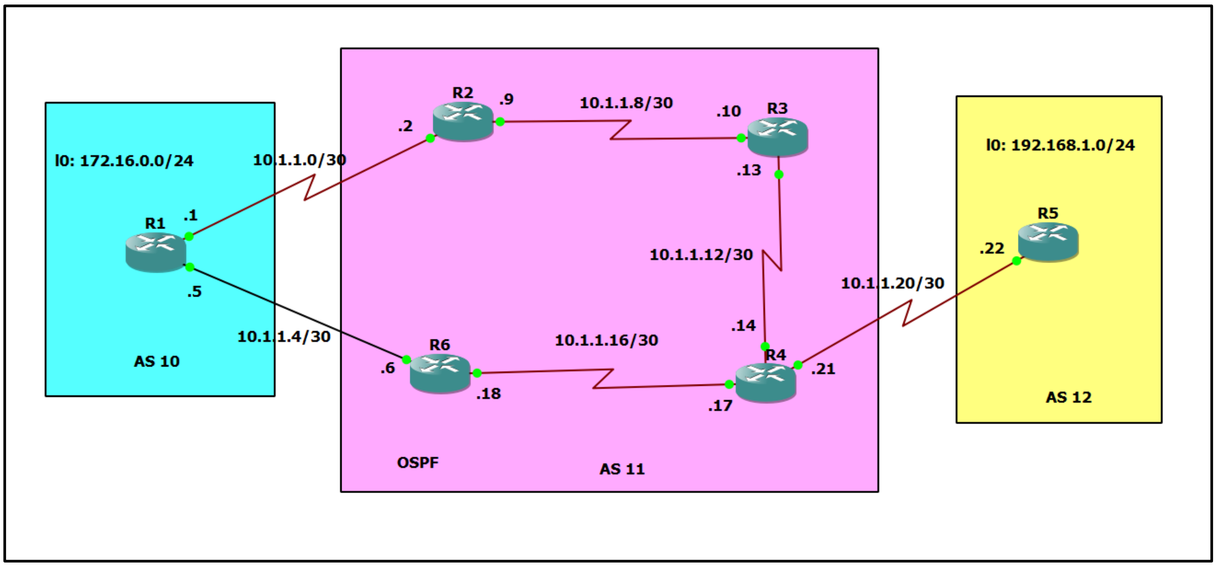
BGP Multi-AS Routing with Primary Path Selection using Weight Attribute



Objective:

The objective of this project is to design and implement a multi-Autonomous System (AS) routing environment using OSPF as the Interior Gateway Protocol (IGP) within the internal AS and eBGP for inter-AS communication. The project aims to establish reliable end-to-end connectivity between separate autonomous systems, ensure proper route advertisement and propagation across AS boundaries, and analyze BGP best path selection behavior. Additionally, the project demonstrates traffic engineering by manipulating the BGP Weight attribute to control outbound path preference and evaluate routing decisions before and after policy modification.

Routing Design Explanation

Routing Design

This network topology consists of three Autonomous Systems:

- **AS10** – Represents the source network (Router1).
- **AS11** – Acts as the transit Autonomous System containing internal routers (Router2, Router3, Router4, and Router6).
- **AS12** – Represents the destination network (Router5).

Internal Routing (AS11)

Within AS11, **OSPF** is implemented as the Interior Gateway Protocol (IGP) to provide dynamic routing between internal routers. OSPF ensures:

- Fast convergence
- Efficient internal route propagation

- Scalable intra-domain routing

All internal point-to-point links within AS11 are advertised in OSPF Area 0.

Inter-AS Routing

eBGP is configured between:

- AS10 and AS11
- AS11 and AS12

BGP is responsible for:

- Advertising loopback networks (172.16.0.0/24 and 192.168.1.0/24)
- Exchanging routes between Autonomous Systems
- Maintaining policy-based routing control

iBGP and Next-Hop Handling

Within AS11, iBGP sessions are configured between internal routers to ensure proper propagation of external routes. The **next-hop-self command is applied on Router4** to prevent next-hop reachability issues and ensure consistent route advertisement within the AS.

Traffic Engineering

To demonstrate BGP path manipulation, the **Weight attribute** is modified on Router1. By assigning a higher weight to one neighbor, Router1 is forced to prefer that path as the primary outbound route. This validates the BGP best path selection process and demonstrates policy-based routing control.

Configuring Router

Router1:

R1# conf t
R1(config)# int l0
R1(config-if)# ip addr 172.16.0.1 255.255.255.0
R1(config-if)# int s1/0
R1(config-if)# ip addr 10.1.1.1 255.255.255.252
R1(config-if)# no shut
R1(config-if)# int f0/0
R1(config-if)# ip addr 10.1.1.5 255.255.255.252
R1(config-if)# no shut
ctrl + z
wr

Router2:

R2# conf t
R2(config)# int s1/0
R2(config-if)# ip addr 10.1.1.2 255.255.255.252

R2(config-if)# no shut
R2(config-if)# int s1/1
R2(config-if)# ip addr 10.1.1.9 255.255.255.252
R2(config-if)# no shut
ctrl + z
wr

Router3:
R3# conf t
R3(config)# int s1/0
R3(config-if)# ip addr 10.1.1.10 255.255.255.252
R3(config-if)# no shut
R3(config-if)# int s1/1
R3(config-if)# ip addr 10.1.1.13 255.255.255.252
R3(config-if)# no shut
ctrl + z
wr

Router4:
R4# conf t
R4(config)# int s1/0
R4(config-if)# ip addr 10.1.1.14 255.255.255.252
R4(config-if)# no shut
R4(config-if)# int s1/1
R4(config-if)# ip addr 10.1.1.17 255.255.255.252
R4(config-if)# no shut
R4(config-if)# int s1/2
R4(config-if)# ip addr 10.1.1.21 255.255.255.252
R4(config-if)# no shut
ctrl + z
wr

Router5:
R5# conf t
R5(config)# int l0
R5(config-if)# ip addr 192.168.1.1 255.255.255.0
R5(config-if)# int s1/0
R5(config-if)# ip addr 10.1.1.22 255.255.255.252
R5(config-if)# no shut
ctrl + z
wr

Router6:

R6# conf t
R6(config)# int s1/0
R6(config-if)# ip addr 10.1.1.18 255.255.255.252
R6(config-if)# no shut
R6(config-if)# int f1/1
R6(config-if)# ip addr 10.1.1.6 255.255.255.252
R6(config-if)# no shut
ctrl + z
wr

OSPF Configuration

Router2:
R2# conf t
R2(config)# router ospf 1
R2(config-router)# router-id 2.2.2.2
R2(config-router)# network 10.1.1.0 0.0.0.3 area 0
R2(config-router)# network 10.1.1.8 0.0.0.3 area 0
ctrl+z
wr

Router3:
R3# conf t
R3(config)# router ospf 1
R3(config-router)# router-id 3.3.3.3
R3(config-router)# network 10.1.1.8 0.0.0.3 area 0
R3(config-router)# network 10.1.1.12 0.0.0.3 area 0
ctrl+z
wr

Router4:
R4# conf t
R4(config)# router ospf 1
R4(config-router)# router-id 4.4.4.4
R4(config-router)# network 10.1.1.12 0.0.0.3 area 0
R4(config-router)# network 10.1.1.16 0.0.0.3 area 0
ctrl+z
wr

Router6:
R6# conf t
R6(config)# router ospf 1
R6(config-router)# router-id 6.6.6.6

R6(config-router)# network 10.1.1.16 0.0.0.3 area 0
R6(config-router)# network 10.1.1.4 0.0.0.3 area 0
ctrl+z
wr

BGP Configuration

Router1:
R1# conf t
R1(config)# router bgp 10
R1(config-router)# neighbor 10.1.1.2 remote-as 11
R1(config-router)# neighbor 10.1.1.6 remote-as 11
R1(config-router)# network 172.16.0.0 mask 255.255.255.0
ctrl+z
wr

Router2:
R2# conf t
R2(config)# router bgp 11
R2(config-router)# neighbor 10.1.1.1 remote-as 10
R2(config-router)# neighbor 10.1.1.14 remote-as 11
ctrl+z
wr

Router4:
R4# conf t
R4(config)# router bgp 11
R4(config-router)# neighbor 10.1.1.9 remote-as 11
R4(config-router)# neighbor 10.1.1.22 remote-as 12
R4(config-router)# neighbor 10.1.1.18 remote-as 11
ctrl+z
wr

Router5:
R5# conf t
R5(config)# router bgp 12
R5(config-router)# neighbor 10.1.1.21 remote-as 11
R5(config-router)# network 192.168.1.0 mask 255.255.255.0
ctrl+z
wr

Router6:
R6# conf t

R6(config)# router bgp 11
R6(config-router)# neighbor 10.1.1.5 remote-as 10
R6(config-router)# neighbor 10.1.1.17 remote-as 11
ctrl+z
wr

Router4:
R4# conf t
R4(config)# router bgp 11
R4(config-router)# neighbor 10.1.1.9 next-hop-self
R4(config-router)# neighbor 10.1.1.18 next-hop-self
ctrl+z
wr

Verification (Before weight change)

R1# sh ip bgp

```
R1#sh ip bgp
BGP table version is 3, local router ID is 172.16.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  172.16.0.0/24    0.0.0.0                  0         32768 i
 *   192.168.1.0      10.1.1.2                               0 11 12 i
 *>                   10.1.1.6                               0 11 12 i
```

Weight changing configuration

R1# conf t
R1(config)# router bgp 10
R1(config-router)# neighbor 10.1.1.6 weight 1000
R1(config-router)# do clear ip bgp *
ctrl+z
wr

Verification (After weight change)

R1# sh ip bgp

```
R1#sh ip bgp
BGP table version is 3, local router ID is 172.16.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  172.16.0.0/24    0.0.0.0                  0         32768 i
 *>  192.168.1.0      10.1.1.6                            1000 11 12 i
 *                    10.1.1.2                               0 11 12 i
```

Before weight modification, Router1 selected the path via 10.1.1.2 as the best route. After assigning weight 1000 to neighbor 10.1.1.6, Router1 preferred that path as the primary outbound route.

Troubleshooting & Challenges Faced

Packet Tracer Limitation (iBGP Not Supported)
One of the major challenges faced during this project was the limitation of Cisco Packet Tracer, which does not support internal BGP (iBGP) in certain versions. Initially, attempts to configure iBGP sessions resulted in errors stating that internal neighbors were not supported.

To overcome this limitation, the project was migrated to GNS3, which provides full BGP functionality and behaves closer to real Cisco IOS environments.

Confusion Between Routing Table and BGP Table
Another challenge encountered was understanding the different outputs between:

- show ip route (Routing Table)
- show ip bgp (BGP Table)

In several instances, a network was visible in the routing table but not in the BGP table. This caused routes not to propagate to other Autonomous Systems.

The project was migrated to GNS3 to solve the problem with iBGP configuration.

Next-Hop Reachability Issue
During iBGP configuration, some routes were learned but not installed due to next-hop reachability problems. This was resolved by applying: **neighbor <IP> next-hop-self** on the appropriate router to ensure proper next-hop handling within the Autonomous System.

Understanding BGP Weight Behavior
There was initial confusion regarding the Weight attribute, particularly whether it affected link speed or traffic load balancing. Through testing and verification, it was confirmed that:

- Weight only influences outbound path selection.
- Higher weight makes a path more preferred.
- It does not affect bandwidth or congestion handling.

Verification using show ip bgp before and after modifying the weight confirmed correct path selection behavior.

Conclusion:

This project successfully demonstrated multi-AS BGP routing integrated with OSPF and validated BGP path selection behavior using the Weight attribute. The lab strengthened understanding of inter-domain routing, next-hop handling, redistribution concepts, and troubleshooting techniques in a realistic network environment.