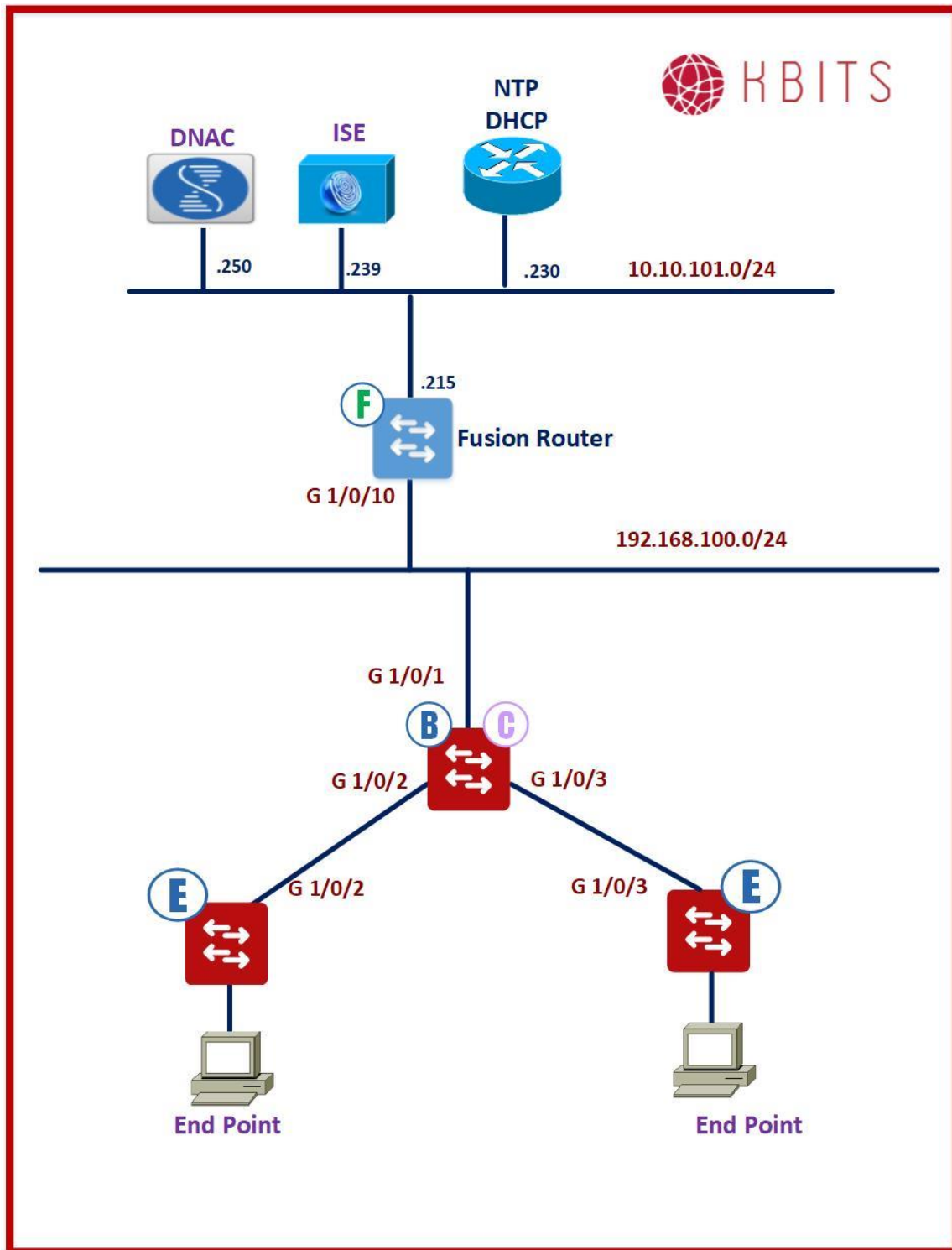


## **Task 1 – Configure the Authorization Profiles to link the Authorization Profile name in DNAC for the VNs & Pools**

**Policy -> Policy Elements -> Results -> Authorization -> Authorization Profile -> Create**

- **Name: IT-DATA-1-PROF**
- **VLAN: IT-DATA-1 (Copy from DNAC)**
  
- **Name: IT-DATA-2-PROF**
- **VLAN: IT-DATA-2 (Copy from DNAC)**
  
- **Name: SALES-DATA-1-PROF**
- **VLAN: SALES-DATA-1 (Copy from DNAC)**
  
- **Name: SALES-DATA-2-PROF**
- **VLAN: SALES-DATA-2 (Copy from DNAC)**

# Lab 24 - Configure Authorization Policies for the DNAC VNs





## **Task 1 – Configure the Authorization Policies to assign the IT & SALES Groups appropriate profiles for 802.1x authentication**

**Policy -> Policy Sets -> default -> Authorization Policies -> Insert at the Top**

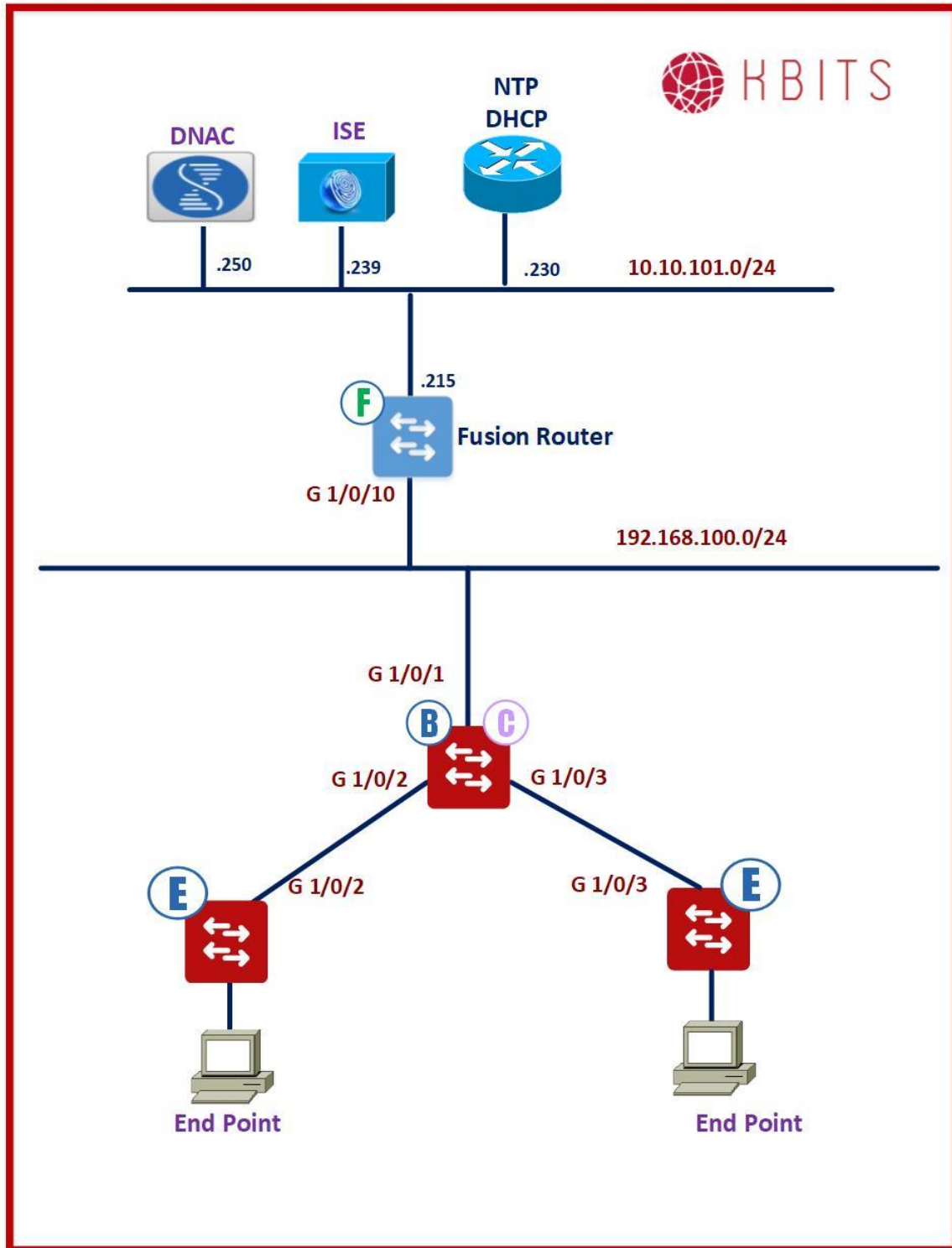
- Name: **IT-DATA-1-POLICY**
- Identity Group: **IT-DATA-1**
- Authentication Method: **Wired\_802.1x**
- Permission: **IT-DATA-1-PROF**

- Name: **IT-DATA-2-POLICY**
- Identity Group: **IT-DATA-2**
- Authentication Method: **Wired\_802.1x**
- Permission: **IT-DATA-2-PROF**

- Name: **SALES-DATA-1-POLICY**
- Identity Group: **SALES-DATA-1**
- Authentication Method: **Wired\_802.1x**
- Permission: **SALES-DATA-1-PROF**

- Name: **SALES-DATA-2-POLICY**
- Identity Group: **SALES-DATA-2**
- Authentication Method: **Wired\_802.1x**
- Permission: **SALES-DATA-2-PROF**

# Lab 25 - Configure the DHCP Server to provide IP Configuration to Clients



## **Task 1 – Configure the Exclusions for the 4 Data Pools**

### **Access Server**

```
ip dhcp excluded-address 172.16.1.1 172.16.1.50
ip dhcp excluded-address 172.16.1.254
ip dhcp excluded-address 172.16.2.1 172.16.2.50
ip dhcp excluded-address 172.16.2.254
ip dhcp excluded-address 172.16.3.1 172.16.3.50
ip dhcp excluded-address 172.16.3.254
ip dhcp excluded-address 172.16.4.1 172.16.4.50
ip dhcp excluded-address 172.16.4.254
```

## **Task 2 – Configure the 4 Data Pools with the default gateway being the Last IP in the network**

### **Access Server**

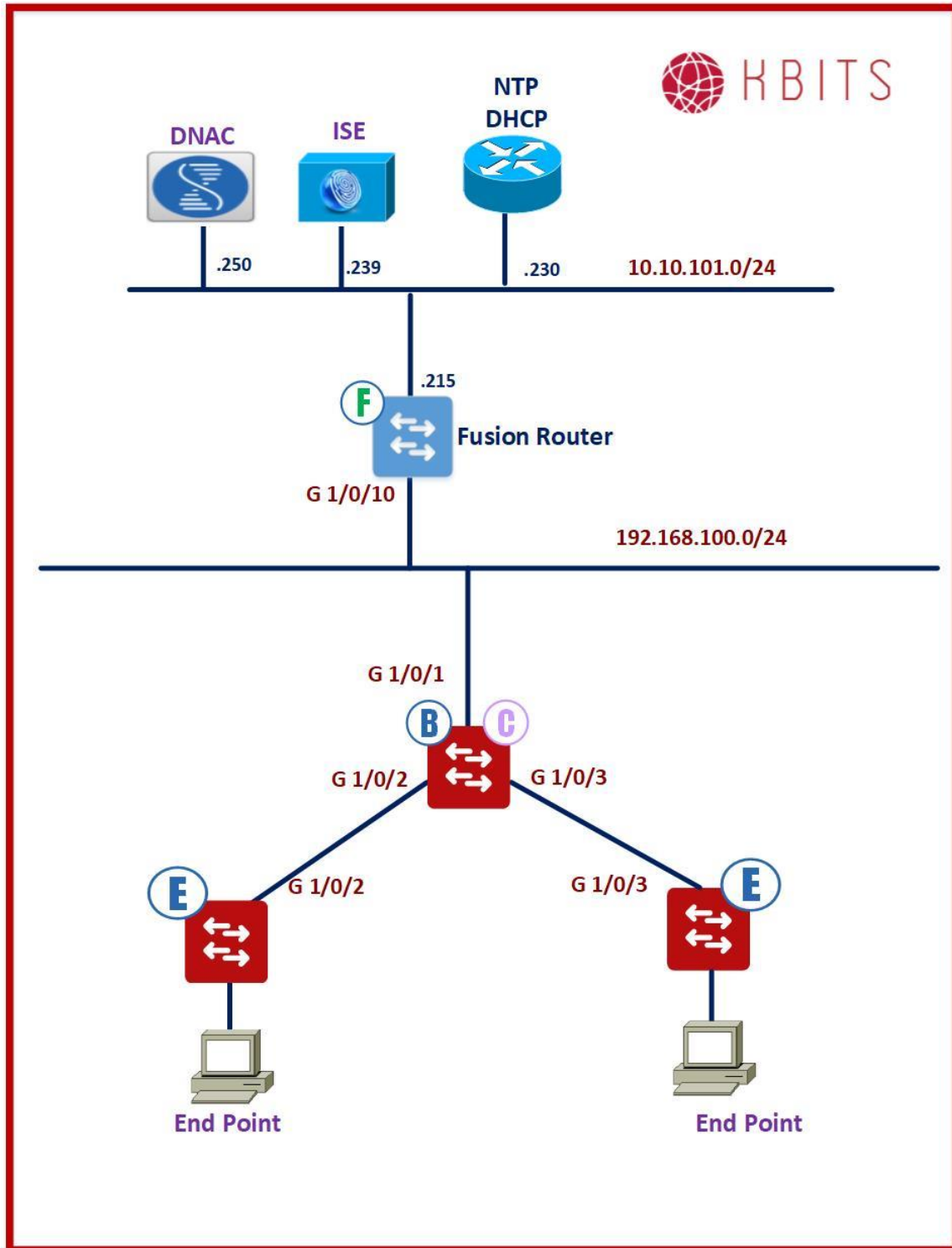
```
ip dhcp pool IT-DATA-1
  network 172.16.1.0 255.255.255.0
  default-router 172.16.1.254
!
ip dhcp pool IT-DATA-2
  network 172.16.2.0 255.255.255.0
  default-router 172.16.2.254
!
ip dhcp pool SALES-DATA-1
  network 172.16.3.0 255.255.255.0
  default-router 172.16.3.254
!
ip dhcp pool SALES-DATA-2
  network 172.16.4.0 255.255.255.0
  default-router 172.16.4.254
```

## **Task 3 – Configure static Routes for the Overlay Data networks pointing towards the Fusion Router**

### **Access Server**

```
ip route 172.16.0.0 255.255.0.0 10.10.101.215
ip route 172.20.0.0 255.255.0.0 10.10.101.215
```

# Lab 26 – Verifying Macro Segmentation



## **Task 1 – Devices within the same VN should be able to communicate to each other**

- Configure the Native Windows supplicant on **ES1** to log in using **IT1** credentials.
- Configure the Native Windows supplicant on **ES2** to log in using **IT2** credentials.

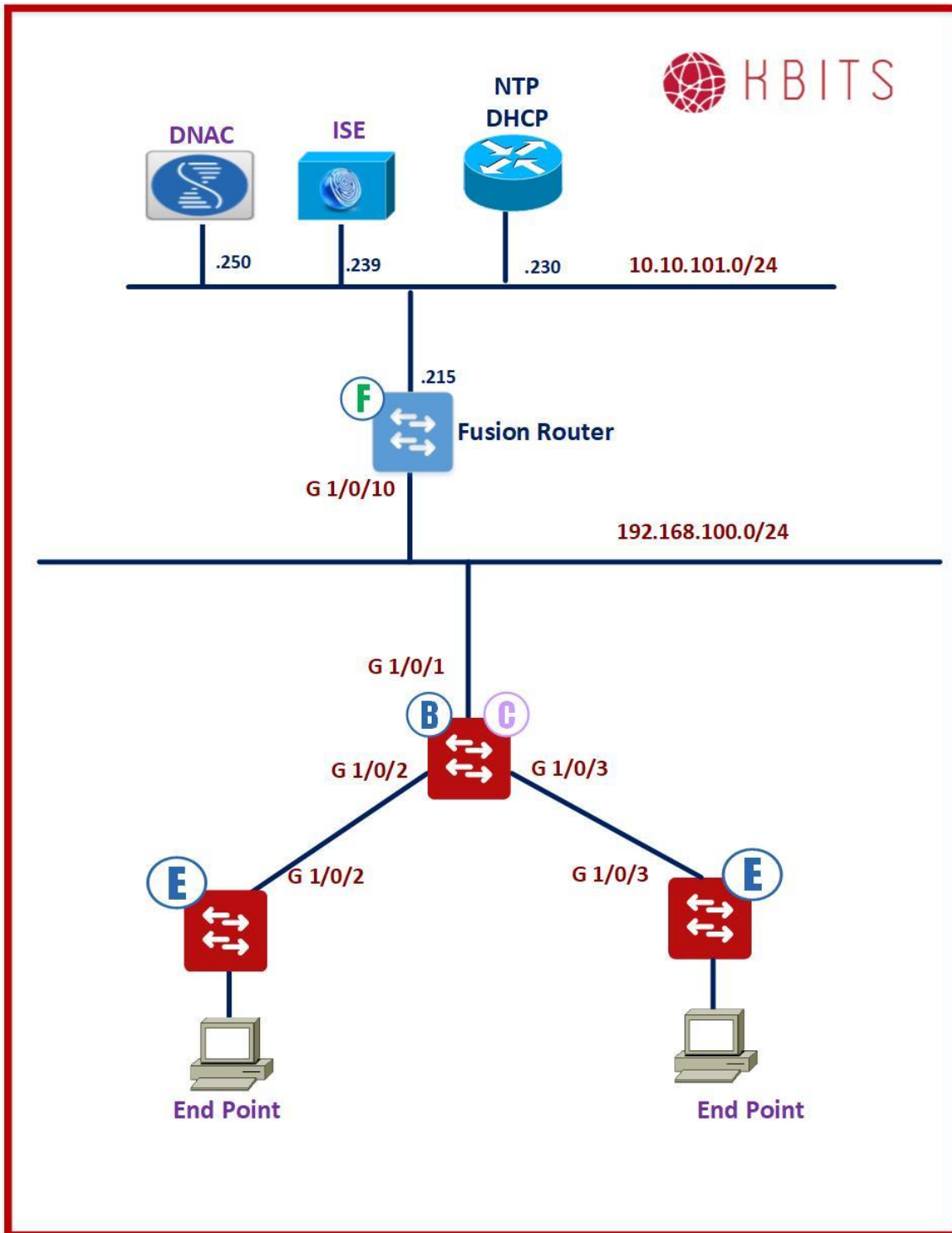
**Verification:** They should be in 2 different subnets but should be able to communicate to each other.

## **Task 2 – Devices in 2 different segments should not be able to communicate to each other**

- Change the credentials on **ES2** to **SALES1**.

**Verification:** They should not be able to communicate to each other as they are in 2 different VNs.

# Lab 27 – Micro Segmentation – Creating SGTs



## **Task 1 – Configure SGT for IT Subnets on DNAC**

**Policy -> Group Based Access Control -> Scalable Groups -> Create**

- Name: **IT\_DATA\_1**
- SGT: **6001**
- VN: **IT\_VN**

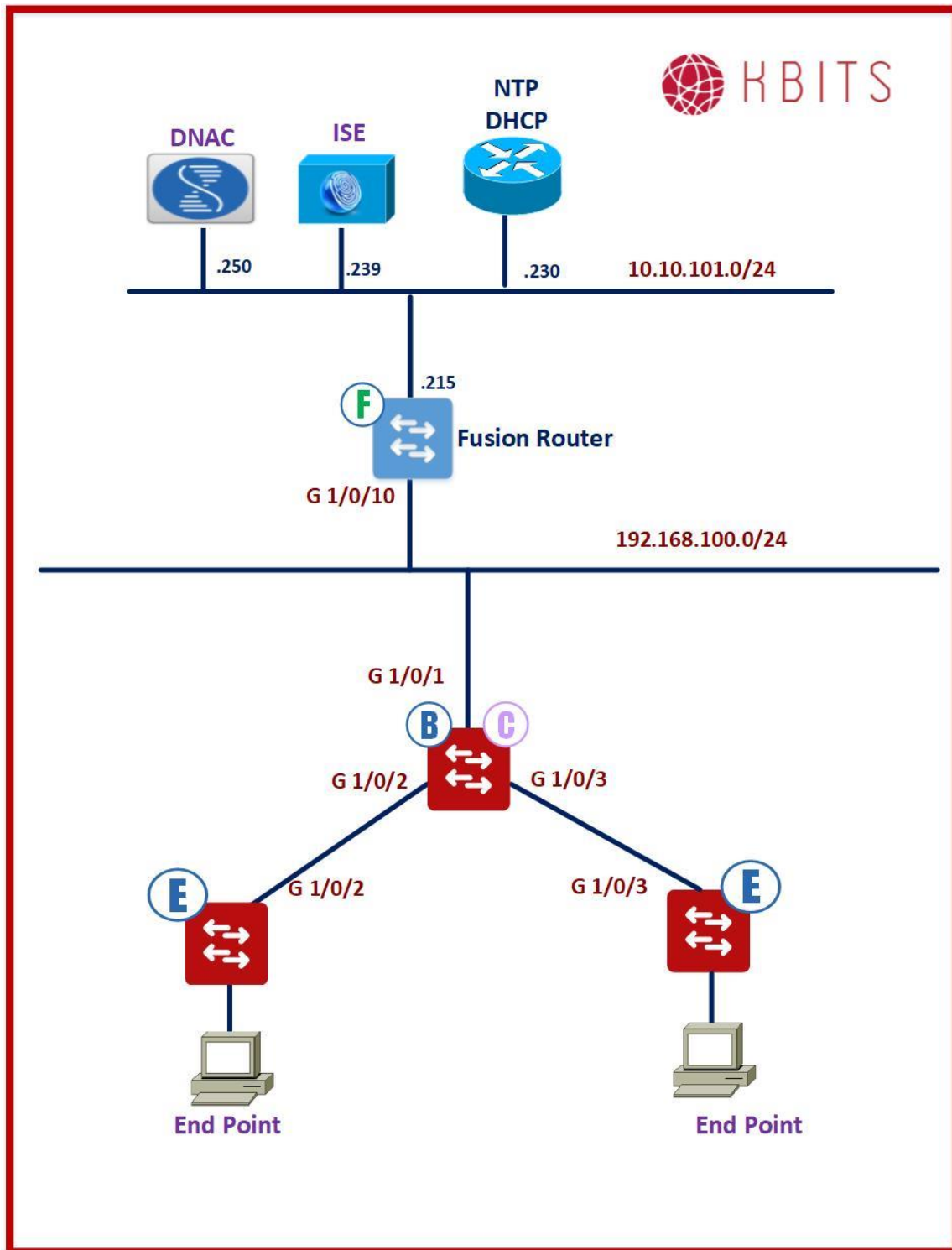
- Name: **IT\_DATA\_2**
- SGT: **6002**
- VN: **IT\_VN**

## **Task 2 – Verify the SGTs are propagated to ISE**

**Work Centers -> TrustSe -> Components -> Security Groups**

**The IT\_DATA\_1 & IT\_DATA\_2 SGTs should be available in ISE**

# Lab 28 – Micro Segmentation – Assigning SGTS via Authorization Policies on ISE



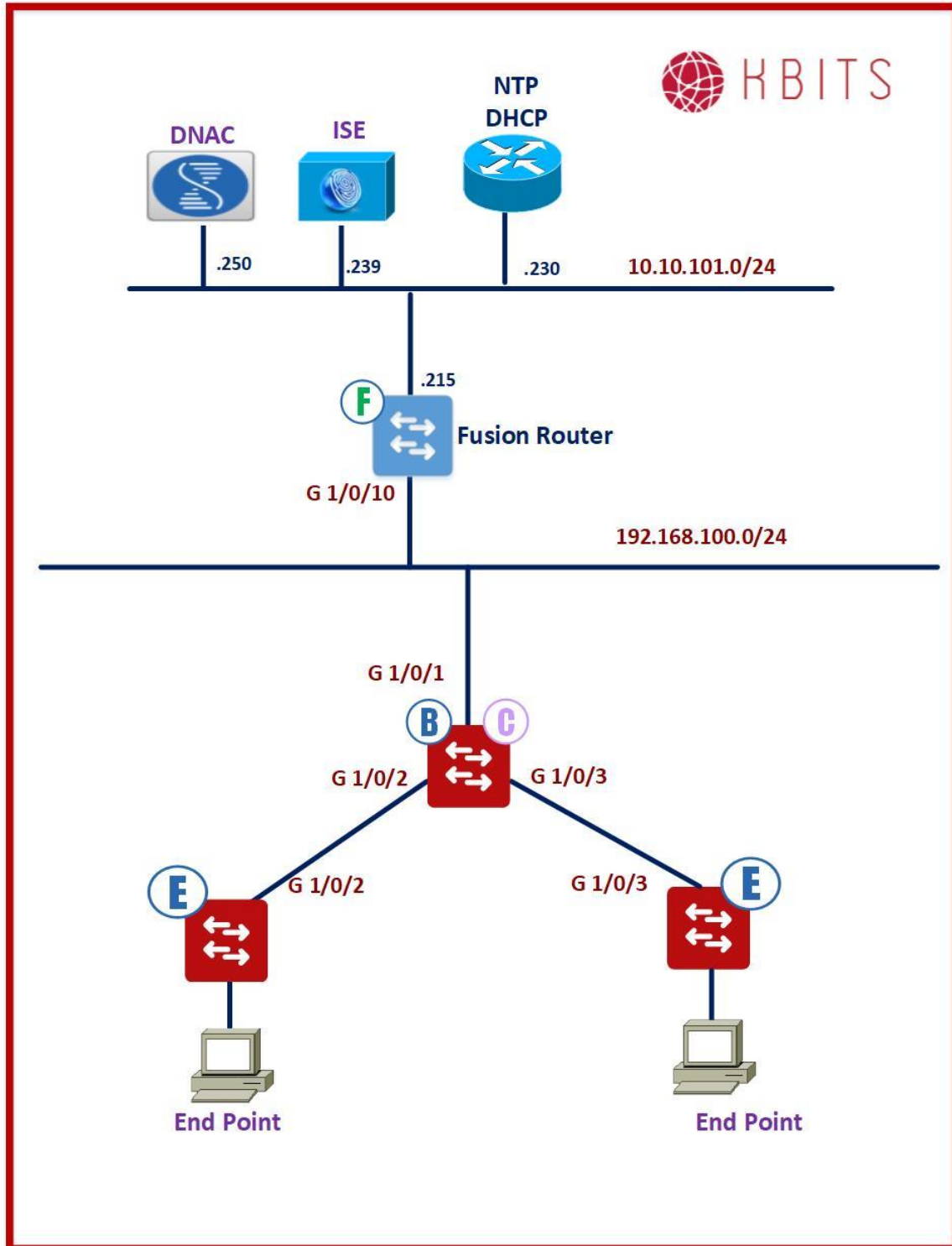


**Task 1 – Re-configure the Authorization Policies to assign the IT\_DATA1 & IT\_DATA2 Groups appropriate SGT in addition to the Authorization profiles**

**Policy -> Policy Sets -> default -> Authorization Policies -> Edit the following Policies**

- Name: **IT-DATA-1-POLICY**
- Identity Group: **IT\_DATA-1**
- Authentication Method: **Wired\_802.1x**
- Permission: **IT-DATA-1-PROF**
- Security Group: **IT\_DATA\_1**
  
- Name: **IT-DATA-2-POLICY**
- Identity Group: **IT\_DATA-2**
- Authentication Method: **Wired\_802.1x**
- Permission: **IT-DATA-2-PROF**
- Security Group: **IT\_DATA\_2**

# Lab 29 - Micro Segmentation – Using Default Contract to Block all communications between SGTs



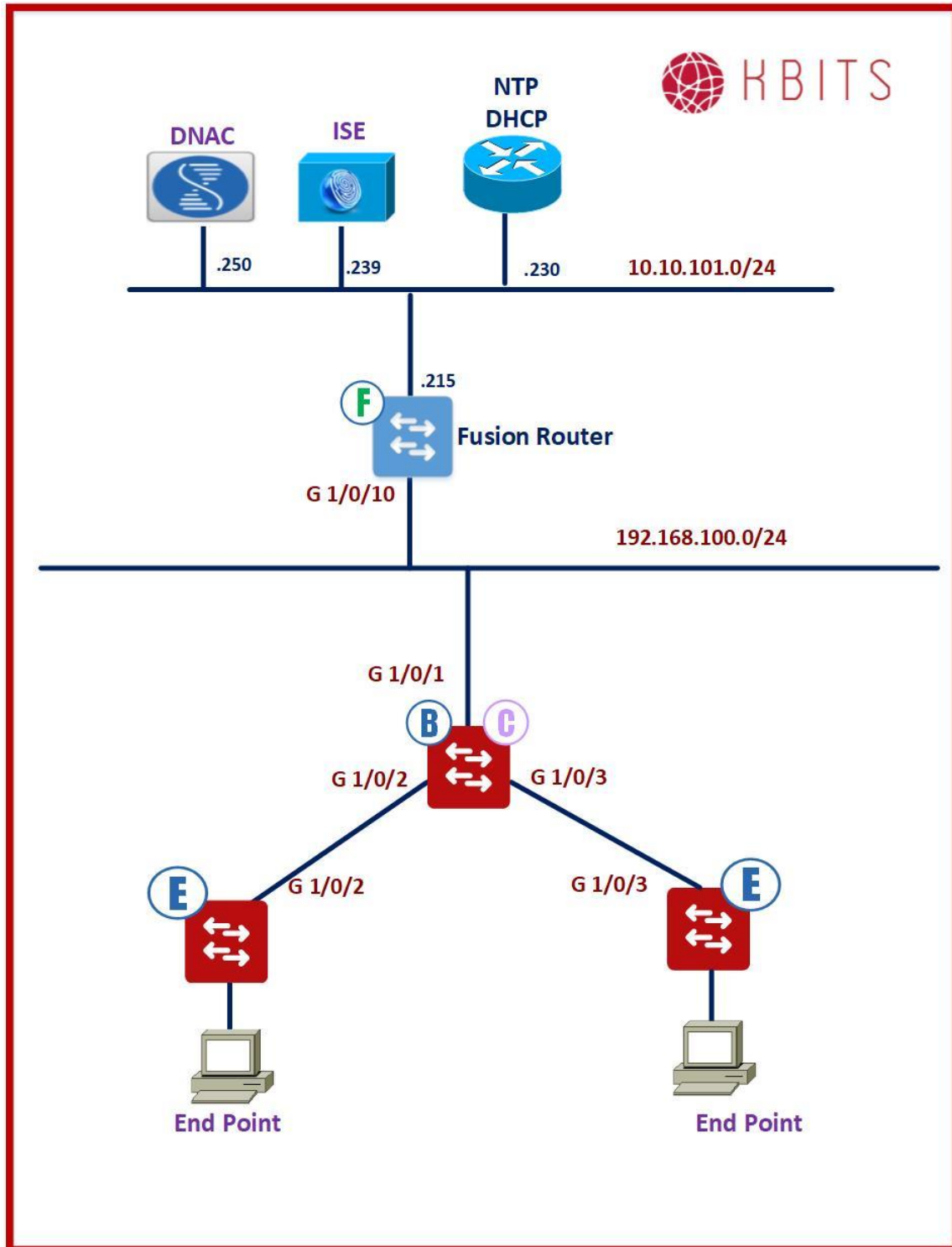
**Task 1 – Configure the SG Access Contract such that all traffic from IT-DATA1 to IT-DATA2 gets denied using the built-in Deny IP contract.**

- Click **Policy -> Group-Based Access Control -> Access Contracts -> Create Access Contract**
- Click on the Policy Matrix box that intersects **IT\_DATA\_1 & IT\_DATA\_2**.
- Click “**Change Contract**”.
- Select “**Deny IP**”.
- Click **Change & Save**. Click **Deploy** to implement the policy.

**Verification:**

- ➔ Configure the Native Windows supplicant on **ES1** to log in using **IT1** credentials.
- ➔ Configure the Native Windows supplicant on **ES2** to log in using **IT2** credentials.
- ➔ Verify that the SG ACL is applied on the Egress Switch (9300E2) using the **show cts role-based-permissions** command.
- ➔ Ping ES2 from ES1. The ping should not work.

# Lab 30 - Micro Segmentation – Creating a SG ACL - Contract



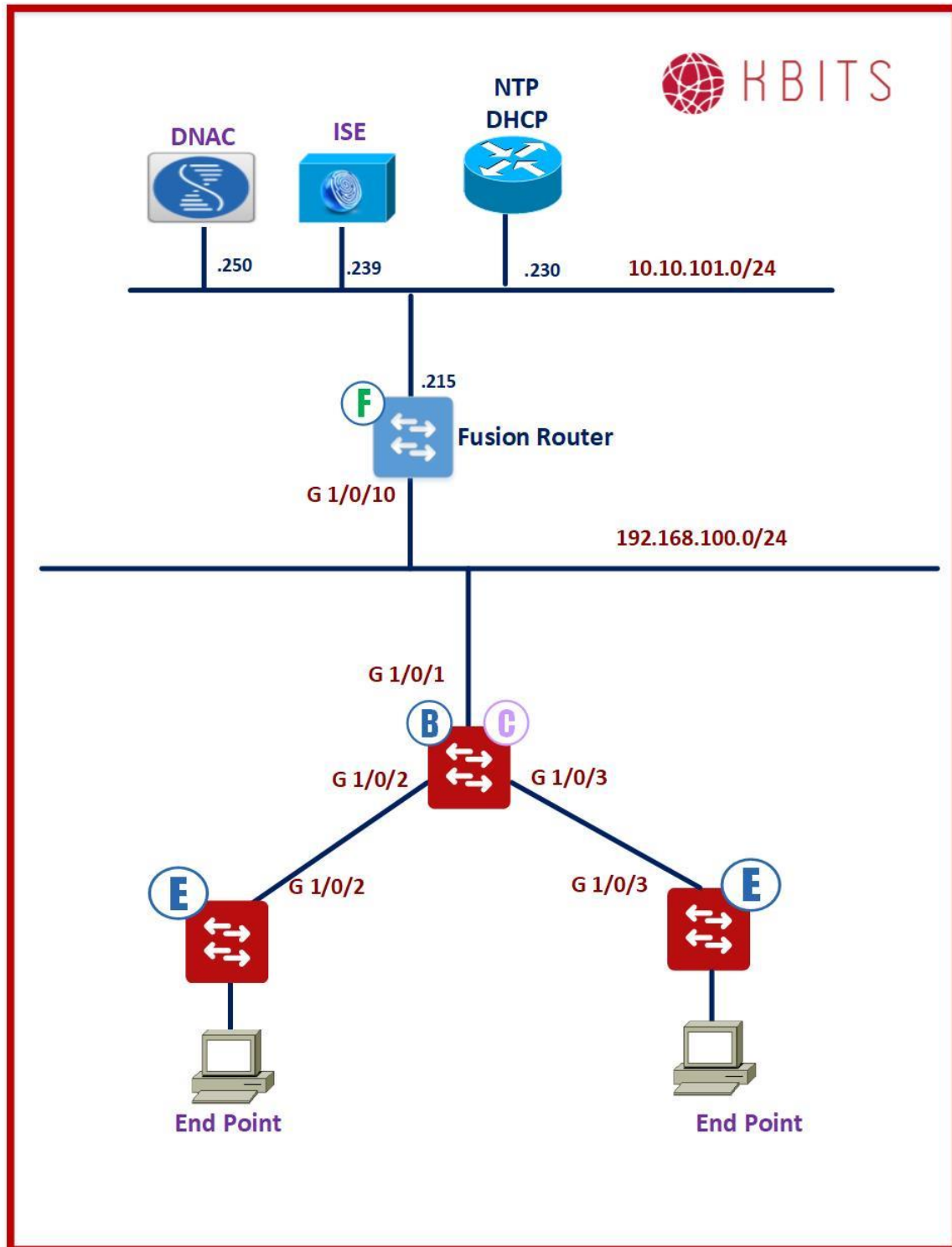
## **Task 1 – Remove the Contract between IT-DATA1 & IT-DATA2**

- Click **Policy** -> **Group-Based Access Control** -> **Policies**
- Click on the Policy Matrix box that intersects **IT\_DATA\_1** & **IT\_DATA\_2**.
- Click “**Set it Default:Permit IP**”
- Click **Change** & **Save**.
- Click **Deploy** to implement the policy.

## **Task 2 – Create a Custom SG ACL (Access Contract) to only allow the following traffic from IT-DATA\_2 to IT-DATA\_1:**

- **Permit - TCP - 80,443**
- **Permit - CIFS**
- Click **Policy** -> **Group-Based Access Control** -> **Access Contracts** -> **Create Access Contracts**
- ➔ **Name: IT\_DATA\_2\_TO\_IT\_DATA\_1**
- ➔ **Rules:**
  - 1. Permit – TCP/80
  - 2. Permit – TCP/443
  - 3. Permit – CIFS
  - Default Action: Deny
- Click **Save**.

# Lab 31 - Micro Segmentation – Applying & Verifying a Custom SG-ACL - Contract



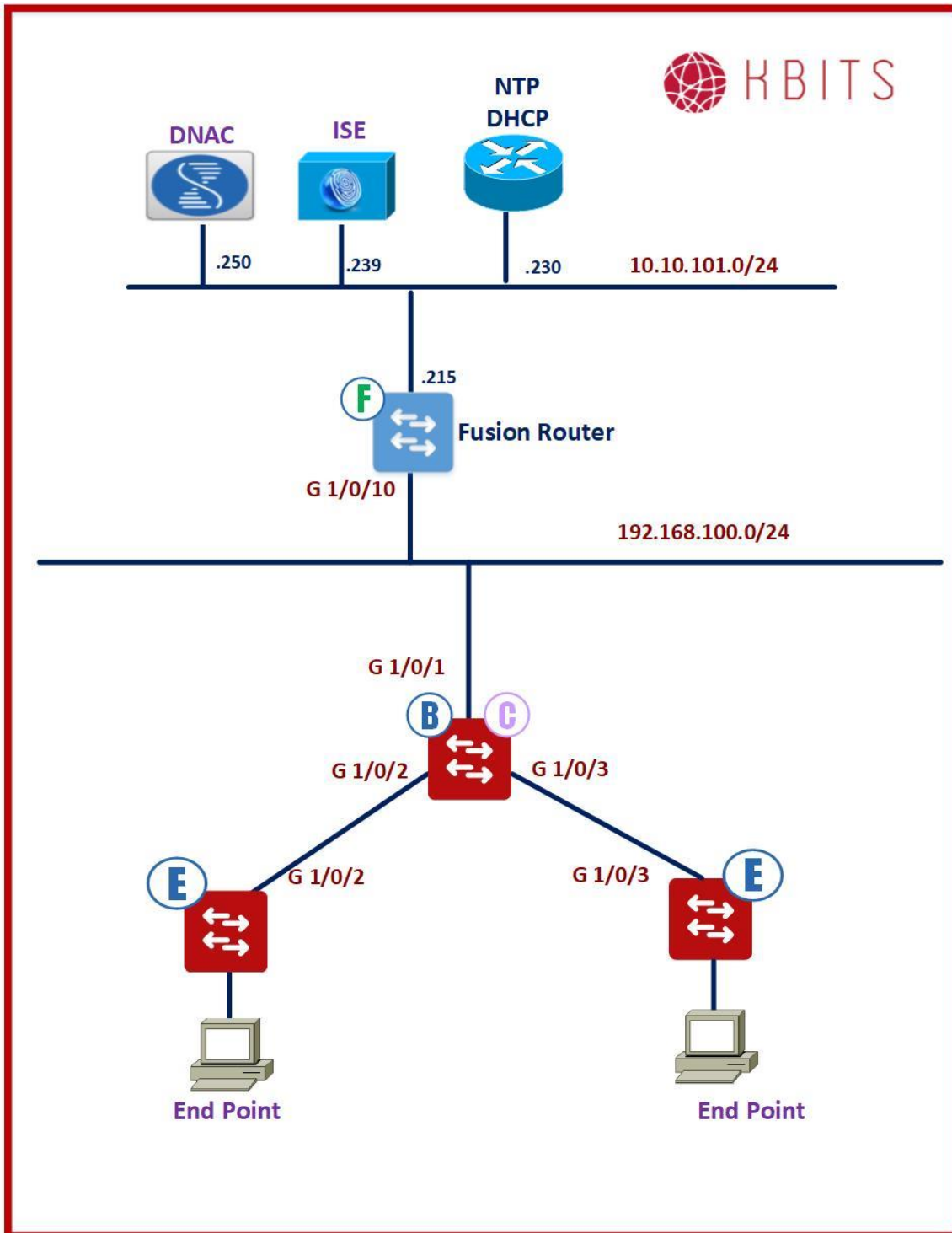
## **Task 1 – Configure the SG Access Contract such that all traffic from IT-DATA1 to IT-DATA2 controlled by the Custom contract created in the previous lab**

- Click **Policy** -> **Group-Based Access Control** -> **Access Contracts** -> **Create Access Contract**
- Click on the Policy Matrix box that intersects **IT\_DATA\_2** & **IT\_DATA\_1**.
- Click “**Change Contract**”.
- Select “**IT\_DATA2\_2\_TO\_IT\_DATA1**”.
- Click **Change** & **Save**. Click **Deploy** to implement the policy.

### **Verification:**

- ➔ Configure the Native Windows supplicant on **ES1** to log in using **IT1** credentials.
- ➔ Configure the Native Windows supplicant on **ES2** to log in using **IT2** credentials.
- ➔ Verify that the SG ACL is applied on the Egress Switch (9300E2) using the **show cts role-based-permissions** command.
- ➔ Ping ES2 from ES1. The ping should not work.
- ➔ Browse to a shared folder on ES2. It should work.

# Lab 32 – Configurig L2 Handoff





## Task 1 – Configure the Existing VN for L2 Flooding

- Click on **Provision -> Fabric -> HQ\_Fabric -> Host Onboarding -> Virtual Networks -> IT\_VN -> Select IT-VN-DATA1 -> Click Action.**
- Click to "**Enable L2 Flooding**"
- Check the "**Common Pool**" Checkbox
- Click **Save** & **Save & Cancel.**

## Task 2 – Configure the Border as VTP Transparent

### Border Switch

Vtp mode transparent

## Task 3 – Configure L2 Handoff

Click **Provision -> Fabric -> HQ\_Fabric -> Fabric Infrastructure -> Click Border -> Click Configure -> Select Layer 2 Hand off -> Click IT\_VN.**

- Select the "**Interface G 1/0/1**" as the L2 Handoff Link.
- Specify the VLAN as **555** for **IT-VN-DATA1**
- Click **Save** and **Add**
- Click **Add**
- Click **Save** and **Apply**

## **Task 4 – Configure L2 Network in the Non-Fabric Devices (Fusion Router)**

- Configure the Fusion Router/Switch with a VLAN **555**.
- Create a SVI Interface for VLAN 555 with an IP Address of 172.16.1.70/24.

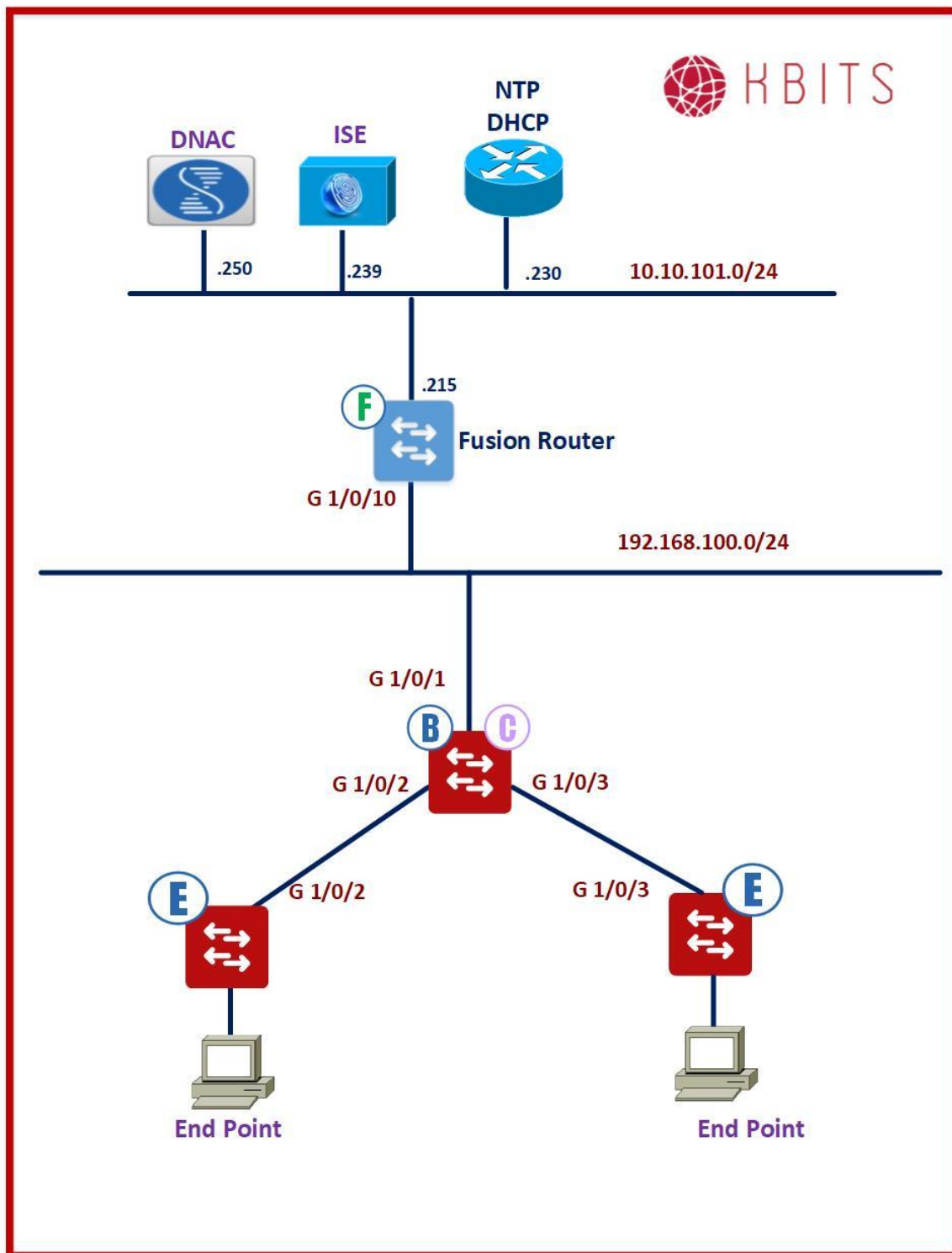
### **Fusion Router**

```
Vlan 555
!  
Interface vlan 555  
Ip add 172.16.1.55 255.255.255.0  
No shut
```

## **Task 5 – Verifying L2Handoff**

- Login from the PC as IT1 (IT-DATA-1).
- You should be able to connect to the Fusion router on 172.16.1.55.

# Lab 33 – Configuring Templates



### **Task 1 – Create a template with the following:**

- Click **Tools** -> **Template Editor** -> “+” -> **Create Template**
- ➔ Name: **Basic**
- ➔ Project: **Cloud DayN Templates**
- ➔ Device Type: **Switches & Hubs**
- ➔ Software Type: **IOS-XE**
- ➔ Click **Save**

### **Task 2 – Configure the parameters within the template:**

- Click **Tools** -> **Template Editor** -> **Cloud DayN Templates** -> **Basic**
- Banner MOTD: **Authorized Users Only**  
**Banner MOTD ^Authorized Users Only^**
- Click **Action** -> **Save**.
- Click **Action** -> **Commit**.

### **Task 3 – Assign the Template**

- Click **Design** -> **Network Profiles** -> **Add** -> **Switching** -> **DayN Template** -> “+” -> **Switches & Hubs** -> **Basic (Under Templates)**
- Profile Name: **BannerProfile**
- Click **Save**.
- Click **Action** -> **Commit**.
- Click **Assign**.
- Assign it across the Global, Los Angeles & HQ.

## **Task 4 – Provision the Template**

- Click **Provision** -> **HQ** -> **Select all the Switches**
- Click **Action** -> **Provision Device** -> **Assign to All Devices.**
- Click **Push these templates even if its deployed before**
- Click **Next**
- Click **Deploy & Apply.**

# IP Services & Security

**Authored By:**

**Khawar Butt**

CCIE # 12353

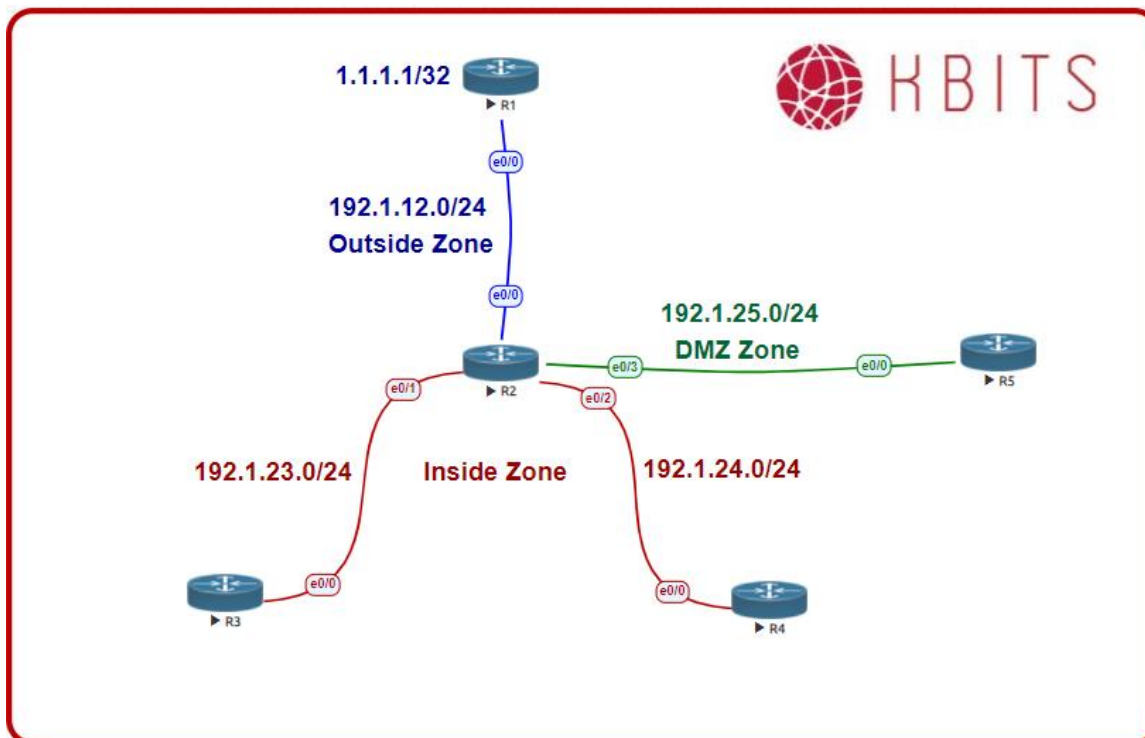
Hepta CCIE#12353

CCDE # 20110020

**IP Services & Security**



# Lab 1 – Zone-Based Firewalls



## Interface IP Address Configuration

### R1

Interface	IP Address	Subnet Mask
Loopback 0	11.1.1.1	255.255.255.255
E 0/0	192.1.12.1	255.255.255.0

### R2

Interface	IP Address	Subnet Mask
E 0/0	192.1.12.2	255.255.255.0
E 0/1	192.1.23.2	255.255.255.0
E 0/2	192.1.24.2	255.255.255.0
E 0/3	192.1.25.2	255.255.255.0

### R3

Interface	IP Address	Subnet Mask
E 0/0	192.1.23.2	255.255.255.0

## R4

Interface	IP Address	Subnet Mask
E 0/0	192.1.24.2	255.255.255.0

## R5

Interface	IP Address	Subnet Mask
E 0/0	192.1.25.2	255.255.255.0

### Task 1

Configure Default Routes on R3, R4 & R5 pointing towards R2. Configure a default route on R2 pointing towards R1. Configure Static Routes on R1 for the 192.1.23.0/24, 192.1.24.0/24 & 192.1.25.0/24 networks with R2 as the next hop.

<b>R1</b> Ip route 192.1.23.0 255.255.255.0 199.1.12.2 Ip route 192.1.24.0 255.255.255.0 199.1.12.2 Ip route 192.1.25.0 255.255.255.0 199.1.12.2	<b>R2</b> Ip route 0.0.0.0 0.0.0.0 192.1.12.1
<b>R3</b> Ip route 0.0.0.0 0.0.0.0 192.1.23.2	<b>R4</b> Ip route 0.0.0.0 0.0.0.0 192.1.24.2
<b>R5</b> Ip route 0.0.0.0 0.0.0.0 192.1.25.2	

### Task 2

Configure R2 as a Zone-Based Firewall. Create the following zones on R2:

- zone security OUTSIDE
- zone security INSIDE
- zone security DMZ

## R2

```
zone security OUTSIDE
zone security INSIDE
zone security DMZ
```



### Task 3

Assign the Interfaces to zones based on the diagram.

#### R2

```
Interface E 0/0
zone-member security OUTSIDE
!
Interface E 0/1
zone-member security INSIDE
Interface E 0/2
zone-member security INSIDE
!
Interface E 0/3
zone-member security DMZ
```

#### Task 4

Configure a Zone-pair policy to allow the following traffic to successfully communicate from the INSIDE zone to the OUTSIDE zone.

- HTTP
- HTTPS
- SMTP
- FTP
- DNS
- TFTP
- Telnet
- SSH
- ICMP

#### R2

```
class-map type inspect match-any CM-I-O
  match protocol http
  match protocol https
  match protocol smtp
  match protocol ftp
  match protocol dns
  match protocol tftp
  match protocol ssh
  match protocol Telnet
  match protocol icmp
!
policy-map type inspect PM-I-O
  class CM-I-O
    inspect
!
zone-pair security I-O source INSIDE destination OUTSIDE
service-policy type inspect PM-I-O
```

#### Task 5

RDP (TCP/3389) should also be allowed to communicate from INSIDE to OUTSIDE.

#### R2

```
Ip port-map user-RDP port tcp 3389
!
class-map type inspect match-any CM-I-O
  match protocol user-rdp
```

## Task 6

Configure a Zone-pair policy to allow the following traffic to successfully communicate from the INSIDE zone to the DMZ zone.

- HTTP
- HTTPS
- SMTP
- DNS
- Telnet
- SSH
- ICMP

## R2

```
class-map type inspect match-any CM-I-D
  match protocol http
  match protocol https
  match protocol smtp
  match protocol dns
  match protocol ssh
  match protocol Telnet
  match protocol icmp
!
policy-map type inspect PM-I-D
  class CM-I-D
    inspect
!
zone-pair security I-D source INSIDE destination DMZ
service-policy type inspect PM-I-D
```

### Task 7

Configure a Zone-pair policy to allow the following traffic to successfully communicate from the OUTSIDE zone to the DMZ zone.

- HTTP – 192.1.25.11
- DNS – 192.1.25.12
- SMTP – 192.1.25.13
- Telnet – 192.1.25.5

### R2

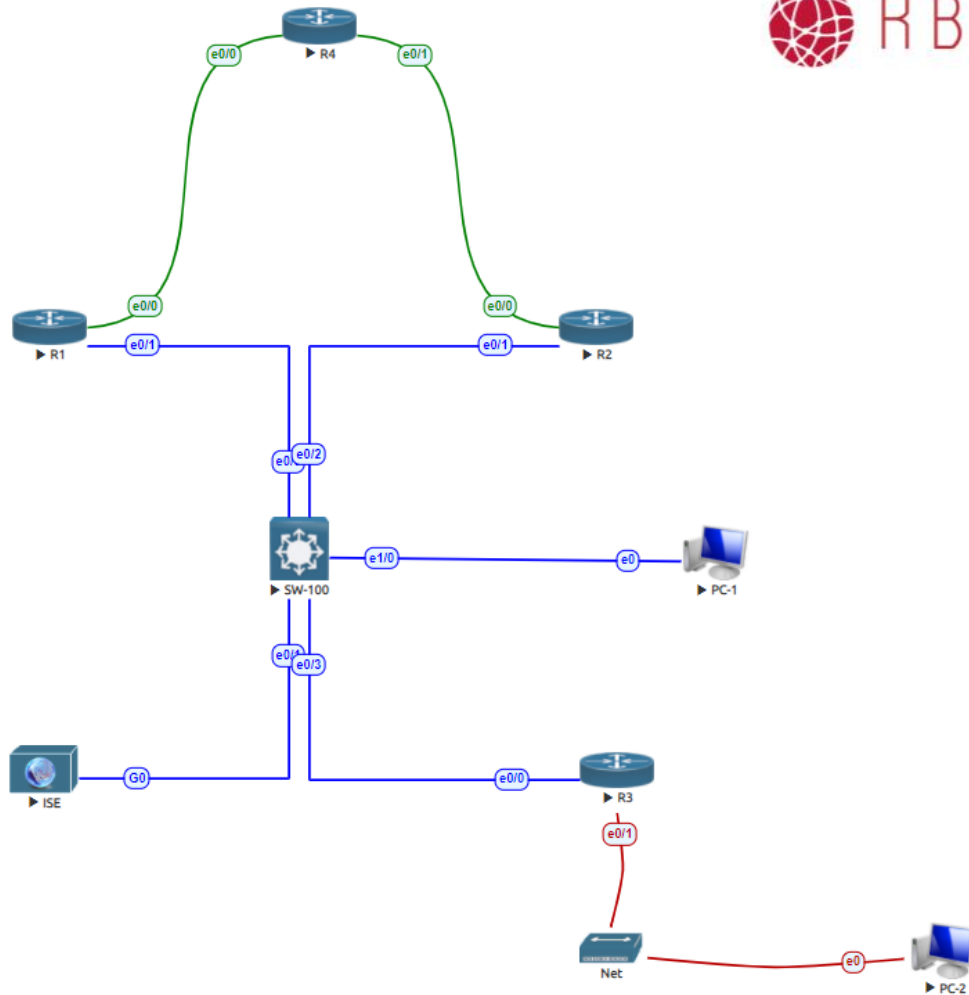
```
access-list 101 permit ip any host 192.1.25.11
access-list 102 permit ip any host 192.1.25.12
access-list 103 permit ip any host 192.1.25.13
access-list 104 permit ip any host 192.1.25.5
!
class-map type inspect match-all CM-O-D-WEB
  match protocol http
  match access-group 101
!
class-map type inspect match-all CM-O-D-DNS
  match protocol dns
  match access-group 102
!
class-map type inspect match-all CM-O-D-MAIL
  match protocol smtp
  match access-group 103
!
class-map type inspect match-all CM-O-D-R5
  match protocol telnet
  match access-group 104

policy-map type inspect PM-O-D
  class CM-O-D-WEB
    inspect
  class CM-O-D-DNS
    inspect
  class CM-O-D-MAIL
    inspect
  class CM-O-D-R5
    inspect

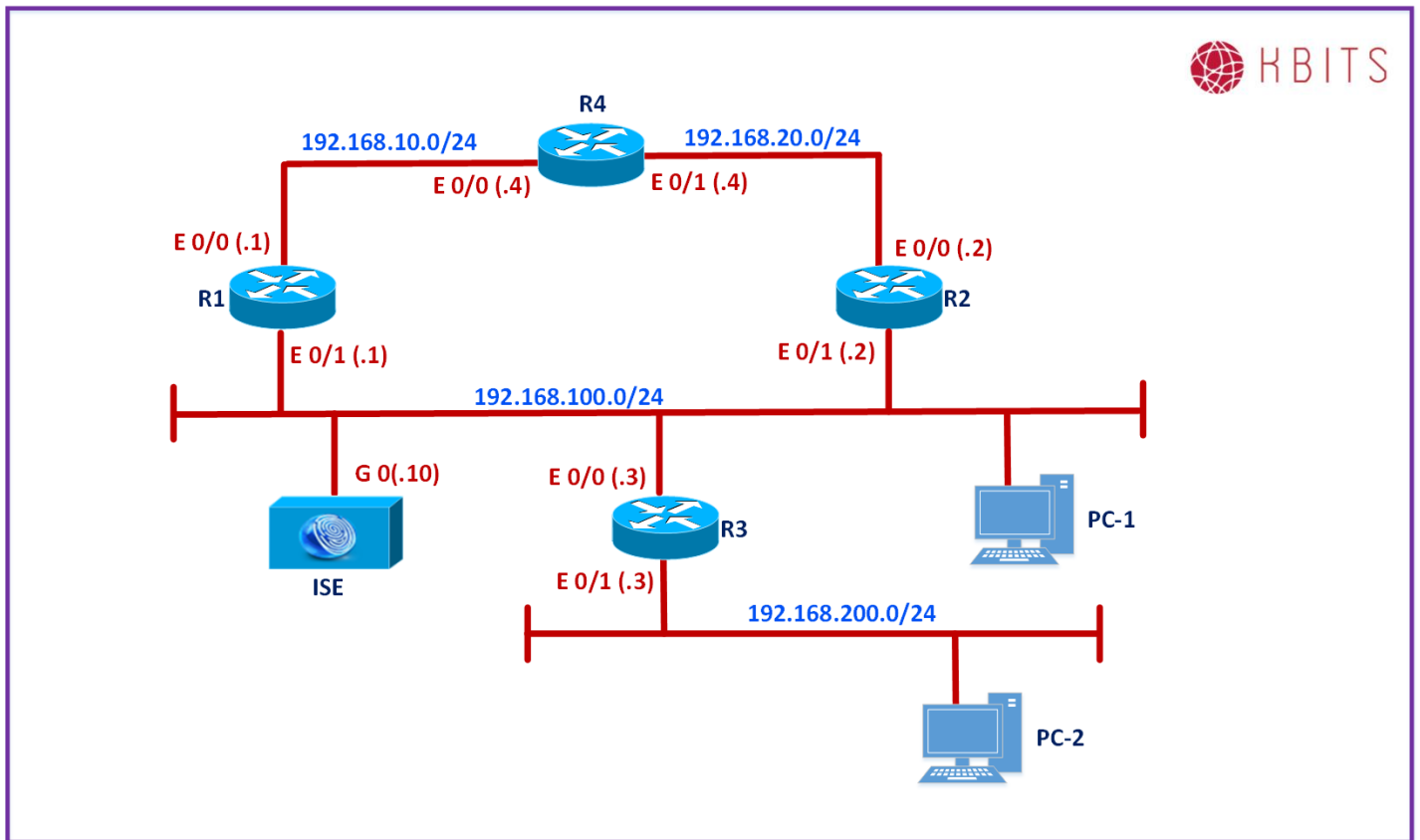
zone-pair security O-D source OUTSIDE destination DMZ
service-policy type inspect PM-O-D
```

# Lab 2 – Configuring FHRP – HSRP

## Physical Topology



## Logical Topology



### Interface IP Address Configuration

#### R1

Interface	IP Address	Subnet Mask
E 0/0	192.168.10.1	255.255.255.0
E 0/1	192.168.100.1	255.255.255.0

#### R2

Interface	IP Address	Subnet Mask
E 0/0	192.168.20.2	255.255.255.0
E 0/1	192.168.100.2	255.255.255.0

### R3

Interface	IP Address	Subnet Mask
E 0/0	192.168.100.3	255.255.255.0
E 0/1	192.168.200.3	255.255.255.0

### R4

Interface	IP Address	Subnet Mask
E 0/0	192.168.10.4	255.255.255.0
E 0/1	192.168.20.4	255.255.255.0
Loopback 0	10.4.4.4	255.255.255.255

### Task 1

Create a VLAN 100 on SW-100. Assign all port on SW-100 to VLAN 100.

### SW1

```
Vlan 100
!  
Interface range E 0/0-3, E1/0-3  
Switchport mode access  
Switchport access vlan 100
```

### Task 2

Configure EIGRP in AS 100 between R1, R2, R3 & R4. Enable EIGRP on all interfaces on all routers.

### R1

```
Router eigrp 100  
Network 192.168.10.0  
Network 192.168.100.0
```

### R2

```
Router eigrp 100  
Network 192.168.20.0  
Network 192.168.100.0
```

### R3

```
Router eigrp 100  
Network 192.168.100.0  
Network 192.168.200.0
```

### R4

```
Router eigrp 100  
Network 192.168.10.0  
Network 192.168.20.0  
Network 10.4.4.0 0.0.0.255
```

### Task 3

Configure Multigroup HSRP between R1 and R2 on the 192.168.100.0 segment. Use the following parameters for Group 1.

- Group ID: 1
- Virtual IP: 192.168.100.254
- Priority: R1 (105); R2 (100 – Default)
- Preemption: Enabled on both
- Authentication: MD5 using a key of kbits@123

#### R1

```
Interface E 0/1
Standby version 2
Standby 1 ip 192.168.100.254
Standby 1 priority 105
Standby 1 preempt
Standby 1 authentication md5 key-string kbits@123
```

#### R2

```
Interface E 0/1
Standby version 2
Standby 1 ip 192.168.100.254
Standby 1 preempt
Standby 1 authentication md5 key-string kbits@123
```

### Task 4

Configure HSRP to track the E 0/0 interface. If it goes down on the active HSRP router for this group, it should decrement the priority by 20 and the other router should become the Active HSRP router.

#### R1

```
track 11 interface ethernet 0/0 line-protocol
!
Interface E 0/1
standby 1 track 11 decrement 20
```

#### R2

```
track 11 interface ethernet 0/0 line-protocol
!
Interface E 0/1
standby 1 track 11 decrement 20
```



### Task 5

Configure HSRP between R1 and R2 on the 192.168.100.0 segment. Use the following parameters for Group 2.

- Group ID: 2
- Virtual IP: 192.168.100.253
- Priority: R2 (105); R1 (100 – Default)
- Preemption: Enabled on both
- Authentication: MD5 using a key of kbits@123

#### R1

```
Interface E 0/1
Standby 2 ip 192.168.100.253
Standby 2 preempt
Standby 2 authentication md5 key-string kbits@123
```

#### R2

```
Interface E 0/1
Standby 2 ip 192.168.100.253
Standby 2 priority 105
Standby 2 preempt
Standby 2 authentication md5 key-string kbits@123
```

### Task 6

HSRP is tracking E 0/0 interface using a Track ID of 11. If E 0/0 goes down on the active HSRP router for this group, it should decrement the priority by 20 and the other router should become the Active HSRP router.

#### R1

```
Interface E 0/1
standby 2 track 11 decrement 20
```

#### R2

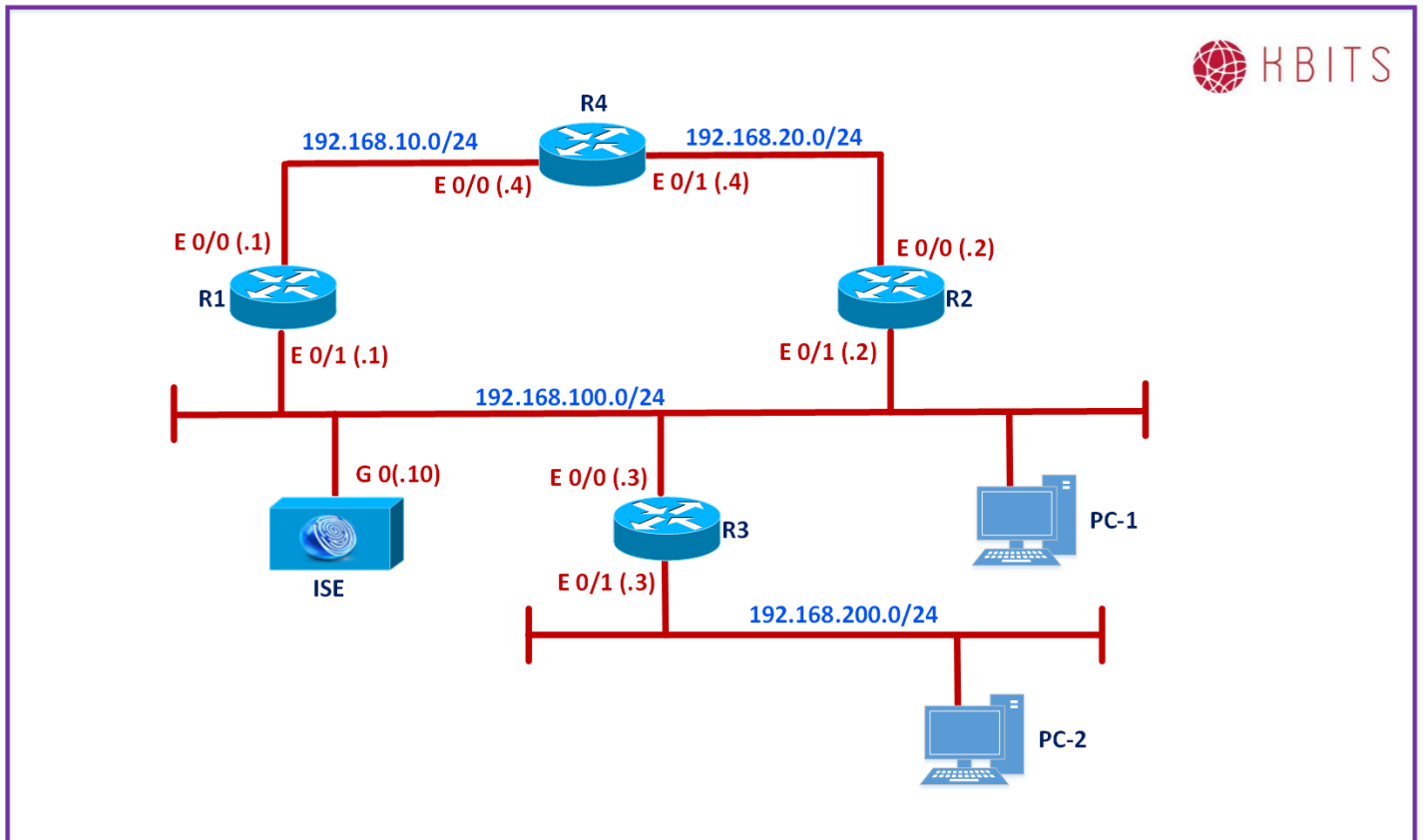
```
Interface E 0/1
standby 2 track 11 decrement 20
```

### Verification:

Verify the HSRP status by issuing the “**Show standby brief**” command on both routers.

# Lab 3 – Configuring FHRP – VRRP

## Logical Topology



### Task 1

De-Configure HSRP on R1 & R2.

R1	R2
Interface E 0/1 No standby 1 No Standby 2 No standby version 2	Interface E 0/1 No standby 1 No Standby 2 No standby version 2

## Task 2

Configure VRRP between R1 and R2 on the 192.168.100.0 segment. Use the following parameters for Group 1.

- Group ID: 1
- Virtual IP: 192.168.100.254
- Priority: R1 (105); R2 (100 – Default)
- Preemption: Enabled on both
- Authentication: MD5 using a key of kbits@123

### R1

```
Interface E 0/1
 vrrp 1 ip 192.168.100.254
 vrrp 1 priority 105
 vrrp 1 authentication md5 key-string kbits@123
```

### R2

```
Interface E 0/1
 vrrp 1 ip 192.168.100.254
 vrrp 1 authentication md5 key-string kbits@123
```

## Task 3

Tracking is enabled for E 0/0 interface based on the previous lab using a Track ID of 11. If E 0/0 goes down on the Master VRRP router for this group, it should decrement the priority by 20 and the other router should become the Master router.

### R1

```
Interface E 0/1
 vrrp 1 track 11 decrement 20
```

### R2

```
Interface E 0/1
 vrrp 1 track 11 decrement 20
```

## Task 4

Configure VRRP between R1 and R2 on the 192.168.100.0 segment. Use the following parameters for Group 2.

- Group ID: 2
- Virtual IP: 192.168.100.253
- Priority: R2 (105); R1 (100-Default)
- Preemption: Enabled on both

- Authentication: MD5 using a key of kbits@123

**R1**

```
Interface E 0/1
 vrrp 2 ip 192.168.100.253
 vrrp 2 authentication md5 key-string kbits@123
```

**R2**

```
Interface E 0/1
 vrrp 2 ip 192.168.100.253
 vrrp 2 priority 105
 vrrp 2 authentication md5 key-string kbits@123
```

**Task 5**

Tracking is enabled for E 0/0 interface based on the previous lab using a Track ID of 11. If E 0/0 goes down on the Master VRRP router for this group, it should decrement the priority by 20 and the other router should become the Master router.

**R1**

```
Interface E 0/1
 vrrp 2 track 11 decrement 20
```

**R2**

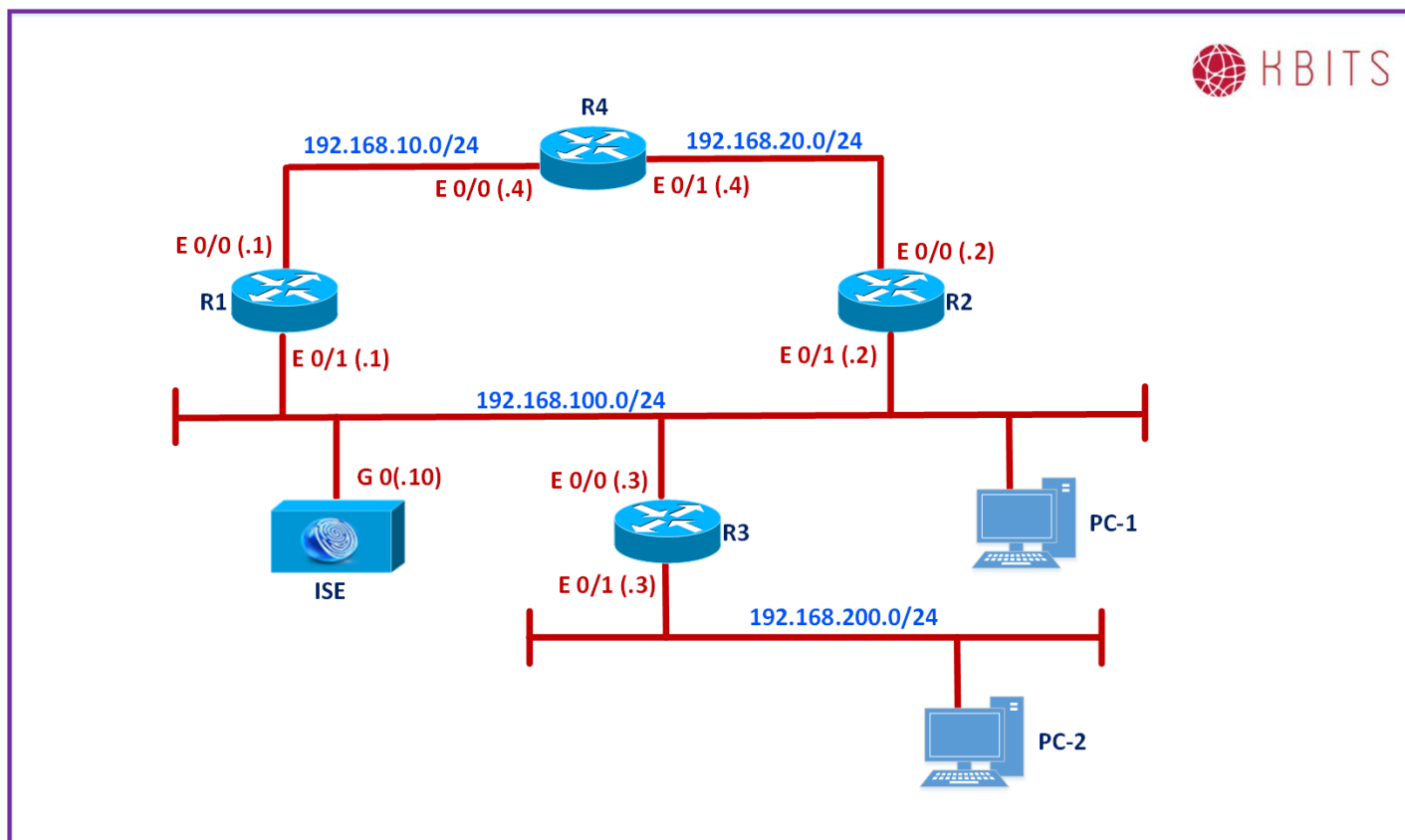
```
Interface E 0/1
 vrrp 2 track 11 decrement 20
```

**Verification:**

Verify the HSRP status by issuing the “**Show standby brief**” command on both routers.

# Lab 4 – Configuring DHCP Server

## Logical Topology



### Task 1

Configure R1 as the DHCP Server a scope for network 192.168.100.0/24 using the following parameters:

- Excluded Addresses:
  - 192.168.100.1 – 192.168.100.20
  - 192.168.100.251 – 192.168.100.254
- Default Gateway: 192.168.100.253
- DNS Server: 192.168.100.2

### R1

```
Ip dhcp excluded-address 192.168.100.1 192.168.100.20
Ip dhcp excluded-address 192.168.100.251 192.168.100.254
!
Ip dhcp pool POOL-100
Network 192.168.100.0 /24
```

```
Default-router 192.168.100.253  
Dns-server 192.168.100.2
```

### **Task 2**

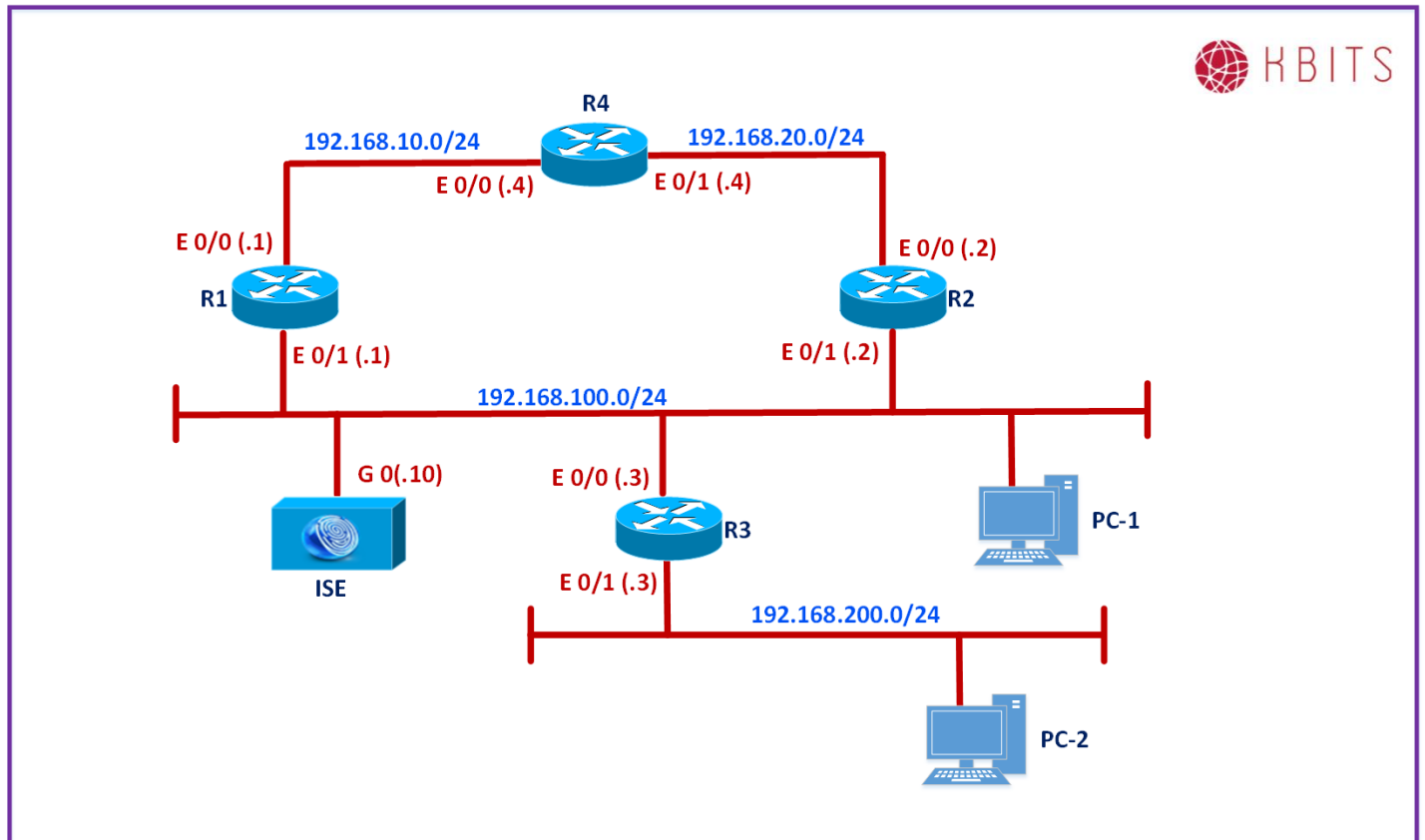
Configure PC-1 to receive an IP Address automatically (DHCP).

### **Verification:**

Verify the configuration on the PC using the “**IPConfig**” command. Also, use the “**Tracert 10.4.4.4**” command to check the flow of traffic. It should use R2 as it should be the Master router for Standby group 2 (**Default Router: 192.168.100.253**)

# Lab 5 – Configuring DHCP Relay Agent

## Logical Topology



### Task 1

Configure R1 as the DHCP Server a scope for network 192.168.200.0/24 using the following parameters:

- Excluded Addresses:
  - 192.168.200.1 – 192.168.200.20
- Default Gateway: 192.168.200.3
- DNS Server: 192.168.100.2

### R1

```
Ip dhcp excluded-address 192.168.200.1 192.168.200.20
!  
Ip dhcp pool POOL-200  
Network 192.168.200.0 /24  
Default-router 192.168.200.3  
Dns-server 192.168.100.2
```

## Task 2

Configure R3 as a DHCP Relay Agent. It should forward DHCP broadcast requests towards R1. Make sure to only forward DHCP Server and Client Broadcasts towards R1.

### R3

```
No Ip forward-protocol udp 37
No Ip forward-protocol udp 49
No Ip forward-protocol udp 53
No Ip forward-protocol udp 69
No Ip forward-protocol udp 137
No Ip forward-protocol udp 138
!
Interface E 0/1
Ip helper-address 192.168.100.1
```

## Task 3

Configure PC-2 to receive an IP Address automatically (DHCP).

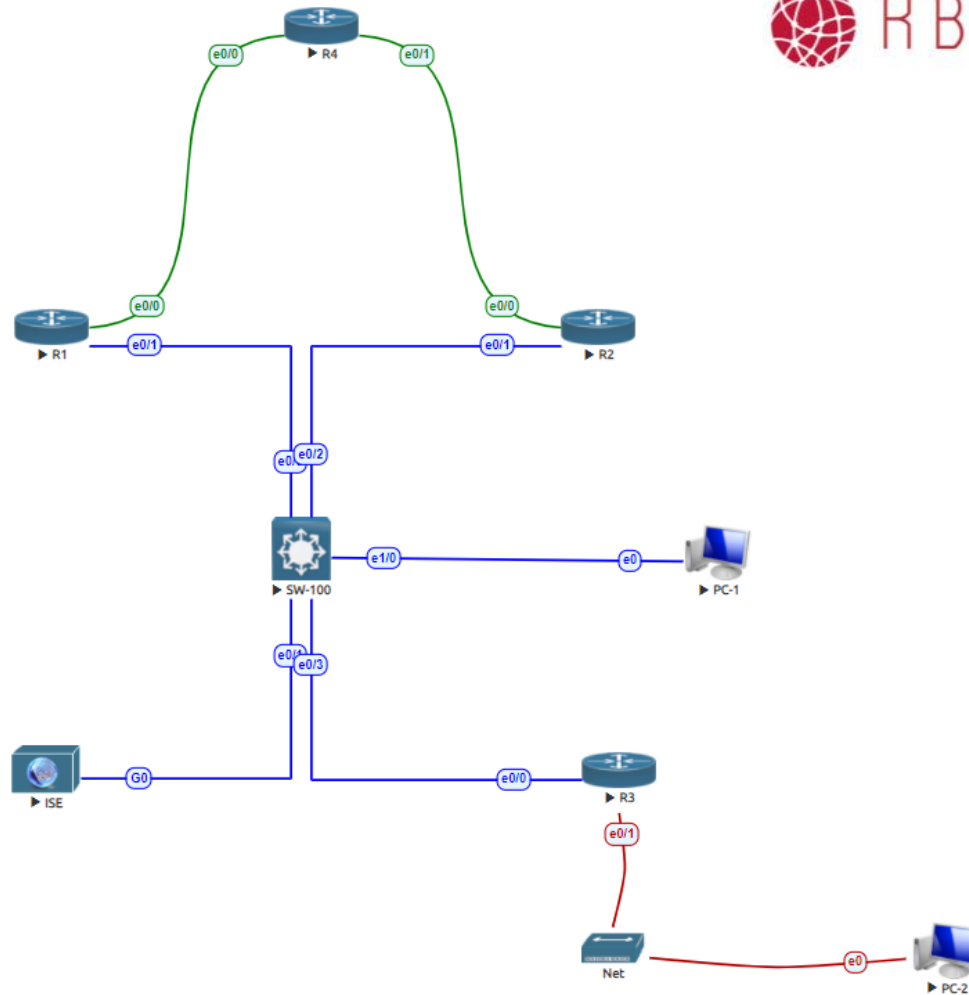
### Verification:

Verify the configuration on the PC using the “**IPConfig**” command. Also, use the “**Ping 10.4.4.4**” command to check connectivity towards R4.

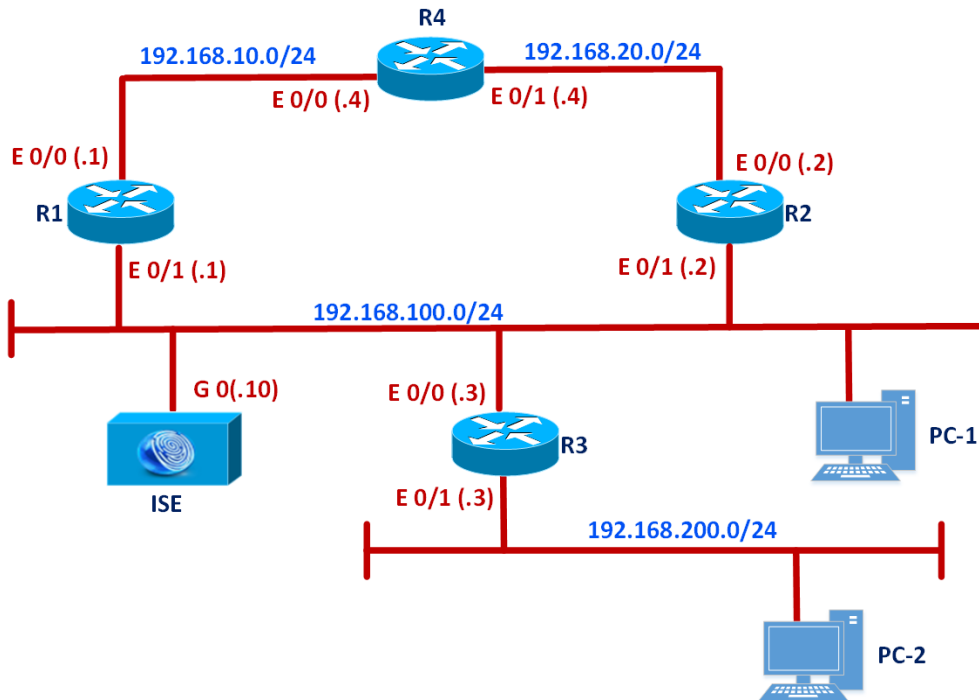


# Lab 6 – Configuring DHCP Snooping

## Physical Topology



## Logical Topology



### Task 1

R1 is the only DHCP Server in the environment. It does not support Option-82. Configure SW-100 such that it only allows DHCP replies from R1 in VLAN 100.

#### SW-100

```
Ip dhcp snooping
Ip dhcp snooping vlan 100
no ip dhcp snooping information option
!
Interface E 0/0
Ip dhcp snooping trust
```

### Task 2

Configure the port on SW-100 connected towards R3 E0/0 also as a trusted port as it is acting as a DHCP Relay Agent.

#### SW-100

```
Interface E 0/3
Ip dhcp snooping trust
```

**Task 3**

Use the “**IPConfig /release**” command to release the assigned address on PC-1 & PC-2.

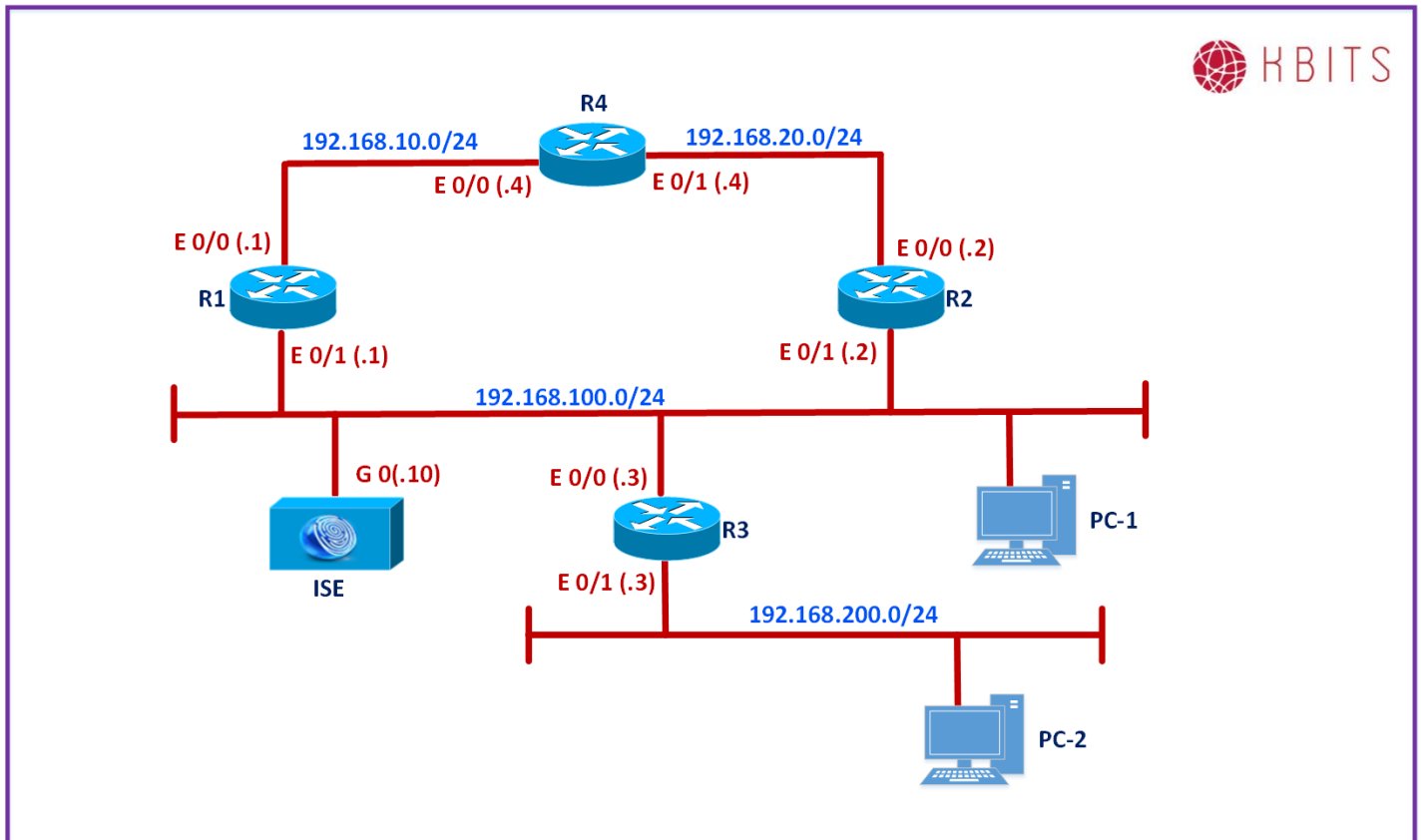
Use the “**IPConfig /renew**” command to renew the Address release.

**Verification:**

You should be assigned addresses from the DHCP Server.

# Lab 7 – Configuring NTP

## Logical Topology



### Task 1

R4 is in Dubai. Dubai is 4 hours ahead of GMT. Configure the Timezone on R4. Set the time based on Dubai. Configure R4 as the NTP Master with a stratum of 4. It should use Loopback0 as the NTP Source.

### R4

```
Clock timezone GST 4
Do clock set 9:30:00 4 Mar 2021
!
Ntp master 4
Ntp source Loopback0
```

## Task 2

Configure R1 & R2 are in New York. Configure them with a timezone with an offset of -5. They should point to R4 Loopback0 as the NTP Server.

### R1

```
Clock timezone EST -5  
!  
NTP Server 10.4.4.4
```

### R2

```
Clock timezone EST -5  
!  
NTP Server 10.4.4.4
```

## Task 3

Authenticate the NTP Associations between the routers using a Key of 11 and key-string of kbits@123.

### R4

```
Ntp authenticate  
Ntp authentication-key 11 md5 kbits@123  
Ntp trusted-key 11
```

### R1

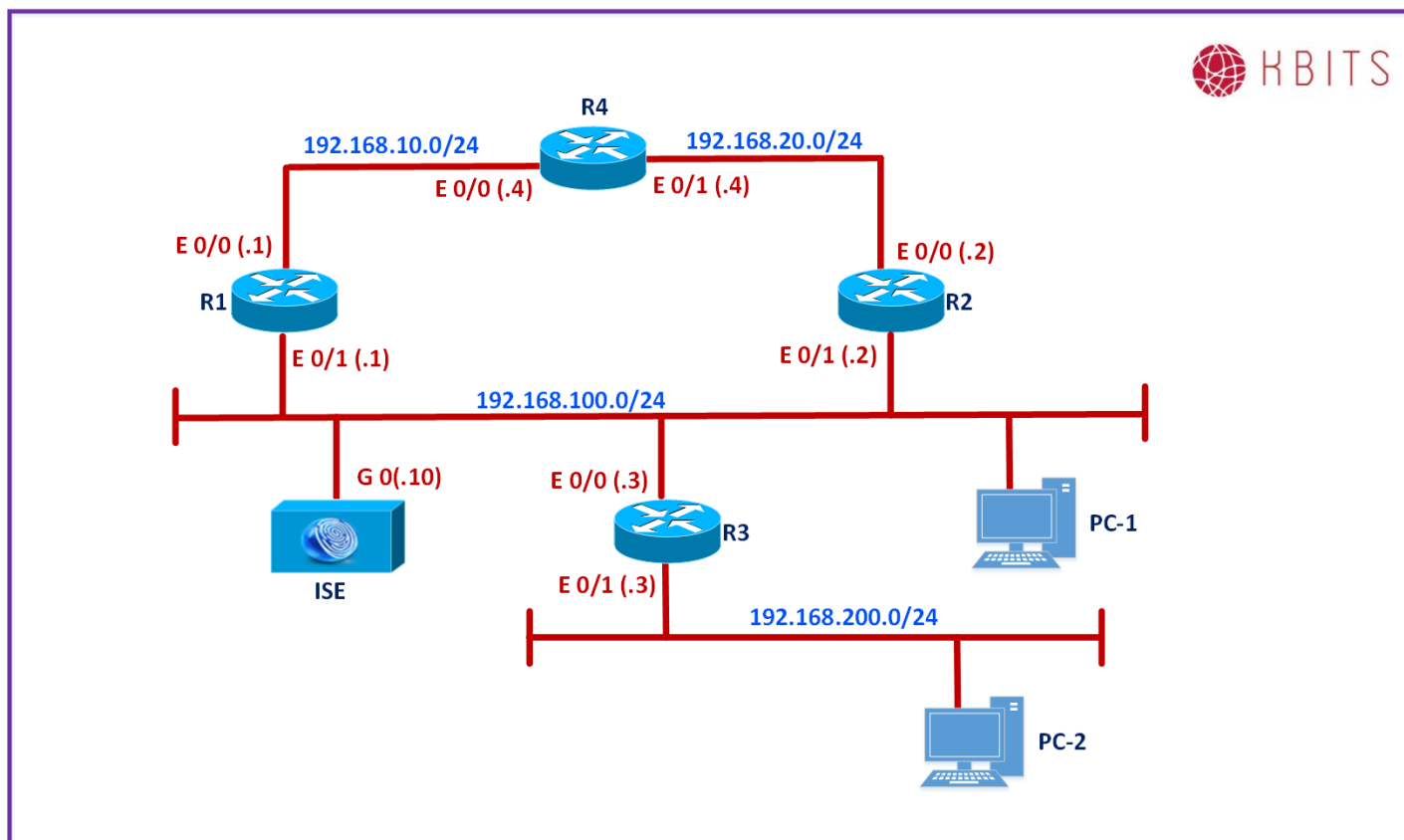
```
Ntp authenticate  
Ntp authentication-key 11 md5 kbits@123  
Ntp trusted-key 11  
Ntp server 10.4.4.4 key 11
```

### R2

```
Ntp authenticate  
Ntp authentication-key 11 md5 kbits@123  
Ntp trusted-key 11  
Ntp server 10.4.4.4 key 11
```

# Lab 8 – Configuring AAA Services

## Logical Topology



### Task 1

Configure R1, R2, R3, and R4 to communicate with ISE for AAA Services. R4 should use the Loopback 0 interface to communicate to ISE. All the devices should use a secret key of kbits@123.

#### R1

```
Aaa new-model
!  
Tacacs server ISE  
Address ipv4 192.168.100.10  
key kbits@123
```

#### R2

```
Aaa new-model
!  
Tacacs server ISE
```

```
Address ipv4 192.168.100.10
key kbits@123
```

### **R3**

```
Aaa new-model
!
Tacacs server ISE
Address ipv4 192.168.100.10
key kbits@123
```

### **R4**

```
Aaa new-model
!
Tacacs server ISE
Address ipv4 192.168.100.10
key kbits@123
!
Ip tacacs source-interface Loopback0
```

## **Task 2**

Create a username **admin1** with a password of **admin1** in the local database. Assign it a privilege level of 15. This needs to be configured on all the routers.

### **R1**

```
Username admin1 privilege 15 password admin1
```

### **R2**

```
Username admin1 privilege 15 password admin1
```

### **R3**

```
Username admin1 privilege 15 password admin1
```

### **R4**

```
Username admin1 privilege 15 password admin1
```

### Task 3

Configure a TACACS Group called ISE-SVRS. Assign ISE to this group. This needs to be configured on all the routers.

#### R1

```
aaa group server tacacs+ ISE-SVRS
server name ISE1
```

#### R2

```
aaa group server tacacs+ ISE-SVRS
server name ISE1
```

#### R3

```
aaa group server tacacs+ ISE-SVRS
server name ISE1
```

#### R4

```
aaa group server tacacs+ ISE-SVRS
server name ISE1
```

### Task 4

Configure all routers to use ISE-SVRS for login authentication. Use a Named-list called T-AUTHEN. T-AUTHEN should use ISE-SVRS as the primary authentication and Local Database for fallback authentication. Enable Telnet & SSH on the Routers and have them use T-AUTHEN for authentication.

#### R1

```
Aaa authentication login T-AUTHEN group ISE-SVRS local
!
Line vty 0 4
Transport input telnet ssh
Login authentication T-AUTHEN
```

#### R2

```
Aaa authentication login T-AUTHEN group ISE-SVRS local
!
Line vty 0 4
Transport input telnet ssh
Login authentication T-AUTHEN
```

#### R3

```
Aaa authentication login T-AUTHEN group ISE-SVRS local
```



```
!  
Line vty 0 4  
Transport input telnet ssh  
Login authentication T-AUTHEN
```

#### **R4**

```
Aaa authentication login T-AUTHEN group ISE-SVRS local  
!  
Line vty 0 4  
Transport input telnet ssh  
Login authentication T-AUTHEN
```

#### **Task 5**

Configure all routers to use the ISE-SVRS for Exec authorization. Use a named list called T-AUTHOR. T-AUTHOR should use ISE-SVRS as the primary exec authorization and Local Database for fallback authorization. Have Telnet & SSH use T-AUTHOR for authorization.

#### **R1**

```
Aaa authorization exec T-AUTHOR group ISE-SVRS local  
!  
Line vty 0 4  
Authorization exec T-AUTHOR
```

#### **R2**

```
Aaa authorization exec T-AUTHOR group ISE-SVRS local  
!  
Line vty 0 4  
Authorization exec T-AUTHOR
```

#### **R3**

```
Aaa authorization exec T-AUTHOR group ISE-SVRS local  
!  
Line vty 0 4  
Authorization exec T-AUTHOR
```

#### **R4**

```
Aaa authorization exec T-AUTHOR group ISE-SVRS local  
!  
Line vty 0 4  
Authorization exec T-AUTHOR
```

## Task 6

Configure all routers to use the ISE-SVRS for Exec & Command level 15 accounting. Use a named list called T-ACCT. T-ACCT should use ISE-SVRS for both Exec & Command Level 15 accounting. Have Telnet & SSH use T-AUTHOR for accounting.

### R1

```
Aaa accounting exec T-ACCT start-stop group ISE-SVRS
Aaa accounting command 15 T-ACCT start-stop group ISE-SVRS
```

```
!
Line vty 0 4
  Accounting exec T-ACCT
  Accounting command 15 T-ACCT
```

### R2

```
Aaa accounting exec T-ACCT start-stop group ISE-SVRS
Aaa accounting command 15 T-ACCT start-stop group ISE-SVRS
```

```
!
Line vty 0 4
  Accounting exec T-ACCT
  Accounting command 15 T-ACCT
```

### R3

```
Aaa accounting exec T-ACCT start-stop group ISE-SVRS
Aaa accounting command 15 T-ACCT start-stop group ISE-SVRS
```

```
!
Line vty 0 4
  Accounting exec T-ACCT
  Accounting command 15 T-ACCT
```

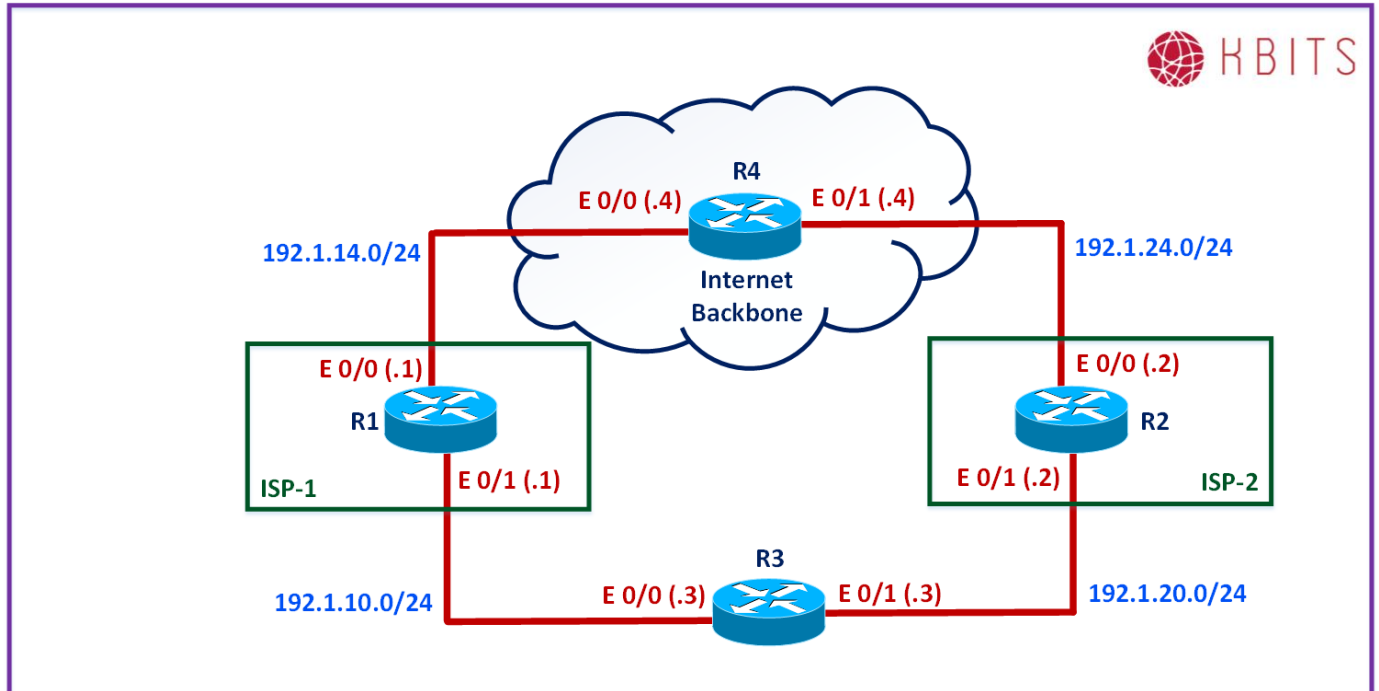
### R4

```
Aaa accounting exec T-ACCT start-stop group ISE-SVRS
Aaa accounting command 15 T-ACCT start-stop group ISE-SVRS
```

```
!
Line vty 0 4
  Accounting exec T-ACCT
  Accounting command 15 T-ACCT
```

# Lab 9 – Configuring IP SLA

## Logical Topology



### Interface IP Address Configuration

#### R1

Interface	IP Address	Subnet Mask
E 0/0	192.1.14.1	255.255.255.0
E 0/1	192.1.10.1	255.255.255.0

#### R2

Interface	IP Address	Subnet Mask
E 0/0	192.1.24.2	255.255.255.0
E 0/1	192.1.20.2	255.255.255.0

### R3

Interface	IP Address	Subnet Mask
E 0/0	192.1.10.3	255.255.255.0
E 0/1	192.1.20.3	255.255.255.0

### R4

Interface	IP Address	Subnet Mask
E 0/0	192.1.14.4	255.255.255.0
E 0/1	192.1.24.4	255.255.255.0
Loopback 0	4.2.2.2	255.255.255.255
Loopback 199	199.1.1.1	255.255.255.0

#### Task 1

Configure EIGRP in AS 111 between R1, R2 & R4. Enable all interfaces on all 3 routers in EIGRP. Configure the E 0/1 interfaces on R1 & R2 as passive-interfaces.

<b>R1</b>  Router eigrp 111 Network 192.1.10.0 Network 192.1.14.0 Passive-interface E 0/1	<b>R2</b>  Router eigrp 111 Network 192.1.20.0 Network 192.1.24.0 Passive-interface E 0/1
<b>R4</b>  Router eigrp 111 Network 192.1.14.0 Network 192.1.24.0 Network 4.0.0.0 Network 199.1.1.0	

#### Task 2

Configure and enable an SLA object on R3 with the following parameters:

- SLA Object #: 33
- Destination IP: 4.2.2.2
- Source-IP: 192.1.10.1
- Protocol: ICMP Echo
- Frequency: 20

Create a host route for 4.2.2.2 via R1.

<b>R3</b>
-----------

```
Ip route 4.2.2.2 255.255.255.255 192.1.10.1
!  
ip sla 33  
  icmp-echo 4.2.2.2 source-ip 192.1.10.3  
  frequency 20  
!  
ip sla schedule 33 start-time now life forever
```

### Task 3

Configure a Track object 33. It have a state of **“UP”** based on the state of SLA 33.

### R3

```
track 33 ip sla 33 state
```

### Task 4

Configure floating default static routes via R1 or R2. R1 should be used as the preferred default route if track object 33 has a state of **“UP”**.

### R3

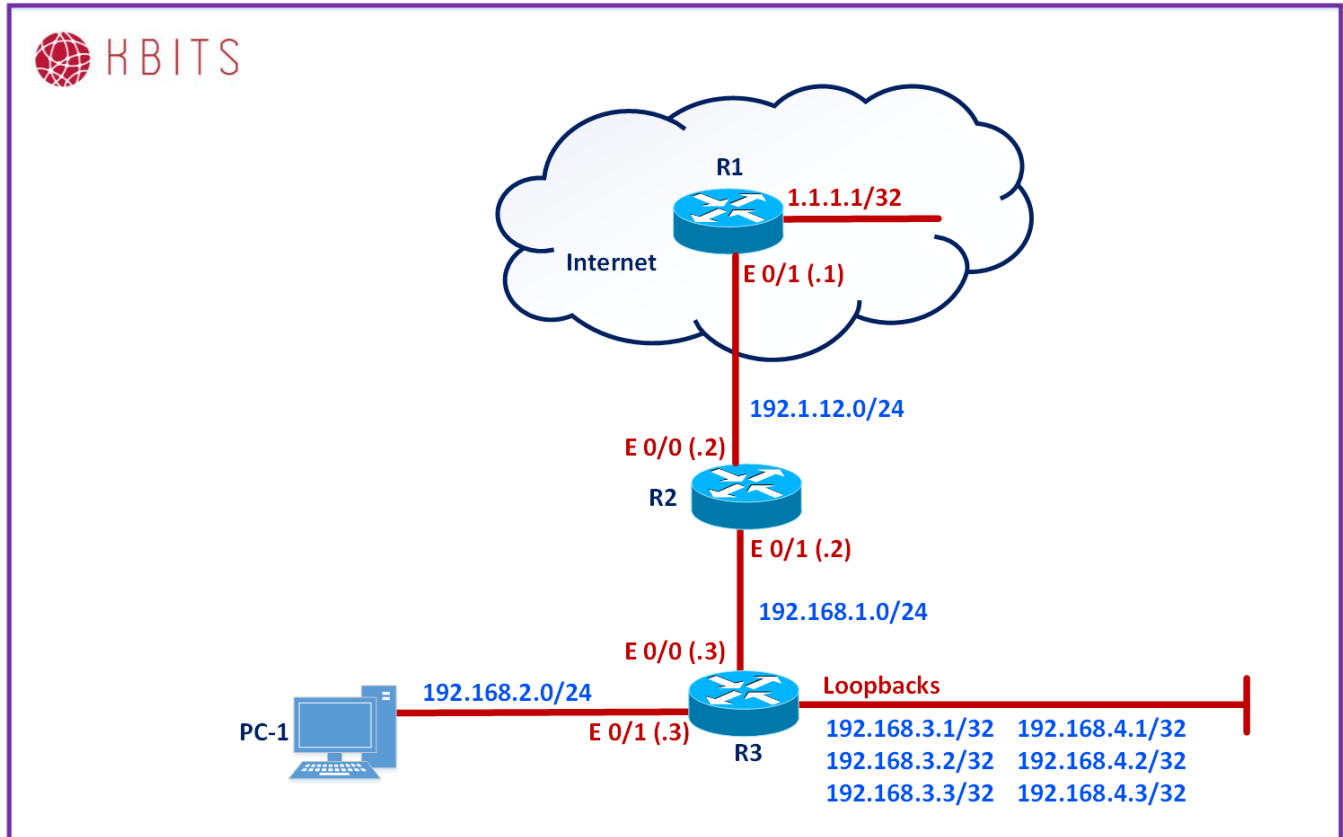
```
Ip route 0.0.0.0 0.0.0.0 192.1.10.1 track 33  
Ip route 0.0.0.0 0.0.0.0 192.1.20.2 5
```

### Verification:

- Verify that the Default Route installed in the routing table is via R1. Verify the connectivity by pinging 199.1.1.1.
- Shut the E0/0 Interface on R1.
- Check the Track state on R3.
- Verify that the Default Route installed in the routing table is via R2. Verify the connectivity by pinging 199.1.1.1.
- Bring up the E0/0 Interface on R1.
- Check the Track state on R3.
- Verify that the Default Route installed in the routing table is via R1. Verify the connectivity by pinging 199.1.1.1.

# Lab 10 – Configuring Dynamic NAT

## Logical Topology



### Interface IP Address Configuration

#### R1

Interface	IP Address	Subnet Mask
E 0/0	192.1.12.1	255.255.255.0
Loopback0	1.1.1.1	255.255.255.255

#### R2

Interface	IP Address	Subnet Mask
E 0/0	192.1.12.2	255.255.255.0
E 0/1	192.168.1.2	255.255.255.0

### R3

Interface	IP Address	Subnet Mask
E 0/0	192.168.1.3	255.255.255.0
E 0/1	192.168.2.3	255.255.255.0
Loopback0	192.168.3.1	255.255.255.255
Loopback1	192.168.3.2	255.255.255.255
Loopback2	192.168.3.3	255.255.255.255
Loopback3	192.168.4.1	255.255.255.255
Loopback4	192.168.4.2	255.255.255.255
Loopback5	192.168.4.3	255.255.255.255

#### Task 1

Configure a Default routes on R2 pointing towards R1. Configure a Default routes on R3 pointing towards R2.

#### R2

```
Ip route 0.0.0.0 0.0.0.0 192.1.12.1
```

#### R3

```
Ip route 0.0.0.0 0.0.0.0 192.168.1.2
```

#### Task 2

Configure Routing for Internal networks between R2 & R3. Use EIGRP in AS 111 as the routing protocol.

#### R2

```
Router eigrp 111  
Network 192.168.1.0
```

#### R3

```
Router eigrp 111  
Network 192.168.1.0  
Network 192.168.2.0  
Network 192.168.3.0  
Network 192.168.4.0
```

### Task 3

Configure a pool of 192.1.10.51 thru 192.1.10.254 on R2. The pool will be used by the 192.168.2.0/24 network. Use POOL1 as the name of the pool. Configure an ACL to classify the outbound traffic. Create a Static Route on R1 for the 192.1.10.0/24 network pointing towards R2.

#### R1

```
Ip route 192.1.10.0 255.255.255.0 192.1.12.2
```

#### R2

```
Ip nat pool POOL1 192.1.10.51 192.1.10.254 netmask 255.255.255.0  
!  
Access-list 101 permit ip 192.168.2.0 0.0.0.255 any
```

### Task 4

Configure R2 for NAT. Enable the E 0/0 interface as the Outside interface for NAT. Enable the E 0/1 interface as the Inside interface for NAT.

#### R2

```
Ip nat inside source list 101 pool POOL1  
!  
Interface E 0/0  
Ip nat outside  
!  
Interface E 0/1  
Ip nat inside
```

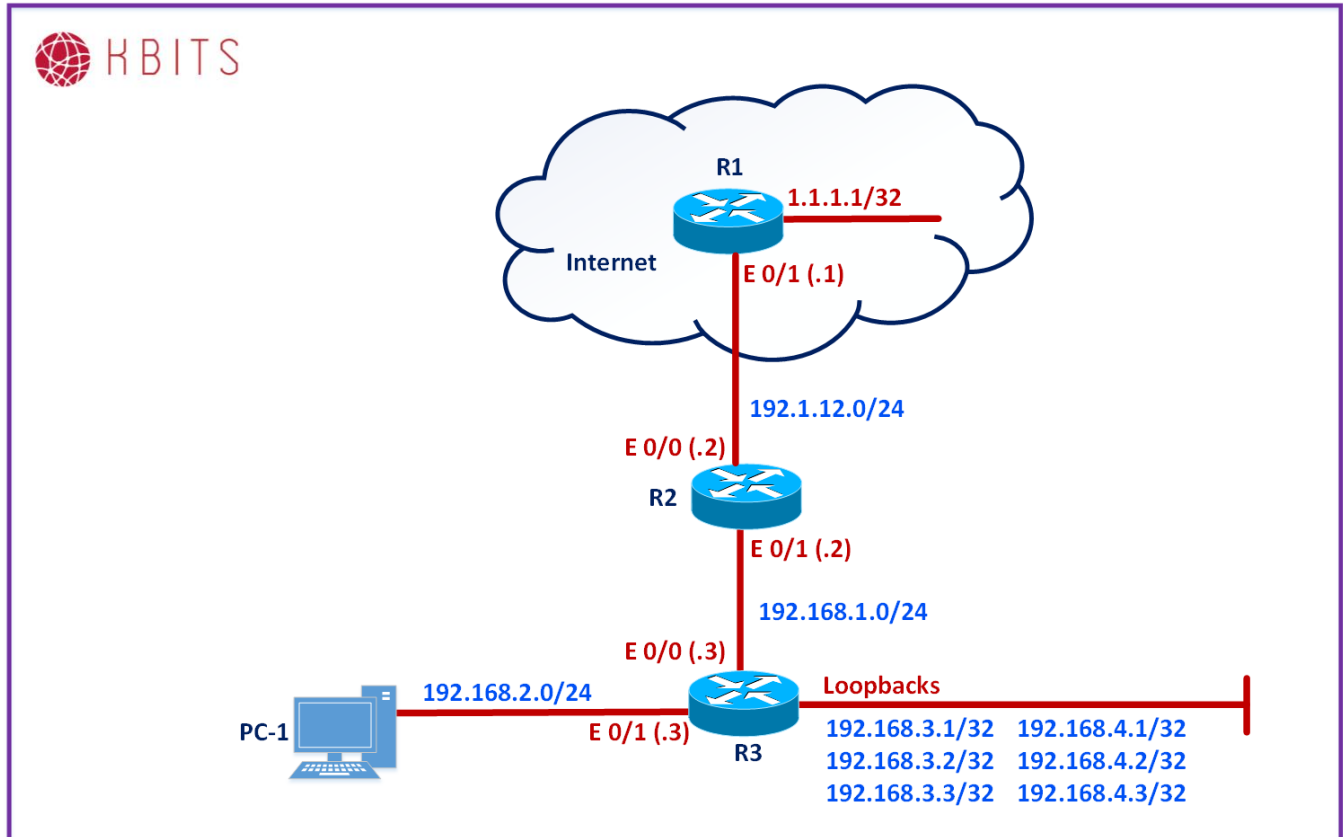
### Verification:

- Telnet to 1.1.1.1 from PC-1.
- Type “**show user**” to verify the ip address used to telnet in.
- Verify the translation table on R2 by using the “**show ip nat translations**” command.



# Lab 11 – Configuring Dynamic PAT

## Logical Topology



### Task 1

Configure a PAT Pool of 192.1.10.11 & 192.1.10.12 on R2. The pool will be used by the 192.168.1.0/24 network. Use POOL2 as the name of the pool. This pool should use Dynamic PAT. Configure an ACL to classify the outbound traffic.

#### R2

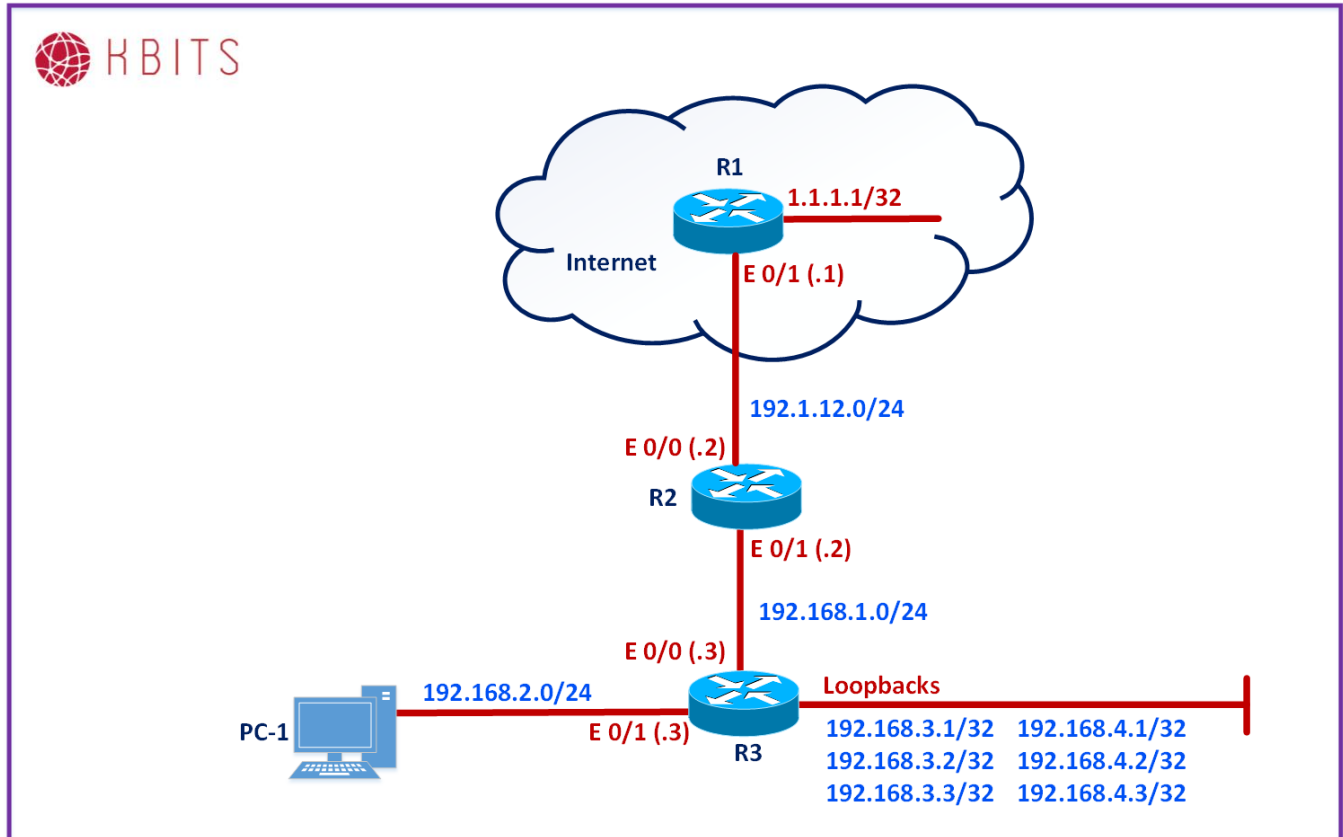
```
Ip nat pool POOL2 192.1.10.11 192.1.10.12 netmask 255.255.255.0
!  
Access-list 102 permit ip 192.168.1.0 0.0.0.255 any
!  
Ip nat inside source list 102 pool POOL2 overload
```

#### Verification:

- Telnet to 1.1.1.1 from R3.
- Type “**show user**” to verify the ip address used to telnet in.
- Verify the translation table on R2 by using the “**show ip nat translations**” command.

# Lab 12 – Configuring Static NAT

## Logical Topology



### Task 1

Configure Static NAT on R2 based on the following:

- 192.168.3.1 – 192.1.10.31
- 192.168.3.2 – 192.1.10.32
- 192.168.3.3 – 192.1.10.33

#### R2

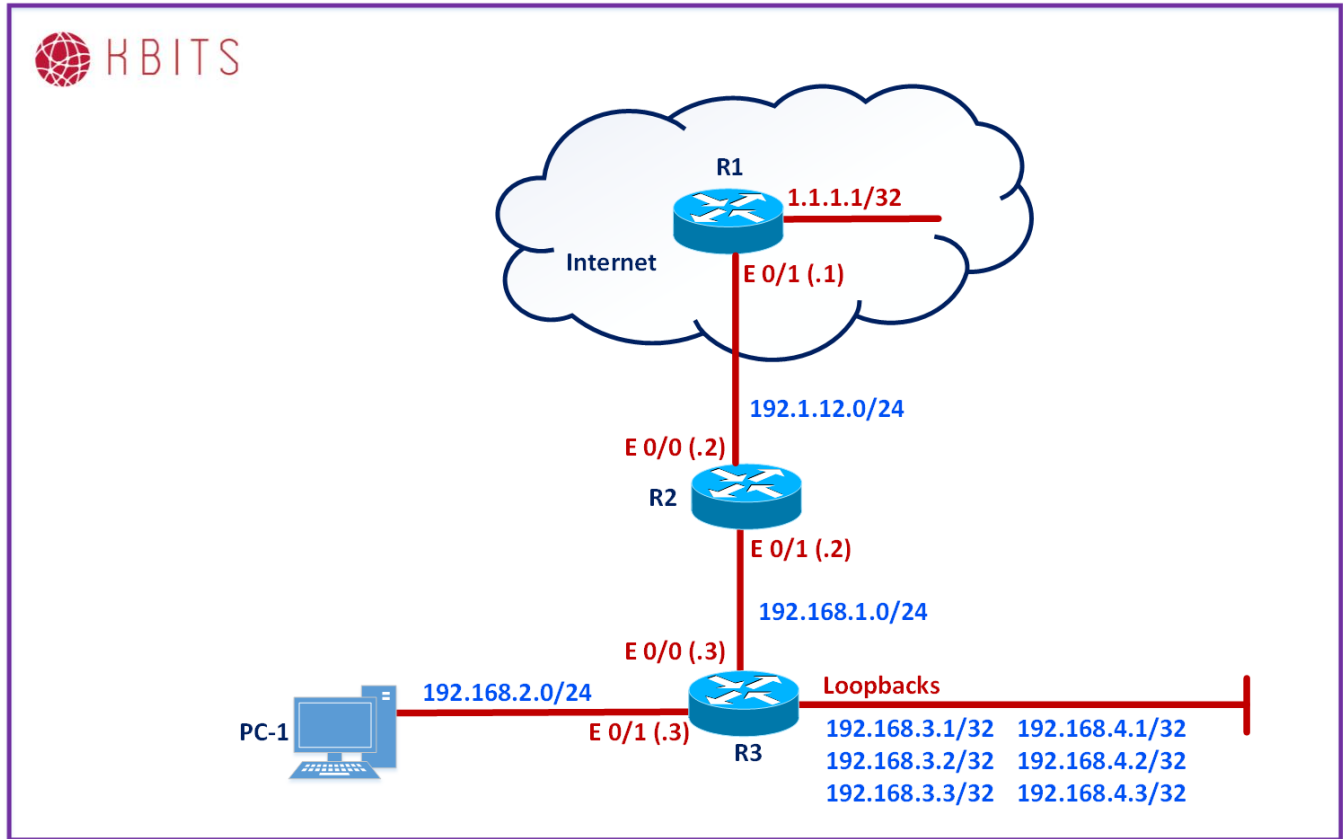
```
Ip nat inside source static 192.168.3.1 192.1.10.31
Ip nat inside source static 192.168.3.2 192.1.10.32
Ip nat inside source static 192.168.3.3 192.1.10.33
```

#### Verification:

- Telnet to 192.1.10.31, 192.1.10.32 & 192.1.10.33 from R1.
- Verify the translation table on R2 by using the “**show ip nat translations**” command.

# Lab 13 – Configuring Static PAT

## Logical Topology



### Task 1

Configure Static PAT on R2 based on the following:

- 192.168.4.1 – 192.1.10.25 – TCP/80
- 192.168.4.2 – 192.1.10.25 – TCP/25
- 192.168.4.3 – 192.1.10.25 – TCP/23
- 192.168.4.3 – 192.1.10.25 – UDP/53

### R2

```
ip nat inside source static tcp 192.168.4.1 80 192.1.10.25 80
ip nat inside source static tcp 192.168.4.2 25 192.1.10.25 25
ip nat inside source static tcp 192.168.4.3 23 192.1.10.25 23
ip nat inside source static udp 192.168.4.3 53 192.1.10.25 53
```

### Verification:

- Telnet to 192.1.10.25 from R1.
- Verify the translation table on R2 by using the “**show ip nat translations**” command.

# Quality of Service (QoS)

**Authored By:**

**Khawar Butt**

CCIE # 12353

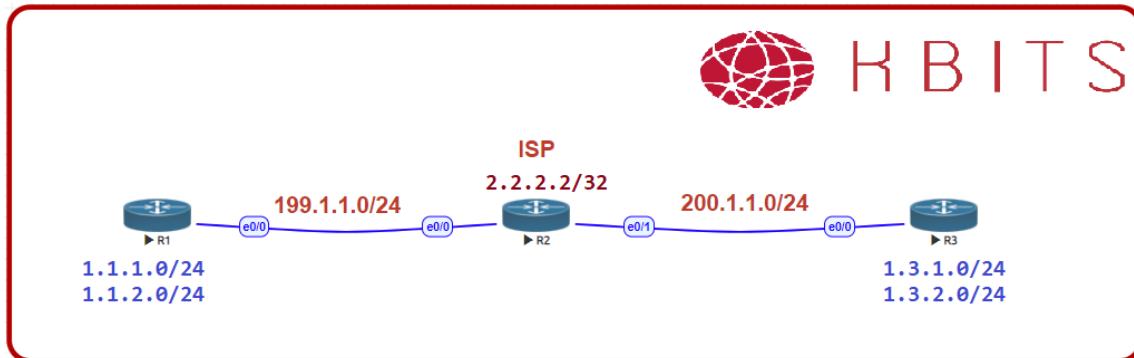
Hepta CCIE#12353

CCDE # 20110020

**Quality of Service (QoS)**



# Lab 1 – Configuring Policing



## Interface IP Address Configuration

### R1

Interface	IP Address	Subnet Mask
Loopback 0	1.1.1.1	255.255.255.0
Loopback 1	1.1.2.1	255.255.255.0
E 0/0	199.1.1.1	255.255.255.0

### R2

Interface	IP Address	Subnet Mask
E 0/0	199.1.1.2	255.255.255.0
E 0/1	200.1.1.2	255.255.255.0
Loopback0	2.2.2.2	255.255.255.255

### R3

Interface	IP Address	Subnet Mask
Loopback 0	1.3.1.1	255.255.255.0
Loopback 1	1.3.2.1	255.255.255.0
E 0/0	200.1.1.3	255.255.255.0



### Task 1

Configure Default routes on R1 & R3 pointing towards R2 (ISP). Configure a Static Route on R2 for 1.1.0.0/16 network pointing towards R1. Configure a Static Route on R2 for 1.3.0.0/16 network point towards R3.

<b>R1</b>	<b>R3</b>
Ip route 0.0.0.0 0.0.0.0 199.1.1.2	Ip route 0.0.0.0 0.0.0.0 200.1.1.2
<b>R2</b>	
Ip route 1.1.0.0 255.255.0.0 199.1.1.1 Ip route 1.3.0.0 255.255.0.0 200.1.1.3	

### Task 2

Configure R1 for Rate Limiting (Policing) for traffic originating from the 1.1.1.0/24 network going towards the Internet using the following parameters:

- ICMP traffic should be limited to 450 kbps
- FTP traffic should be limited to 2 mbps

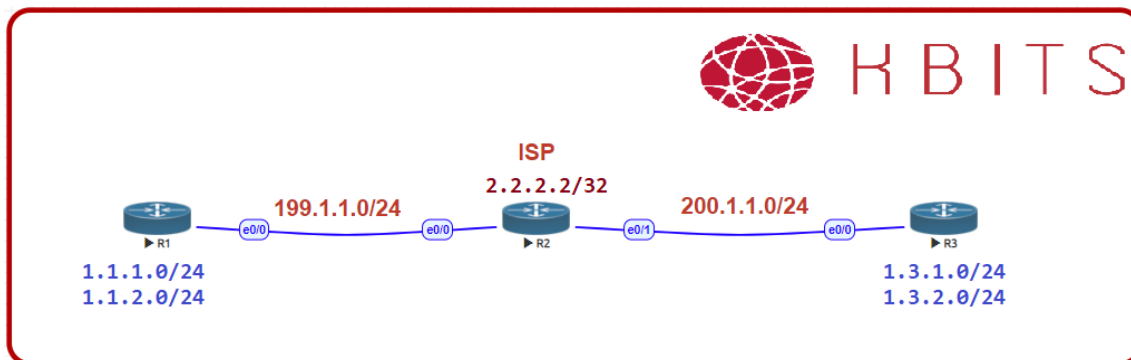
#### R1

```
Access-list 101 permit icmp 1.1.1.0 0.0.0.255 any
!
Access-list 102 permit tcp 1.1.1.0 0.0.0.255 any eq 21
Access-list 102 permit tcp 1.1.1.0 0.0.0.255 any eq 20
!
Class-map CM-ICMP
  Match access-group 101
!
Class-map CM-FTP
  Match access-group 102
!
Policy-map PM-QOS
  Class CM-ICMP
    Police 450000
  Class CM-FTP
    Police 2000000
!
Interface E 0/0
  Service-policy output PM-QOS
```

**Verification:**

- Ping 2.2.2.2 from R1 using a source of 1.1.1.1.
- Verify the QoS Policy on R1 by using the “**show policy-map interface E 0/0**” command. You should see hit counts on the Policy for ICMP Policing.

# Lab 2 – Configuring Congestion Management with Bandwidth Reservation



## Task 1

Configure R3 Ethernet interface for Bandwidth Reservation using the following:

- HTTP and HTTPS traffic = Reserve 30% of the Bandwidth
- Telnet Traffic = Reserve 10% of the Bandwidth

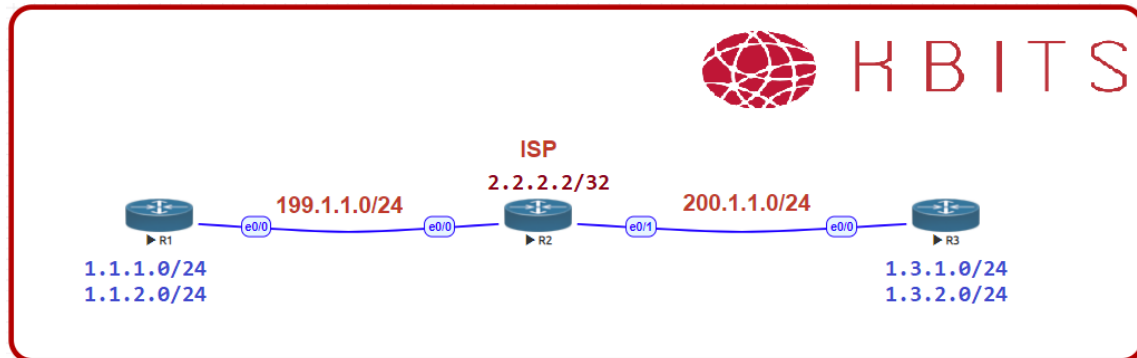
## R3

```
Access-list 111 permit tcp any any eq 80
Access-list 111 permit tcp any any eq 443
!
Access-list 112 permit tcp any any eq 23
!
class-map CM-WEB
 match access-group 111
!
class-map CM-TELNET
 match access-group 112
!
policy-map PM-QOS
 class CM-TELNET
  bandwidth percent 10
 class CM-WEB
  bandwidth percent 30
!
Interface E 0/0
Service-policy output PM-QOS
```

**Verification:**

- Telnet into 2.2.2.2 from R3 using a source of 1.3.1.1.
- Verify the QoS Policy on R3 by using the “**show policy-map interface E 0/0**” command. You should see hit counts on the Policy for the Telnet Policy.

# Lab 3 – Configuring Congestion Management with Low-Latency Queuing (LLQ)



## Task 1

Configure R3 Ethernet interface for LLQ using the following:

- SSH Traffic = 15% of the bandwidth. SSH traffic should use LLQ.

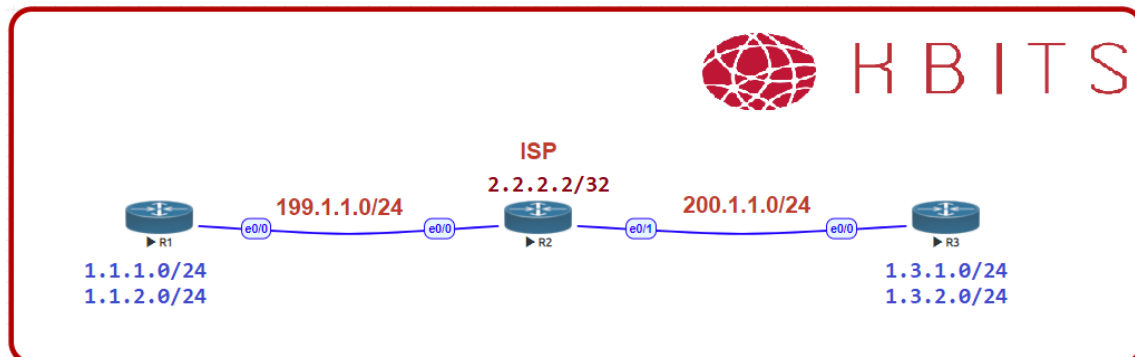
## R3

```
Access-list 113 permit tcp any any eq 22
!
class-map CM-SSH
 match access-group 113
!
policy-map PM-QOS
 class CM-SSH
  priority percent 15
```

## Verification:

- SSH into 2.2.2.2 from R3 using a source of 1.3.1.1.
- Verify the QoS Policy on R3 by using the “**show policy-map interface E 0/0**” command. You should see hit counts on the Policy for the SSH Policy.

# Lab 4 – Classifying Traffic Using NBAR



## Task 1

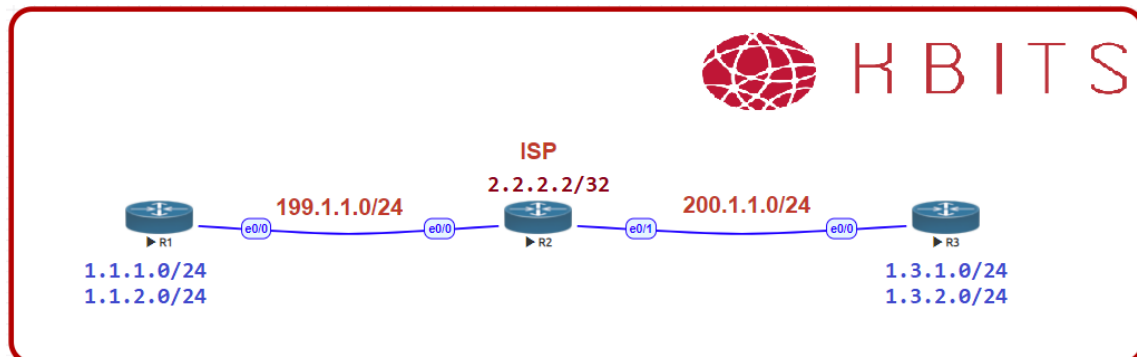
Configure R1 Ethernet interface for QoS using the following criteria:

- Configure Policing such that HTTP downloading is policed at 100 kbps for \*.gif or \*.jpg files.

## R1

```
class-map match-any CM-WEB-P2P
  match protocol http url "*.gif"
  match protocol http url "*.jpg"
!
policy-map PM-QOS
  class CM-WEB-P2P
    police 100000
```

# Lab 5 Configuring Shaping



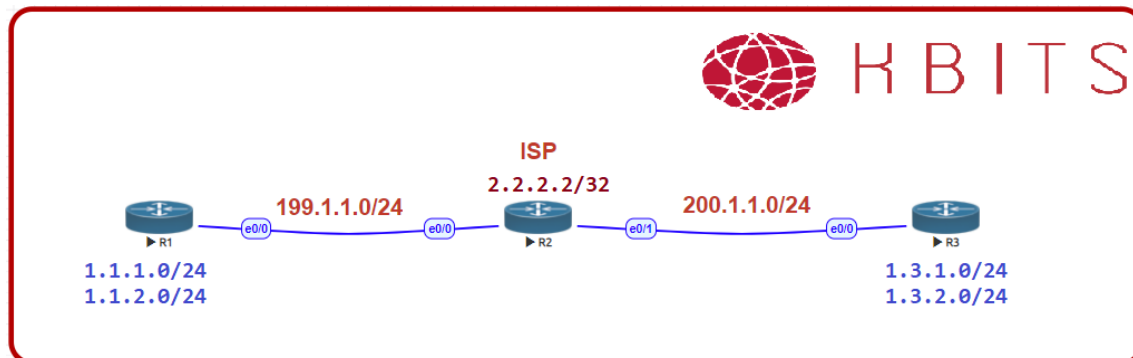
## Task 1

Configure R1 such that all FTP or TFTP traffic going towards 1.3.0.0/16 network should have a minimum reserved bandwidth of 256 kbps and should be shaped to the 512 kbps.

## R1

```
Access-list 115 permit tcp any 1.3.0.0 0.0.255.255 eq 20
Access-list 115 permit tcp any 1.3.0.0 0.0.255.255 eq 21
Access-list 115 permit udp any 1.3.0.0 0.0.255.255 eq 69
!
Class-map match-any CM-FT
Match access-group 115
!
Policy-map PM-QOS
Class CM-FT
Bandwidth 256
Shape average 512000
```

# Lab 6 – Configuring Advanced Class Maps



## Task 1

Any traffic that is destined to either of the URLs specified below should be prioritized from the 1.3.2.0/24 network should be prioritized going towards the Internet. Set the priority percent to 20.

- Cisco.com
- Kbits.live

## R3

```
Access-list 116 permit ip 1.3.2.0 0.0.0.255 any
!
Class-map match-any CM-URLS
  Match protocol http url "*"Cisco.com*"
  Match protocol http url "*"Kbits.live*"
!
Class-map match-all CM-NESTING
  Match class-map CM-URLS
  Match access-group 116
!
Policy-map PM-QOS
  Class CM-NESTING
    Priority percent 20
```



# Multicast Routing

**Authored By:**

**Khawar Butt**

CCIE # 12353

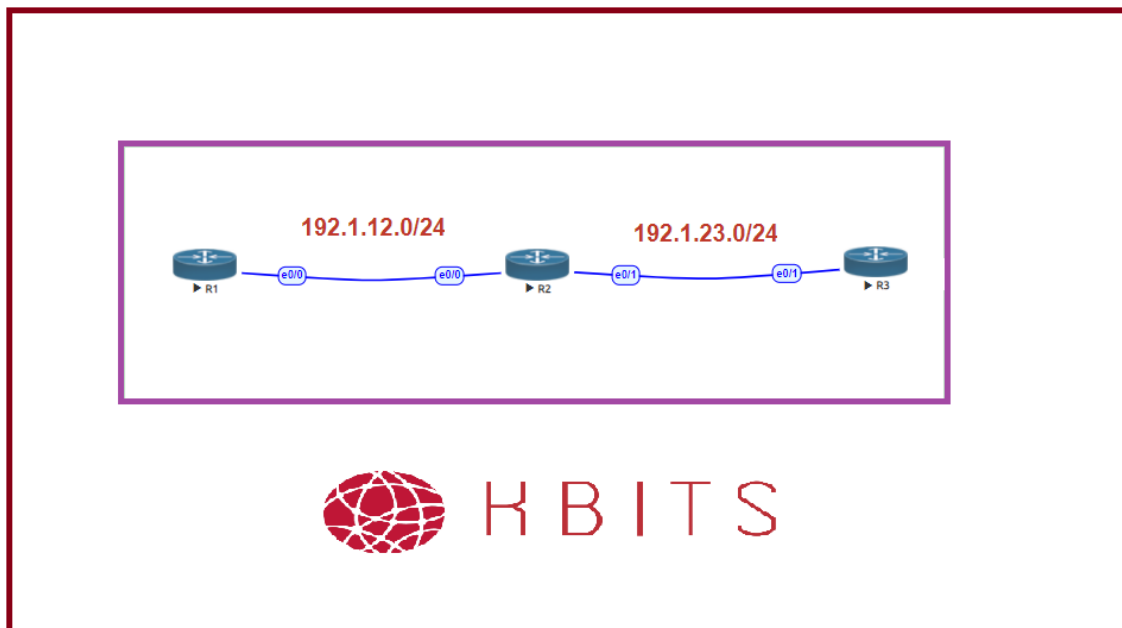
Hepta CCIE#12353

CCDE # 20110020

**Multicast-Routing**



# Lab 1 – Configuring PIM – Dense-Mode



## Interface IP Address Configuration

### R1

Interface	IP Address	Subnet Mask
Loopback 0	1.1.1.1	255.255.255.0
Loopback 1	1.1.2.1	255.255.255.0
E 0/0	192.1.12.1	255.255.255.0

### R2

Interface	IP Address	Subnet Mask
Loopback0	1.2.1.1	255.255.255.0
Loopback1	1.2.2.1	
E 0/0	192.1.12.2	255.255.255.0
E 0/1	192.1.23.2	255.255.255.0

### R3

Interface	IP Address	Subnet Mask
Loopback 0	1.3.1.1	255.255.255.0
Loopback 1	1.3.2.1	255.255.255.0
E 0/0	192.1.23.3	255.255.255.0

### Task 1

Configure EIGRP as the Routing protocol in AS 111 between R1, R2 & R3. Enable all interfaces on all 3 routers in EIGRP.

<b>R1</b> Router eigrp 111 Network 192.1.12.0 Network 1.0.0.0	<b>R3</b> Router eigrp 111 Network 192.1.23.0 Network 1.0.0.0
<b>R2</b> Router eigrp 111 Network 192.1.12.0 Network 192.1.23.0 Network 1.0.0.0	

### Task 2

Configure PIM dense mode on the routers.

<b>R1</b> Ip multicast-routing ! Int E 0/0 Ip pim dense-mode ! Int Loopback0 Ip pim dense-mode Int Loopback1 Ip pim dense-mode	<b>R3</b> Ip multicast-routing ! Int E 0/0 Ip pim dense-mode ! Int Loopback0 Ip pim dense-mode Int Loopback1 Ip pim dense-mode
<b>R2</b> Ip multicast-routing ! Int E 0/0 Ip pim dense-mode ! Int E 0/1 Ip pim dense-mode ! Int Loopback0 Ip pim dense-mode ! Int Loopback1 Ip pim dense-mode	

### Task 3

Configure R1 & R3 to join the multicast group 224.1.1.3 on the Loopback 0 interfaces.

<b>R1</b> Int Loopback 0 Ip igmp join-group 224.1.1.3	<b>R3</b> Int Loopback 0 Ip igmp join-group 224.1.1.3
---	---

### Task 4

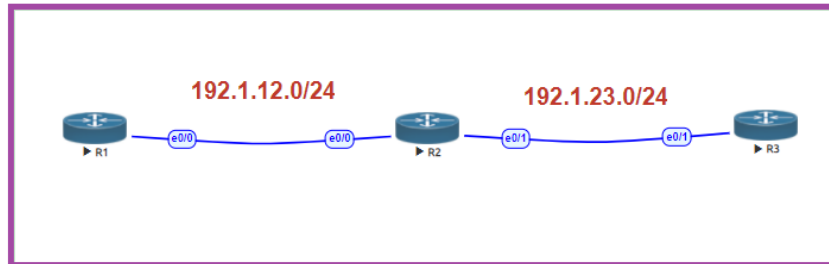
Configure R1, R2 & R3 to join the multicast group 224.1.2.3 on the Loopback 1 interfaces.

<b>R1</b> Int Loopback 1 Ip igmp join-group 224.1.2.3	<b>R3</b> Int Loopback 0 Ip igmp join-group 224.1.2.3
<b>R2</b> Int Loopback 1 Ip igmp join-group 224.1.2.3	

### Verification:

- Ping 224.1.1.3 from R2. You should receive a reply from R1 & R3.
- Ping 224.1.2.3 from R2. You should receive a reply from all 3 routers.

## Lab 2 – Configuring PIM – Sparse-Mode using Single Static RP



### Task 1

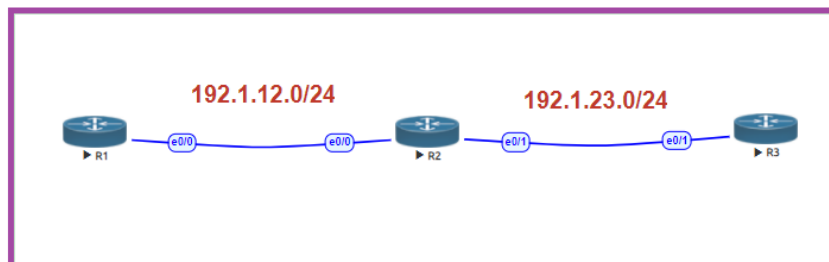
Configure R2 to be the RP for all groups. Use the Loopback0 as the IP Address of the RP.

<b>R1</b>  Interface loopback0 Ip pim sparse-mode ! Interface loopback1 Ip pim sparse-mode ! Int E 0/0 Ip pim sparse-mode ! Ip pim rp-address 1.2.1.1	<b>R2</b>  Interface loopback0 Ip pim sparse-mode ! Interface loopback1 Ip pim sparse-mode ! Int E 0/0 Ip pim sparse-mode ! Int E 0/1 Ip pim sparse-mode ! Ip pim rp-address 1.2.1.1
<b>R3</b>  Interface loopback0 Ip pim sparse-mode ! Interface loopback1 Ip pim sparse-mode ! Int E 0/0 Ip pim sparse-mode ! Ip pim rp-address 1.2.1.1	

### Verification:

- Ping 224.1.1.3 from R2. You should receive a reply from R1 & R3.
- Ping 224.1.2.3 from R2. You should receive a reply from all 3 routers.

# Lab 3 – Configuring PIM – Sparse-Mode using Multiple Static RP



### Task 1

De-configure R2 (1.2.1.1) as the Static RP on all 3 Routers.

<b>R1</b>  No ip pim rp-address 1.2.1.1	<b>R2</b>  No ip pim rp-address 1.2.1.1
<b>R3</b>  No ip pim rp-address 1.2.1.1	

### Task 2

Configure R1 to be the RP for Multicast groups 224.1.1.3, and R2 to be the RP for the groups 224.1.2.3. These two RPs should use their Loopback 0 interface for this purpose.

<b>R1</b>  Access-list 10 permit 224.1.1.3 Access-list 20 permit 224.1.2.3 ! Ip pim rp-address 1.1.1.1 10 Ip pim rp-address 1.2.1.1 20	<b>R2</b>  Access-list 10 permit 224.1.1.3 Access-list 20 permit 224.1.2.3 ! Ip pim rp-address 1.1.1.1 10 Ip pim rp-address 1.2.1.1 20!
<b>R3</b>	

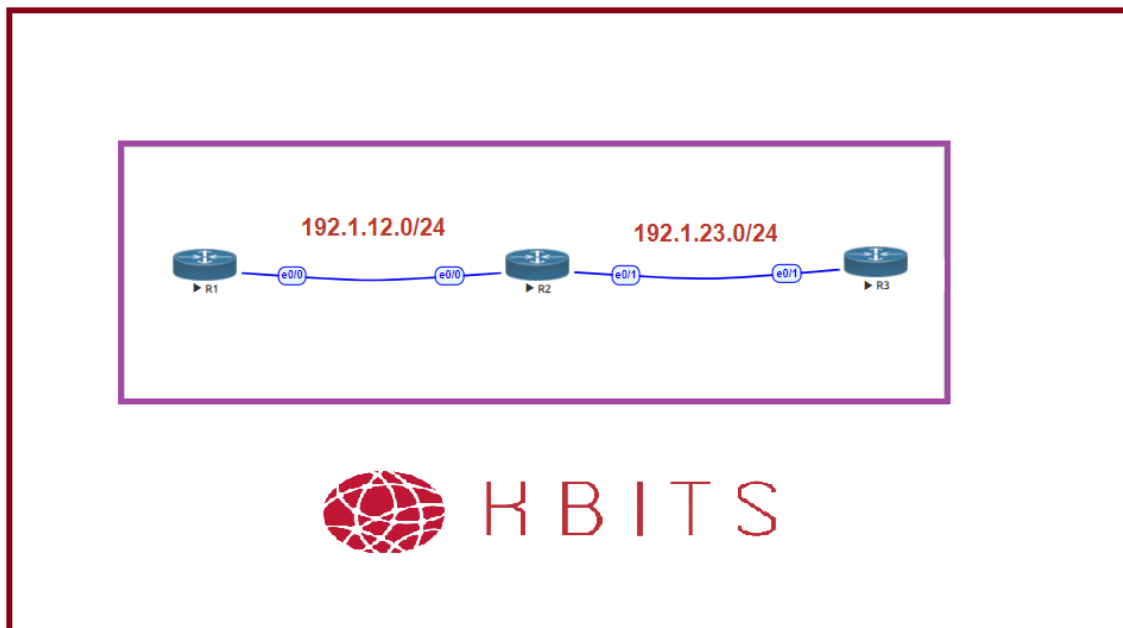
Access-list 10 permit 224.1.1.3 Access-list 20 permit 224.1.2.3 ! Ip pim rp-address 1.1.1.1 10 Ip pim rp-address 1.2.1.1 20	
---	--

**Verification:**

- Type “**Show ip pim rp mapping**” to verify the RP assignment.
- Ping 224.1.1.3 from R2. You should receive a reply from R1 & R3.
- Ping 224.1.2.3 from R2. You should receive a reply from all 3 routers.



# Lab 4 – Configuring PIM – Sparse-Mode using Dense-Mode for Fallback



## Task 1

Configure R1 Loopback 0 and R3 loopback 0 to join the following Multicast groups:

R1 – 224.11.11.1, 224.11.11.2, 224.11.11.3

R3 – 224.33.33.1, 224.33.33.2, 224.33.33.3

R1	R3
<pre>Interface Loopback0 Ip igmp join-group 224.11.11.1 Ip igmp join-group 224.11.11.2 Ip igmp join-group 224.11.11.3</pre>	<pre>Interface Loopback0 Ip igmp join-group 224.33.33.1 Ip igmp join-group 224.33.33.2 Ip igmp join-group 224.33.33.3</pre>

## Verification:

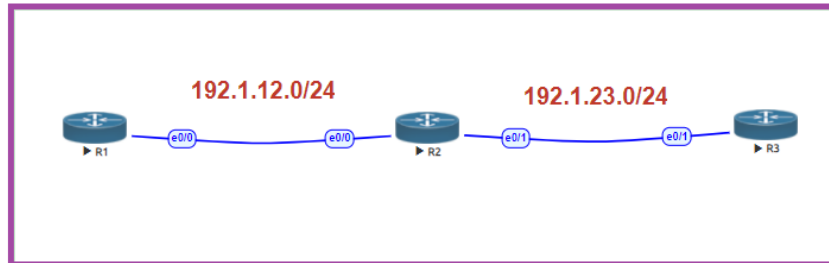
- Ping any of the new joined Multicast groups from R2. Are you receiving a reply?

## Task 2

Configure PIM on the physical and loopback interfaces in such a way that all routers have access to all the multicast groups, including the ones that are not configured for RP's.

<b>R1</b> Interface F 0/0 Ip pim sparse-dense-mode ! Interface Loopback0 Ip pim sparse-dense-mode	<b>R2</b> Interface F 0/0 Ip pim sparse-dense-mode ! Interface F 0/1 Ip pim sparse-dense-mode
<b>R3</b> Interface F 0/0 Ip pim sparse-dense-mode ! Interface F 0/1 Ip pim sparse-dense-mode	

# Lab 5 – Configuring PIM – Sparse Mode – Auto RP



## Task 1

De-configure R1 & R2 as the Static RP on all 3 Routers.

<b>R1</b> Ip pim rp-address 1.1.1.1 10 Ip pim rp-address 1.2.1.1 20	<b>R2</b> Ip pim rp-address 1.1.1.1 10 Ip pim rp-address 1.2.1.1 20
<b>R3</b> Ip pim rp-address 1.1.1.1 10 Ip pim rp-address 1.2.1.1 20	

## Task 2

Configure BSR for RP Election and Announcements. Configure R1 & R3 as RP-Candidates. They should use their Loopback0 to send their RP-Candidate announcements. The announcements should be sent every 10 seconds with a TTL of 5. Configure R2 as the Mapping Agent. It should also use its Loopback 0.

### R2

```
Ip pim send-rp-discovery scope 5
```

### R1

```
Ip pim send-rp-announce loopback0 scope 5 interval 10
```

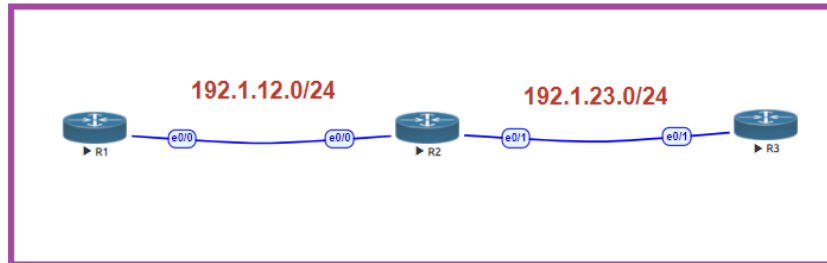
### R3

```
Ip pim send-rp-announce loopback0 scope 5 interval 10
```

### Verification:

- Type “**Show ip pim rp mapping**” to verify the RP assignment. Who has been elected as the RP? What method was used for the election and announcement?
- Ping 224.1.1.3 from R2. You should receive a reply from R1 & R3.
- Ping 224.1.2.3 from R2. You should receive a reply from all 3 routers.

# Lab 6 – Configuring PIM – Sparse Mode – BSR



## Task 1

De-configure Auto-RP on all 3 Routers.

### R2

```
No Ip pim send-rp-discovery scope 5
```

### R1

```
No Ip pim send-rp-announce loopback0 scope 5 interval 10
```

### R3

```
No Ip pim send-rp-announce loopback0 scope 5 interval 10
```

## Task 2

Configure BSR for RP Election and Announcements. Configure R1 & R3 as RP-Candidates. They should use their Loopback0 to send their RP-Candidate announcements. R1 should be the preferred RP. Configure R2 as the Mapping Agent (BSR Candidate). It should also use its Loopback 0.

### R2

```
ip pim bsr-candidate Loopback0
```

### R1

```
ip pim rp-candidate Loopback0 priority 0
```

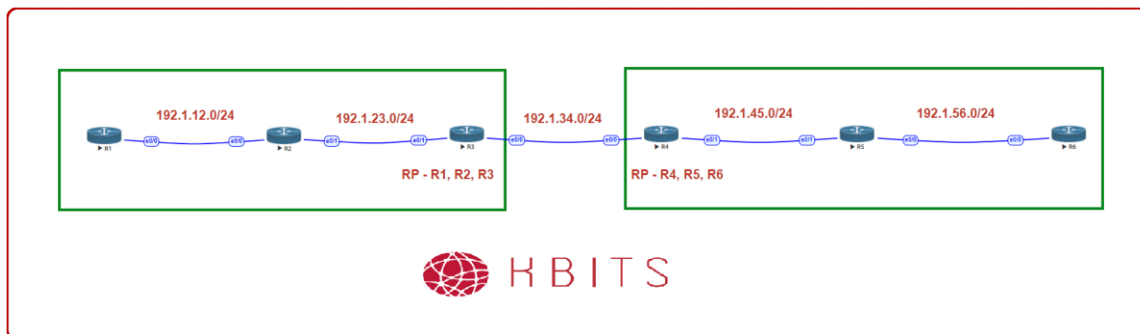
### R3

```
ip pim rp-candidate Loopback0 priority 10
```

### Verification:

- Type “**Show ip pim rp mapping**” to verify the RP assignment. Who has been elected as the RP? What method was used for the election and announcement?
- Ping 224.1.1.3 from R2. You should receive a reply from R1 & R3.
- Ping 224.1.2.3 from R2. You should receive a reply from all 3 routers.

# Lab 7 – Configuring MSDP



## R1

Interface	IP Address	Subnet Mask
Loopback 0	1.1.1.1	255.255.255.0
E 0/0	192.1.12.1	255.255.255.0

## R2

Interface	IP Address	Subnet Mask
E 0/0	192.1.12.2	255.255.255.0
E 0/1	192.1.23.2	255.255.255.0
Loopback 0	1.2.1.1	255.255.255.0

## R3

Interface	IP Address	Subnet Mask
E 0/1	192.1.23.3	255.255.255.0
E 0/0	192.1.34.3	255.255.255.0
Loopback 0	1.3.1.1	255.255.255.0

## R4

Interface	IP Address	Subnet Mask
E 0/0	192.1.34.4	255.255.255.0
E 0/1	192.1.34.4	255.255.255.0
Loopback 0	1.4.1.1	255.255.255.0

## R5

Interface	IP Address	Subnet Mask
E 0/0	192.1.45.5	255.255.255.0
E 0/1	192.1.56.5	255.255.255.0
Loopback 0	1.5.1.1	255.255.255.0

## R6

Interface	IP Address	Subnet Mask
E 0/0	192.1.56.6	255.255.255.0
Loopback 0	1.6.1.1	255.255.255.0

### Task 1

Configure EIGRP 100 on all routers and advertise all the directly connected networks.

<b>R1</b>  Router EIGRP 100 Network 1.0.0.0 Network 192.1.12.0	<b>R2</b>  Router EIGRP 100 Network 1.0.0.0 Network 192.1.12.0 Network 192.1.23.0
<b>R3</b>  Router EIGRP 100 Network 1.0.0.0 Network 192.1.23.0 Network 192.1.34.0	<b>R4</b>  Router EIGRP 100 Network 1.0.0.0 Network 192.1.34.0 Network 192.1.45.0
<b>R5</b>  Router EIGRP 100 Network 1.0.0.0 Network 192.1.45.0 Network 192.1.56.0	<b>R6</b>  Router EIGRP 100 Network 1.0.0.0 Network 192.1.56.0



## Task 2

Enable Multicast Routing on R1, R2 & R3 using PIM Sparse-Mode on all the interfaces. They should be configured to use R3 Loopback 0 as the RP address for all Multicast groups. Configure R1, R2 & R3 to join 224.1.2.3 & 224.12.34.56 on the Loopback 0 interfaces.

<pre><b>R1</b>  Ip multicast-routing ! Interface Loopback 0 Ip pim sparse-mode Ip igmp join-group 224.1.2.3 Ip igmp join-group 224.12.34.56 ! Interface E 0/0 Ip pim sparse-mode ! Ip pim rp-address 1.3.1.1</pre>	<pre><b>R2</b>  Ip multicast-routing ! Interface Loopback 0 Ip pim sparse-mode Ip igmp join-group 224.1.2.3 Ip igmp join-group 224.12.34.56 ! Interface E 0/0 Ip pim sparse-mode ! Interface E 0/1 Ip pim sparse-mode ! Ip pim rp-address 1.3.1.1</pre>
<pre><b>R3</b>  Ip multicast-routing ! Interface Loopback 0 Ip pim sparse-mode Ip igmp join-group 224.1.2.3 Ip igmp join-group 224.12.34.56 ! Interface E 0/0 Ip pim sparse-mode ! Interface E 0/1 Ip pim sparse-mode ! Ip pim rp-address 1.3.1.1</pre>	

## Verification:

- Type “**Show ip pim rp mapping**” to verify the RP assignment.
- Ping 224.1.2.3 & 224.12.34.56 from R1, R2 & R3. You should receive a reply from all 3 routers (R1, R2 & R3).

### Task 3

Enable Multicast Routing on R4, R5 & R6 using PIM Sparse-Mode on all the interfaces. They should be configured to use R4 Loopback 0 as the RP address for all Multicast groups. Configure R4, R5 & R6 to join 224.4.5.6 & 224.12.34.56 on the Loopback 0 interfaces.

<pre><b>R4</b>  Ip multicast-routing ! Interface Loopback 0 Ip pim sparse-mode Ip igmp join-group 224.4.5.6 Ip igmp join-group 224.12.34.56 ! Interface E 0/0 Ip pim sparse-mode ! Interface E 0/1 Ip pim sparse-mode ! Ip pim rp-address 1.4.1.1</pre>	<pre><b>R5</b>  Ip multicast-routing ! Interface Loopback 0 Ip pim sparse-mode Ip igmp join-group 224.4.5.6 Ip igmp join-group 224.12.34.56 ! Interface E 0/0 Ip pim sparse-mode ! Interface E 0/1 Ip pim sparse-mode ! Ip pim rp-address 1.4.1.1</pre>
<pre><b>R6</b>  Ip multicast-routing ! Interface Loopback 0 Ip pim sparse-mode Ip igmp join-group 224.4.5.6 Ip igmp join-group 224.12.34.56 ! Interface E 0/0 Ip pim sparse-mode ! Interface E 0/1 Ip pim sparse-mode ! Ip pim rp-address 1.4.1.1</pre>	

### Verification:

- Type “**Show ip pim rp mapping**” to verify the RP assignment.
- Ping 224.4.5.6 & 224.12.34.56 from R4, R5 & R6. You should receive a reply from R4, R5 & R6 only.

#### Task 4

Configure a MSDP peering between the 2 RPs, R3 & R4 based on the loopback interfaces.

##### R3

```
Ip msdp peer 4.4.4.4 connect-source Loopback 0
```

##### R4

```
Ip msdp peer 3.3.3.3 connect-source Loopback 0
```

#### Verification:

- Type “**Show ip msdp peer**” to verify that the connection is up.
- Ping 224.12.34.56 from any router. You should receive a reply from all routers.

# Automation & Python Programming

**Authored By:**

**Khawar Butt**

CCIE # 12353

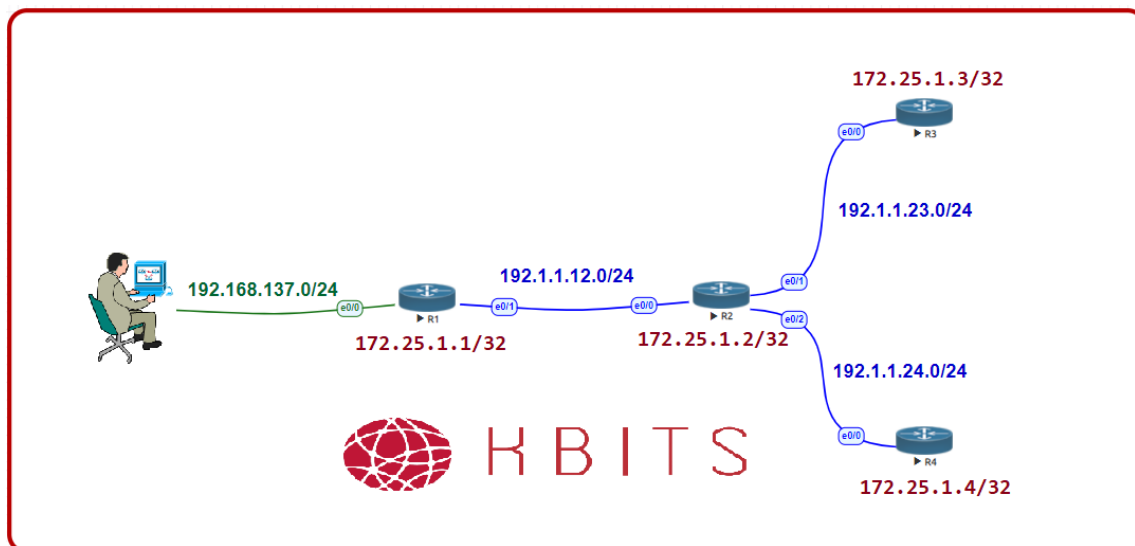
Hepta CCIE#12353

CCDE # 20110020

**Automation & Python  
Programming**



# Lab 1 – Configuring EEM – Controlling Interface Shutdown



## Interface IP Address Configuration

### R1

Interface	IP Address	Subnet Mask
Loopback 0	1.1.1.1	255.255.255.0
Loopback10000	172.25.1.1	255.255.255.255
E 0/0	192.168.137.111	255.255.255.0
E 0/1	192.1.12.1	255.255.255.0

### R2

Interface	IP Address	Subnet Mask
Loopback0	1.2.1.1	255.255.255.0
Loopback10000	172.25.1.2	255.255.255.255
E 0/0	192.1.12.2	255.255.255.0
E 0/1	192.1.23.2	255.255.255.0
E 0/2	192.1.24.2	255.255.255.0

### R3

Interface	IP Address	Subnet Mask
Loopback 0	1.3.1.1	255.255.255.0
Loopback10000	172.25.1.3	255.255.255.255
E 0/0	192.1.23.3	255.255.255.0

### R4

Interface	IP Address	Subnet Mask
Loopback 0	1.4.1.1	255.255.255.0
Loopback10000	172.25.1.4	255.255.255.255
E 0/0	192.1.24.4	255.255.255.0

### Task 1

Configure EIGRP as the Routing protocol in AS 111 between R1, R2, R3 & R4. Enable all interfaces on all 3 routers in EIGRP.

<b>R1</b>  Router eigrp 111 Network 192.1.12.0 Network 192.168.137.0 Network 1.0.0.0 Network 172.25.0.0	<b>R2</b>  Router eigrp 111 Network 192.1.12.0 Network 192.1.23.0 Network 192.1.24.0 Network 1.0.0.0 Network 172.25.0.0
<b>R3</b>  Router eigrp 111 Network 192.1.23.0 Network 1.0.0.0 Network 172.25.0.0	<b>R4</b>  Router eigrp 111 Network 192.1.24.0 Network 1.0.0.0 Network 172.25.0.0

## Task 2

Configure Telnet & SSH Access on all the routers. Create a user “khawar” with a password of “cisco”. Assign it a privilege level of 15.

### R1

```
Ip domain-name kbits.live
Crypto key generate rsa modulus 1024
!
Username khawar privilege 15 password cisco
!
Line vty 0 4
Login loca
Transport input telnet ssh
```

### R2

```
Ip domain-name kbits.live
Crypto key generate rsa modulus 1024
!
Username khawar privilege 15 password cisco
!
Line vty 0 4
Login loca
Transport input telnet ssh
```

### R3

```
Ip domain-name kbits.live
Crypto key generate rsa modulus 1024
!
Username khawar privilege 15 password cisco
!
Line vty 0 4
Login loca
Transport input telnet ssh
```

### R4

```
Ip domain-name kbits.live
Crypto key generate rsa modulus 1024
!
Username khawar privilege 15 password cisco
!
Line vty 0 4
Login loca
Transport input telnet ssh
```

### Task 3

Configure an EEM Applet that will make sure that the E 0/0 interface on R1 is never administratively Shutdown. The Applet should a message stating “This is a critical interface. Please don't shut it”. It should send this console message after bringing the interface back up.

#### R1

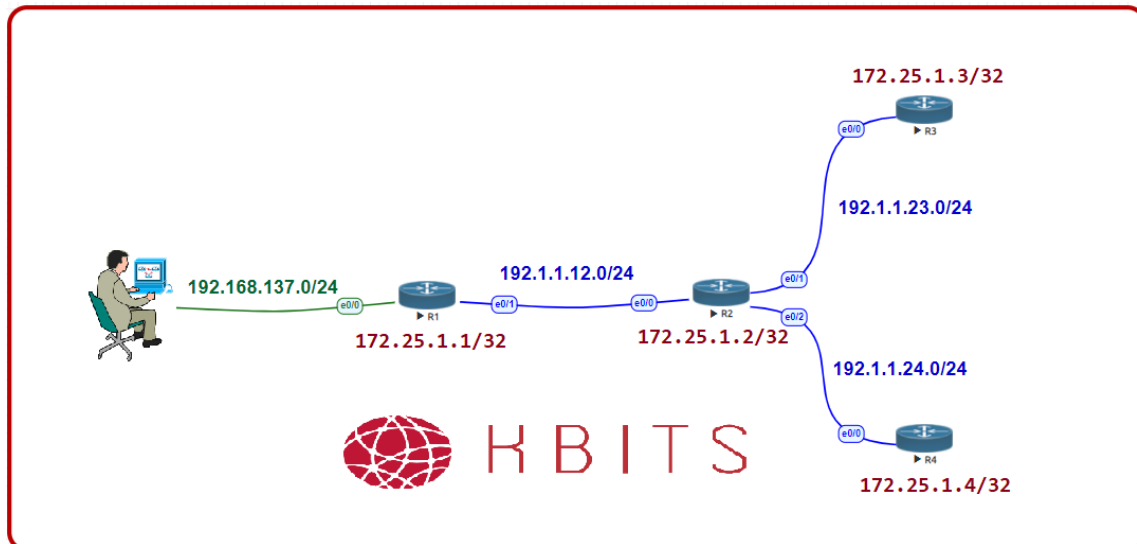
```
event manager applet INTER-RESET-E0/0
event syslog pattern "Interface Ethernet0/0, changed state to administratively down"
action 1.0 cli command "enable"
action 2.0 cli command "conf t"
action 3.0 cli command "interface E0/0"
action 4.0 cli command "no shut"
action 5.0 syslog msg "This is a critical interface. Please don't shut it"
```

#### Verification:

- Shut the E 0/0 interface down on R1.
- The EEM Applet should bring the interface back up and display the message on the console.



# Lab 2 – Configuring EEM – E-Mailing Errors to Administrators



## Task 1

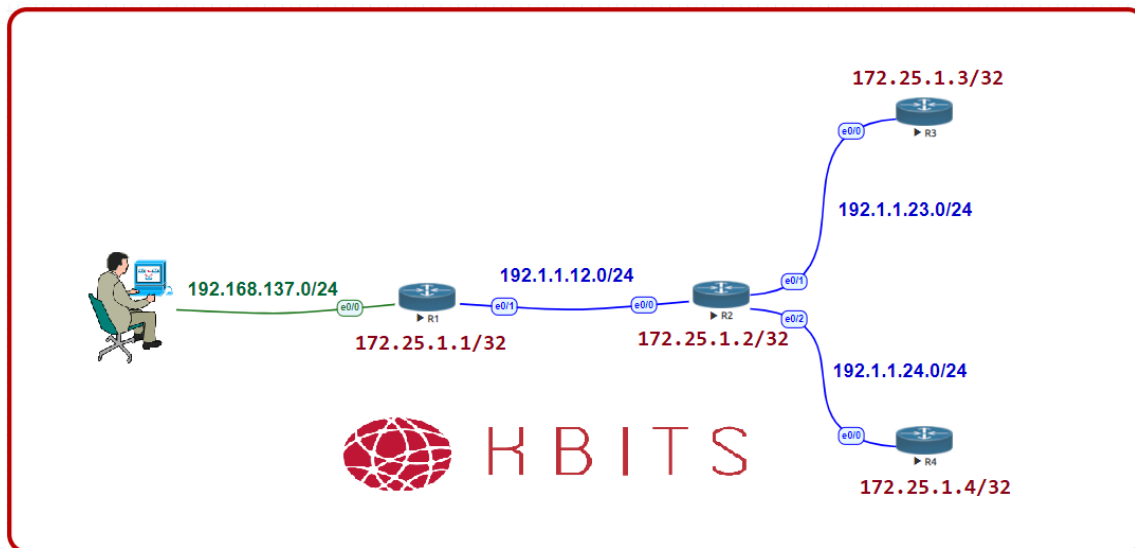
Configure an EEM Applet on R2 such it notifies the Support Team that the EIGRP neighbor relationship with R1 has gone down in EIGRP AS 111. Use the following parameters for the applet:

- Syslog Message Pattern: "EIGRP-IPv4 111: Neighbor 192.1.12.1 (Ethernet0/0) is down"
- Mail Server: 192.1.12.25
- Support Team E-mail: [errors@kbits.live](mailto:errors@kbits.live)
- Router E-mail: [R2@kbits.live](mailto:R2@kbits.live)
- Subject: "EIGRP IS DOWN" body "Please fix EIGRP"

## R2

```
event manager applet EIGRP_DOWN
event syslog pattern " EIGRP-IPv4 111: Neighbor 192.1.12.1 (Ethernet0/0) is down:
holding time expired
action 1.0 cli command "enable"
action 2.0 mail server "192.1.12.25" to "errors@kbits.live" from "R2@kbits.live"
subject "EIGRP IS DOWN" body "Please fix EIGRP"
```

# Lab 3 – Retrieving Information from Routers Using Python – Interactive



### Task 1

Configure a Python Script that allows you to input a show command and retrieves the information from R1 and displays it on the console. Use the following to write the script.

- Use the “Telnet” function from the **telnetlib** library.
- Create a variable called **cmd** to receive the input using the “input” function.
- Host IP: **172.25.1.1**
- Login Username: **khawar**
- Password: **cisco**

#### Admin PC

```
from telnetlib import Telnet

cmd = input('Enter the Command : ')

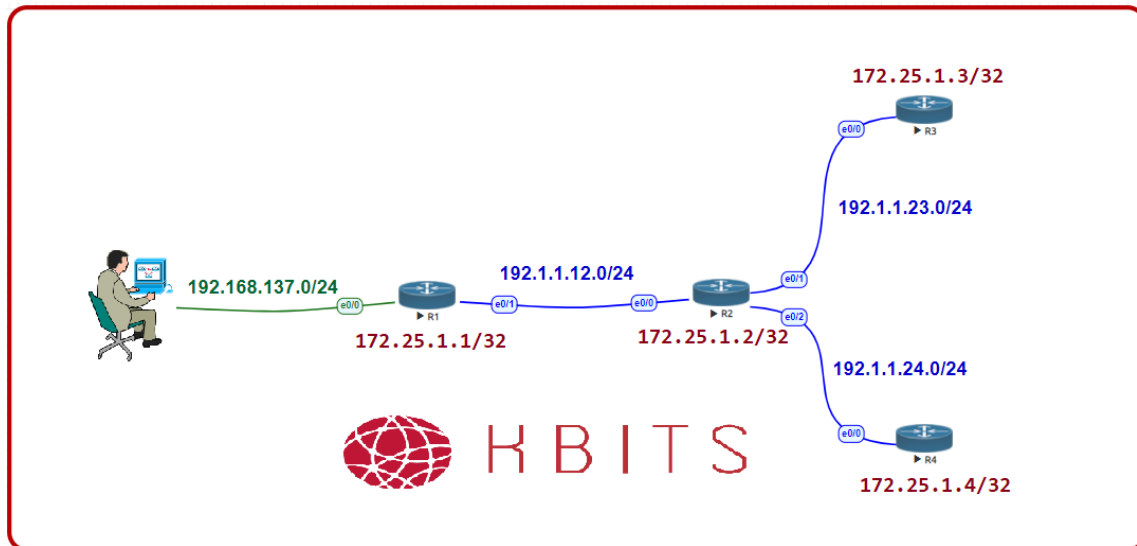
ab = Telnet('172.25.1.1')
ab.write(b'khawar\n')
ab.write(b'cisco\n')
ab.write(b'term len 0\n')
ab.write(cmd.encode('ascii') + b'\n')
ab.write(b'exit\n')

print (ab.read_all().decode('ascii'))
```

#### Verification:

- Save the script as Lab3.py.
- Run the script. It will prompt you for the Command. Type a command of your choice (Example: **show ip interface brief**)
- Verify the output on the console.

# Lab 4 – Configuring Network Devices Using Python



### Task 1

Configure a Python Script that allows you to configure a Loopback Interface on R1 and displays the Interface that was created by using the “**Show ip interface brief**” command. Use the following to write the script.

- Use the “Telnet” function from the **telnetlib** library.
- Host IP: **172.25.1.1**
- Login Username: **khawar**
- Password: **cisco**
- Interface: Loopback99
- IP Address: 99.99.99.99/8

#### Admin PC

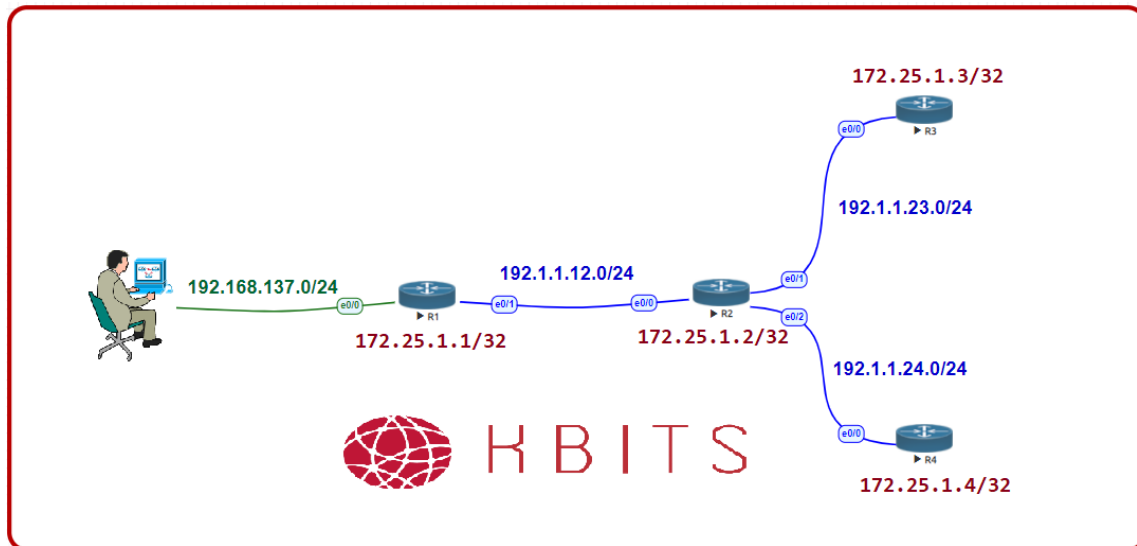
```
from telnetlib import Telnet

ab = Telnet('172.25.1.1')
ab.write(b'khawar\n')
ab.write(b'cisco\n')
ab.write(b'config t\n')
ab.write(b'Interface Loopback99\n')
ab.write(b'ip address 99.99.99.99 255.0.0.0\n')
ab.write(b'end\n')
ab.write(b'sh ip int brief\n')
ab.write(b'exit\n')
print (ab.read_all().decode('ascii'))
```

#### Verification:

- Save the script as Lab4.py.
- Run the script.
- Verify the output on the console.

# Lab 5 – Configuring Network Devices Using Python – Interactive Config



## Task 1

Configure a Python Script that allows you to input the Interface and IP Address that needs to be configured on R1. Display the Interface that was created by using the “**Show ip interface** brief” command. Use the following to write the script.

- Use the “Telnet” function from the **telnetlib** library.
- Create a variable called **Interface** to receive the interface name using the “input” function.
- Create a variable called **Ipaddr** to receive the IP Address to be configured using the “input” function.
- Create a variable called **SMask** to receive the Subnet mask using the “input” function.
- Host IP: **172.25.1.1**
- Login Username: **khawar**
- Password: **cisco**

### Admin PC

```
from telnetlib import Telnet

Interface = input('What Interface would you like to configure : ')
Ipaddr = input('Specify the IP Address : ')
SMask = input('Specify the Subnet mask : ')

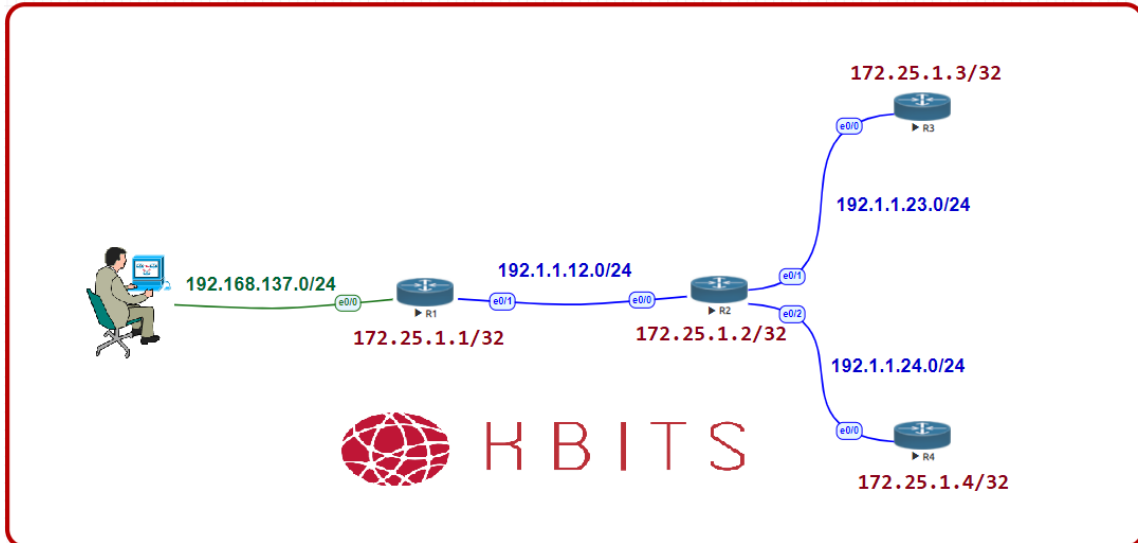
ab = Telnet('172.25.1.1')
ab.write(b'khawar\n')
ab.write(b'cisco\n')
ab.write(b'config t\n')
ab.write(b'Interface ' + Interface.encode('ascii') + b'\n')
ab.write(b'IP Address ' + Ipaddr.encode('ascii') + b' ' + SMask.encode('ascii') + b'\n')
ab.write(b'end\n')
ab.write(b'sh ip int brief\n')
ab.write(b'exit\n')
print (ab.read_all().decode('ascii'))
```

### **Verification:**

- Save the script as Lab5.py.
- Run the script. Specify the following:
  - Interface Name: **Loopback55**
  - IP Address: **55.1.1.1**
  - Subnet Mask: **255.0.0.0**
- Verify the output on the console.



# Lab 6 – Configuring Network Devices Using Python – Interactive Login & Configuration



## Task 1

Configure a Python Script that allows you to input the Interface and IP Address that needs to be configured on a Host that you specify. Allow the user to specify the Username and password as well. Display the Interface that was created by using the “**Show ip interface brief**” command. Use the following to write the script.

- Use the “Telnet” function from the **telnetlib** library.
- Create a variable called **HOST** to receive the IP Address of the host using the “input” function.
- Create a variable called **USER** to receive the Username using the “input” function.
- Use the “getpass” function to retrieve the password and store it in a variable called **PASS**. Import the getpass function.
- Create a variable called **Interface** to receive the interface name using the “input” function.
- Create a variable called **Ipaddr** to receive the IP Address to be configured using the “input” function.
- Create a variable called **SMask** to receive the Subnet mask using the “input” function.

### Admin PC

```
from telnetlib import Telnet
import getpass

HOST = input('Specify the Hostname : ')
USER = input('Specify the Username: ')
PASS = getpass.getpass()
Interface = input('What Interface would you like to configure : ')
Ipaddr = input('Specify the IP Address : ')
SMask = input('Specify the Subnet mask : ')

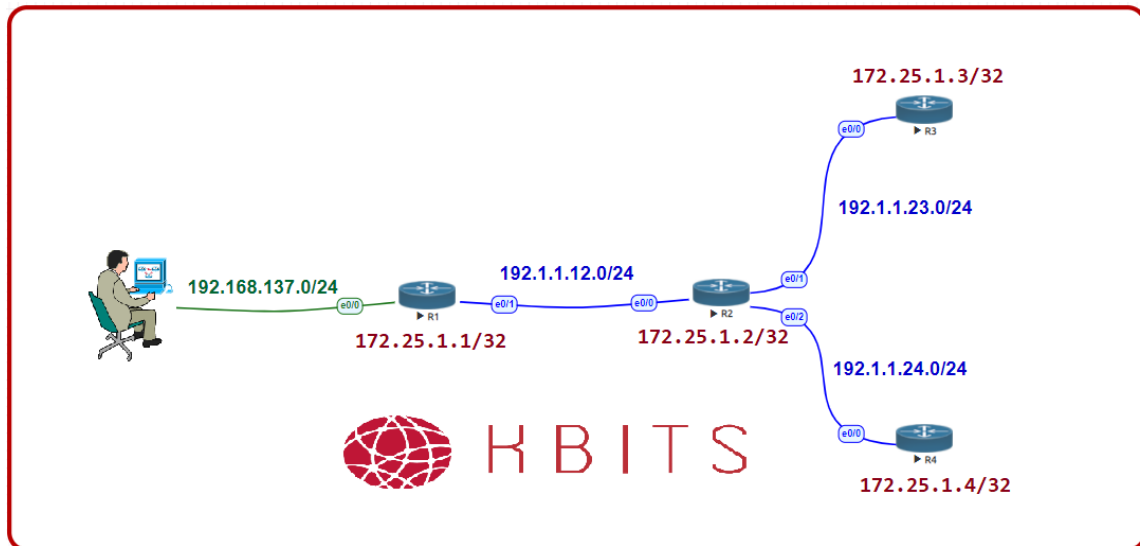
ab = Telnet(HOST)
ab.write(USER.encode('ascii') + b'\n')
ab.write(PASS.encode('ascii') + b'\n')

ab.write(b'config t\n')
ab.write(b'Interface ' + Interface.encode('ascii') + b'\n')
ab.write(b'IP Address ' + Ipaddr.encode('ascii') + b' ' + SMask.encode('ascii') + b'\n')
ab.write(b'end\n')
ab.write(b'sh ip int brief\n')
ab.write(b'exit\n')
print (ab.read_all().decode('ascii'))
```

### Verification:

- Save the script as Lab6.py.
- Run the script. Use Debug mode if you are using PyCharm as it has an issue with the `getpass()` function.
- Specify the following:
  - Hostname: **172.25.1.2**
  - Username: **khawar**
  - Password: **cisco**
  - Interface Name: **Loopback55**
  - IP Address: **55.2.2.2**
  - Subnet Mask: **255.0.0.0**
- Verify the output on the console.

# Lab 7 – Initialize the Router using a Python Script – Netmiko Library



### Task 1

Configure a Python Script that allows you to set a banner on R1 using SSH. Also configure the “logging sync” command for the Console line. Display the banner within the running-config using the “**Show running | inc banner**” command. Use the following to write the script.

- Use the “ConnectHandler” function from the **netmiko** library.
- Host IP: **172.25.1.1**
- Login Username: **khawar**
- Password: **cisco**

#### Admin PC

```
from netmiko import ConnectHandler

ABC = {
    'device_type': 'cisco_ios',
    'host': 172.25.1.1,
    'username': 'khawar',
    'password': 'cisco',
}
MYSSH = ConnectHandler(**ABC)

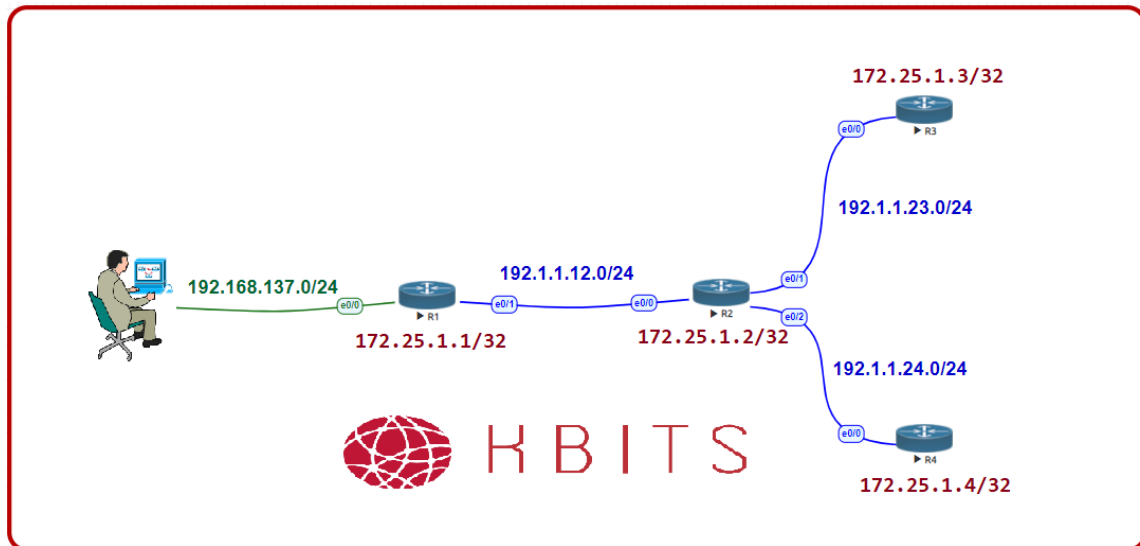
config_commands = [ 'banner motd #Authorized KBITS.LIVE Users Only#',
                    'line con 0',
                    'logg sync'
                  ]

output = MYSSH.send_config_set(config_commands)
output = MYSSH.send_command('show runn | inc banner')
print(output)
```

#### Verification:

- Save the script as Lab7.py.
- Run the script.
- Verify the output on the console.

# Lab 8 – Initialize the Router using a Python Script – Netmiko Library (Interactive)



## Task 1

Configure a Python Script that allows you specify the Host that you would like to Initialize. Allow the user to specify the Username and password as well. Display the banner that was created by using the **“Show running | inc banner”** command. Use the following to write the script.

- Use the “ConnectHandler” function from the **netmiko** library.
- Create a variable called **HOST** to receive the IP Address of the host using the “input” function.
- Create a variable called **user** to receive the Username using the “input” function.
- Use the “getpass” function to retrieve the password.
- Configure the following commands within the script:

```
banner motd #Authorized KBITS.LIVE Users Only#  
line con 0  
logg sync  
no exec-timeout
```

### Admin PC

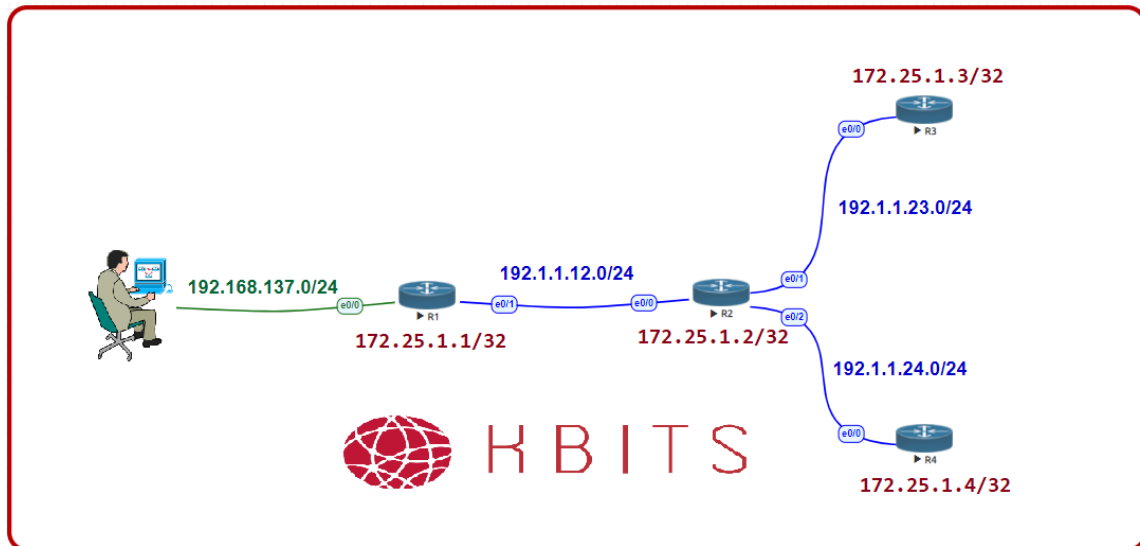
```
from netmiko import ConnectHandler  
from getpass import getpass  
  
HOST = input("Enter Hostname: ")  
user = input("Enter your SSH username: ")  
  
ABC = {  
    'device_type': 'cisco_ios',  
    'host': HOST,  
    'username': user,  
    'password': getpass(),  
    'port' : 22,          # optional, defaults to 22  
    'secret': 'cisco',   # optional, defaults to "  
}  
myconnect = ConnectHandler(**ABC)  
  
config_commands = [ 'banner motd #Authorized KBITS.LIVE Users Only#',  
                    'line con 0',  
                    'logg sync',  
                    'no exec-timeout' ]  
output = myconnect.send_config_set(config_commands)  
output = myconnect.send_command('show runn | inc banner')  
print(output)
```

### Verification:

- Save the script as Lab8.py.
- Run the script. Use Debug mode if you are using PyCharm as it has an issue with the `getpass()` function.
- Specify the following:
  - Hostname: **172.25.1.3**
  - Username: **khawar**
  - Password: **cisco**
- Verify the output on the console.



# Lab 9 – Retrieving Information from Multiple Routers – Netmiko Library



### Task 1

Configure a Python Script that allows you to retrieve the Interfaces and their status from a list of devices specified in a text file. Display the interfaces to the console using the “**Show ip interface brief**” command. Use the following to write the script.

- Use the “ConnectHandler” function from the **netmiko** library.
- Create a file called devices.txt with the loopback10000 IP Addresses for R1 – R4.
- Login Username: **khawar**
- Password: **cisco**

#### Content of the “devices.txt” File

```
172.25.1.1
172.25.1.2
172.25.1.3
172.25.1.4
```

#### Admin PC

```
from netmiko import ConnectHandler

with open('devices.txt') as routers:
    for IP in routers:
        Router = {
            'device_type': 'cisco_ios',
            'ip': IP,
            'username': 'khawar',
            'password': 'cisco'
        }

        net_connect = ConnectHandler(**Router)

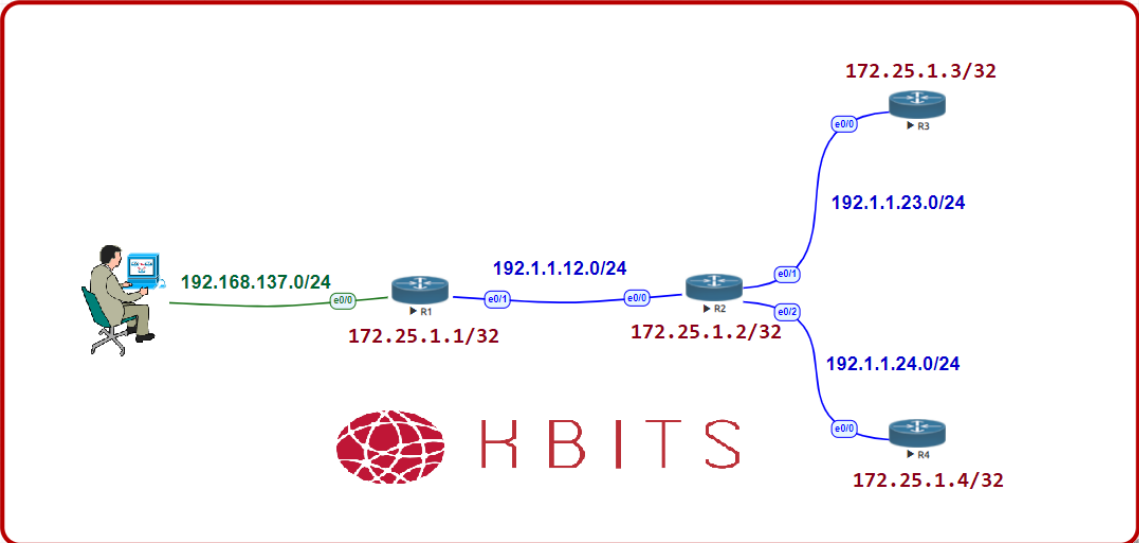
        print('Connecting to ' + IP)
        print('-' * 79)
        output = net_connect.send_command('sh ip int brief')
        print(output)
        print()
        print('-' * 79)

net_connect.disconnect()
```

**Verification:**

- Save the script as Lab9.py.
- Run the script.
- Verify the output on the console.

# Lab 10 – Backing up Configuration of a single Router – Netmiko Library



### Task 1

Configure a Python Script that allows you to retrieve the running config file of R1 and store it on your PC. Use the following to write the script.

- Use the “ConnectHandler” function from the **netmiko** library.
- Host: 172.25.1.1
- Login Username: **khawar**
- Password: **cisco**

#### Admin PC

```
from netmiko import ConnectHandler

ROUTER = {
    'device_type': 'cisco_ios',
    'ip': '172.25.1.1',
    'username': 'khawar',
    'password': 'cisco'
}

net_connect = ConnectHandler(**ROUTER)

hostname = net_connect.send_command('show run | i host')
hostname.split(" ")
hostname,device = hostname.split(" ")
print ("Backing up " + device)

filename = device + '.txt'

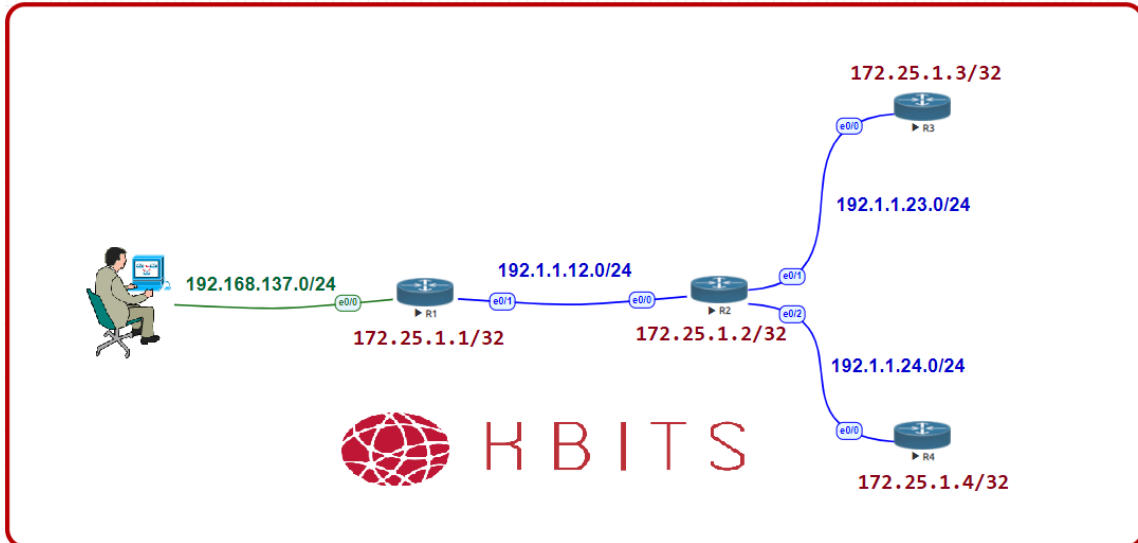
showrun = net_connect.send_command('show run')
log_file = open(filename, "w")
log_file.write(showrun)
log_file.write("\n")

net_connect.disconnect()
```

**Verification:**

- Save the script as Lab10.py.
- Run the script.
- Verify that the file R1.txt is created in your folder.

# Lab 11 – Backing up Configuration of Multiple Routers – Netmiko Library



### Task 1

Configure a Python Script that allows you to backup the running configuration of a list of devices specified in a text file. It should store the configurations using the Device Name of the Router.

- Use the “ConnectHandler” function from the **netmiko** library.
- Host: Use the devices.txt created in Lab 9
- Login Username: **khawar**
- Password: **cisco**

#### Content of the “devices.txt” File

```
172.25.1.1  
172.25.1.2  
172.25.1.3  
172.25.1.4
```

#### Admin PC

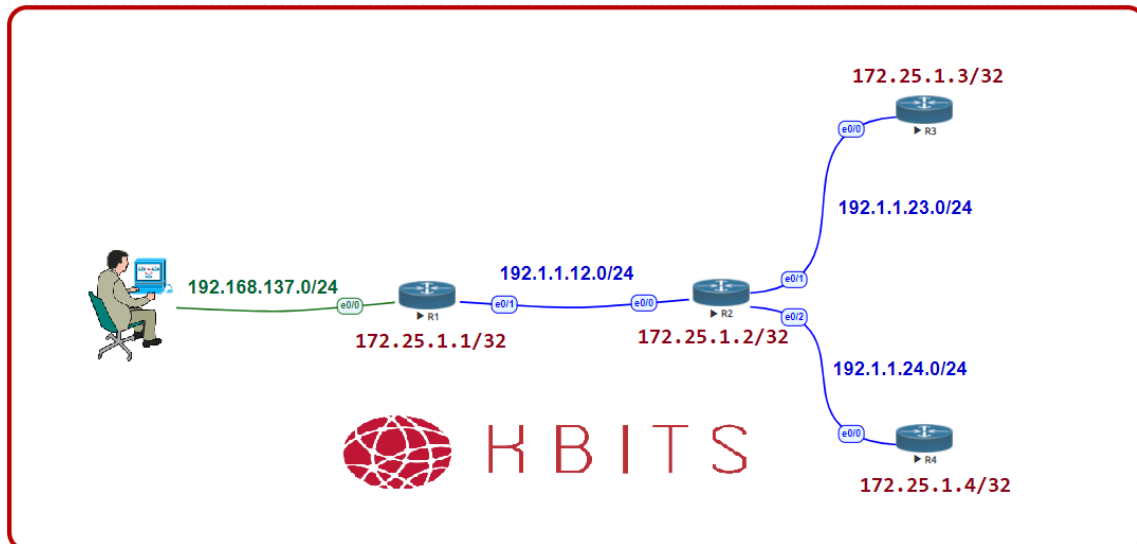
```
from netmiko import ConnectHandler  
  
with open('devices.txt') as routers:  
    for IP in routers:  
        Router = {  
            'device_type': 'cisco_ios',  
            'ip': IP,  
            'username': 'khawar',  
            'password': 'cisco'  
        }  
  
        net_connect = ConnectHandler(**Router)  
  
        hostname = net_connect.send_command('show run | i host')  
        hostname.split(" ")  
        hostname,device = hostname.split(" ")  
        print ("Backing up " + device)  
  
        filename = device + '-Backup.txt'  
  
        showrun = net_connect.send_command('show run')  
        log_file = open(filename, "w")  
        log_file.write(showrun)  
        log_file.write("\n")  
  
net_connect.disconnect()
```



**Verification:**

- Save the script as Lab11.py.
- Run the script.
- Verify that the files are created for all the routers in your folder.

# Lab 12 – Configuring Multiple Devices – Netmiko Library



## Task 1

Configure a Python Script that allows you to configure a set of devices using configuration files.

- Use the “ConnectHandler” function from the **netmiko** library.
- Host: Use the devices.txt created in Lab 9.
- Create the configuration files for the routers based on the table below.
- Login Username: **khawar**
- Password: **cisco**

### Content of the “devices.txt” File

```
172.25.1.1
172.25.1.2
172.25.1.3
172.25.1.4
```

### Content of the “R1.txt” File

```
config terminal
!
interface loo11
 ip address 150.1.1.1 255.255.255.255
!
banner motd "Authorized KBITS Users Only!!!!!!!!!!!"
!
Wr
```

### Content of the “R2.txt” File

```
config terminal
!
interface loo11
 ip address 150.1.1.2 255.255.255.255
!
banner motd "Authorized KBITS Users Only!!!!!!!!!!!"
!
Wr
```

### Content of the “R3.txt” File

```
config terminal
!
interface loo11
 ip address 150.1.1.3 255.255.255.255
!
banner motd "Authorized KBITS Users Only!!!!!!!!!!!"
```

```
!  
Wr
```

### **Content of the "R4.txt" File**

```
config terminal  
!  
interface loo11  
 ip address 150.1.1.4 255.255.255.255  
!  
banner motd "Authorized KBITS Users Only!!!!!!!!!!!"  
!  
wr
```

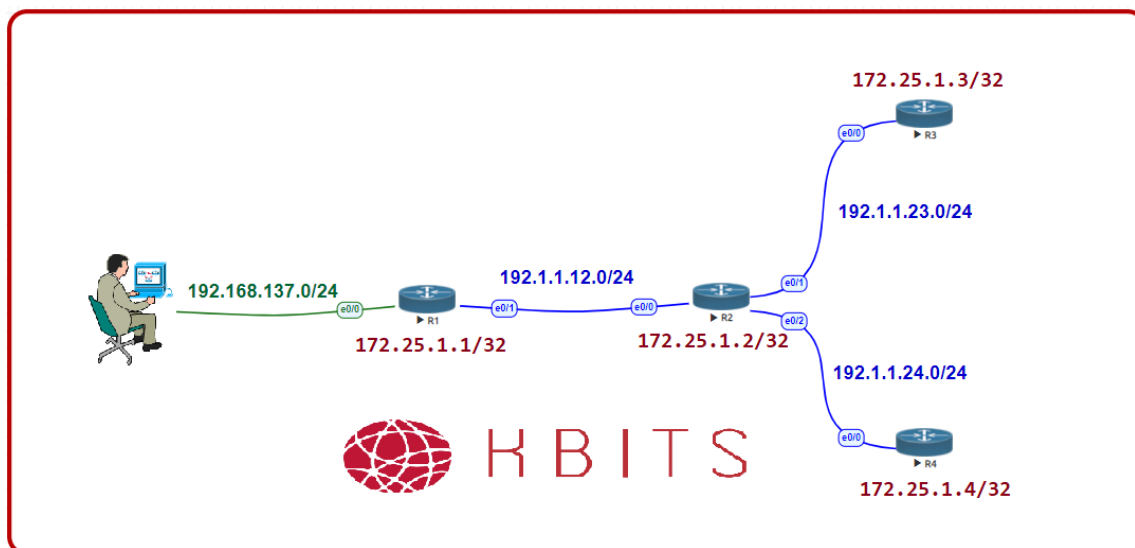
### **Admin PC**

```
from netmiko import ConnectHandler  
  
with open('devices.txt') as routers:  
    for IP in routers:  
        Router = {  
            'device_type': 'cisco_ios',  
            'ip': IP,  
            'username': 'khawar',  
            'password': 'cisco'  
        }  
        net_connect = ConnectHandler(**Router)  
        hostname = net_connect.send_command('show run | i host')  
        hostname.split(" ")  
        hostname, devicename = hostname.split(" ")  
        cmdfile=devicename + ".txt"  
        net_connect.send_config_from_file(cmdfile)  
        print(devicename + ' Configured')  
        net_connect.disconnect()
```

**Verification:**

- Save the script as Lab12.py.
- Run the script.
- Verify the configuration on the Routers.

# Lab 13 – Configuring Multiple Devices – Netmiko Library (Interactive)



### Task 1

Configure a Python Script that allows you to configure a set of devices using configuration files.

- Use the “ConnectHandler” function from the **netmiko** library.
- Allow the User to input the name of the devices file along with the SSH Username and Password.
- Create the configuration files for the routers based on the table below.
- Login Username: **khawar**
- Password: **cisco**

#### Content of the “devices.txt” File

```
172.25.1.1  
172.25.1.2  
172.25.1.3  
172.25.1.4
```

#### Content of the “R1.txt” File

```
config terminal  
!  
mpls ldp router-id loopback0  
!  
Interface E 0/1  
  mpls ip  
!  
wr
```

#### Content of the “R2.txt” File

```
config terminal  
!  
mpls ldp router-id loopback0  
!  
Interface E 0/0  
  mpls ip  
!  
Interface E 0/1  
  mpls ip  
!  
Interface E 0/2  
  mpls ip  
!  
wr
```

#### Content of the “R3.txt” File

```
config terminal
!
mpls ldp router-id loopback0
!
Interface E 0/0
 mpls ip
!
wr
```

### **Content of the "R4.txt" File**

```
config terminal
!
mpls ldp router-id loopback0
!
Interface E 0/0
 mpls ip
!
wr
```

### **Admin PC**

```
from netmiko import ConnectHandler
import getpass

print('Please name the Configuration File based on the Hostname.txt format\n')
HOSTS = input("Enter the name of file for Device List: ")
user = input("Enter your SSH username: ")
PASS = getpass.getpass()
print('\n')

with open(HOSTS) as routers:
    for IP in routers:
        Router = {
            'device_type': 'cisco_ios',
            'ip': IP,
            'username': user,
            'password': PASS,
            'port': 22
        }

        net_connect = ConnectHandler(**Router)
        hostname = net_connect.send_command('show run | i host')
        hostname.split(" ")
        hostname, devicename = hostname.split(" ")
        cmdfile=devicename + ".txt"
```



```
net_connect.send_config_from_file(cmdfile)
print(devicename + ' Configured')
net_connect.disconnect()
```

**Verification:**

- Save the script as Lab13.py.
- Run the script. Use Debug mode if you are using PyCharm as it has an issue with the `getpass()` function.
- Verify the configuration on the Routers.