	NATIONAL UNIVERSITY OF COMPUTER & EMERGING SCIENCE	
	Computer Network Lab (CL-307)	
Experiment Title: BGP (Border Gateway Protocol)		**Lab No:** 11

Objective:

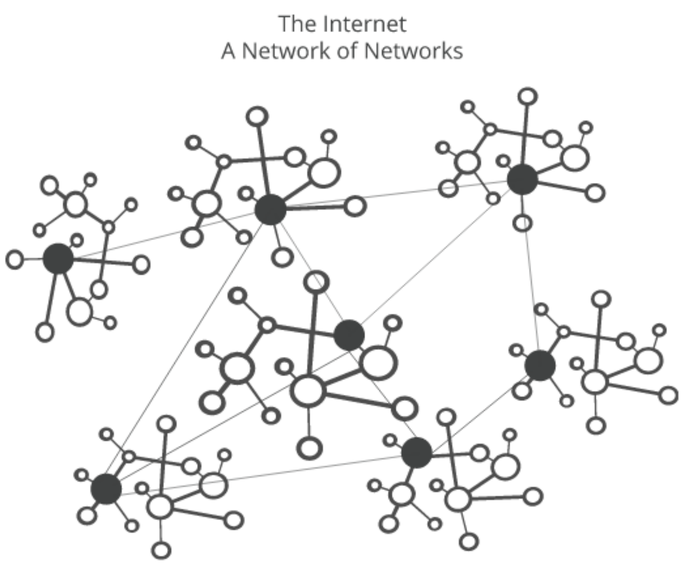
- ✓ Network layer protocol Implementation
    - Understand and implementation of BGP
    - Understanding the reason why BGP is used
    - Understand the working mechanizing of BGP

Border Gateway Protocol (BGP)

The Border Gateway Protocol (BGP) is a standardized exterior gateway protocol (EGP) designed to exchange routing and reachability information between autonomous systems (AS) on the Internet, which are networks managed by a single enterprise or service provider. The protocol is often classified as a path vector protocol but is sometimes also classed as a distance-vector routing protocol. BGP does not advertise incremental updates or refresh network advertisements like OSPF or ISIS. BGP prefers stability within the network, because a link flap could result in route computation for thousands of routes.

✓ Autonomous Systems (AS)

The Internet is a network of networks; it's broken up into hundreds of thousands of smaller networks known as autonomous systems (AS). Each of these networks is essentially a large pool of routers run by a single organization. From the perspective of BGP, an autonomous system (AS) is a collection of routers under a single organization's control, using one or more IGPs, and common metrics to route packets within the AS.



✓ Who operates BGP autonomous systems?

Autonomous systems typically belong to ISPs or other large high-tech organizations, such as tech companies, universities, government agencies, and scientific institutions. Each autonomous system wishing to exchange routing information must have a registered autonomous system number (ASN). Internet Assigned Numbers Authority (IANA) assigns ASNs to Regional Internet Registries (RIRs), which then assigns them to ISPs and networks. ASNs are 16 bit numbers between 1 and 65534 and 32 bit numbers between 131072 and 4294967294. As of 2018, there are approximately 64,000 ASNs in-use worldwide. These ASNs are only required for external BGP.

✓ BGP Categorization

1. **External Border Gateway Protocol (EBGP):**
   EBGP is used between autonomous systems. It is used and implemented at the edge or border router that provides inter-connectivity for two or more autonomous system. It functions as the protocol responsible for interconnection of networks from different organizations or the Internet.
2. **Internal Border Gateway Protocol (IBGP):**
   IBGP is used inside the autonomous systems. It is used to provide information to your internal routers. It requires all the devices in same autonomous systems to form full mesh topology or either of Route reflectors and Confederation for prefix learning.

✓ How does BGP works?

BGP uses TCP as its transport protocol (port 179). Two BGP speaking routers form a TCP connection between one another router typically know as peer routers, neighbor or speaker and exchange messages to open and confirm the connection parameters.

BGP routers will exchange network reachability information, this information is mainly an indication of the full paths (BGP AS numbers) that a route should take in order to reach the destination network. This information will help in constructing a graph of ASs that are loop free and where routing policies can be applied in order to enforce some restrictions on the routing behavior.

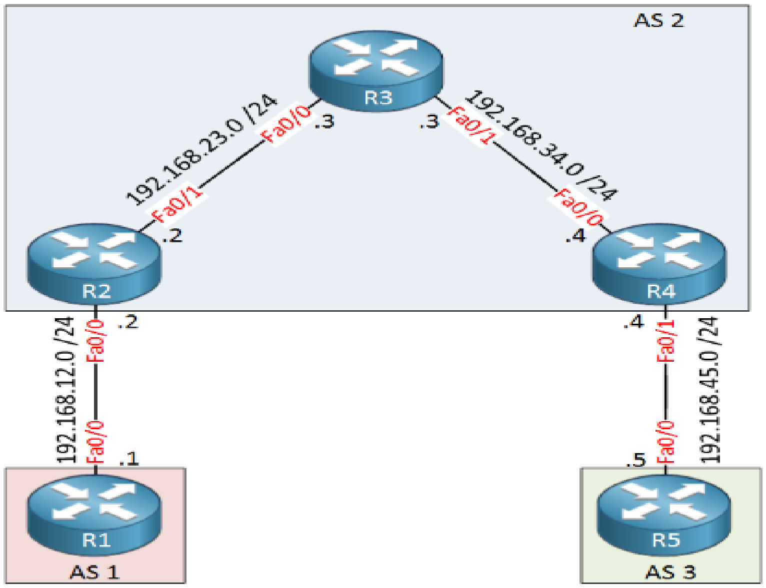
- **Let’s draw an analogy from everyday life to make all of this a bit more understandable. (BGP is Just like GPS for Packets).**

  You can think of an autonomous system in the computer world as a city with many streets. A network prefix is similar to one street with many houses. An IP address is like an address for a particular house in the real world, while a packet is the equivalent of a car travelling from one house to another using the best possible route.

  Taking this comparison to its logical conclusion, the BGP routing protocol is analogous to your trusty GPS navigator. Like Google’s Waze application, the best route is determined by different factors, such as traffic congestion, roads temporarily closed for maintenance, etc. The path is calculated dynamically depending on the situation of the network nodes, which are like roads and junctions on a GPS map.

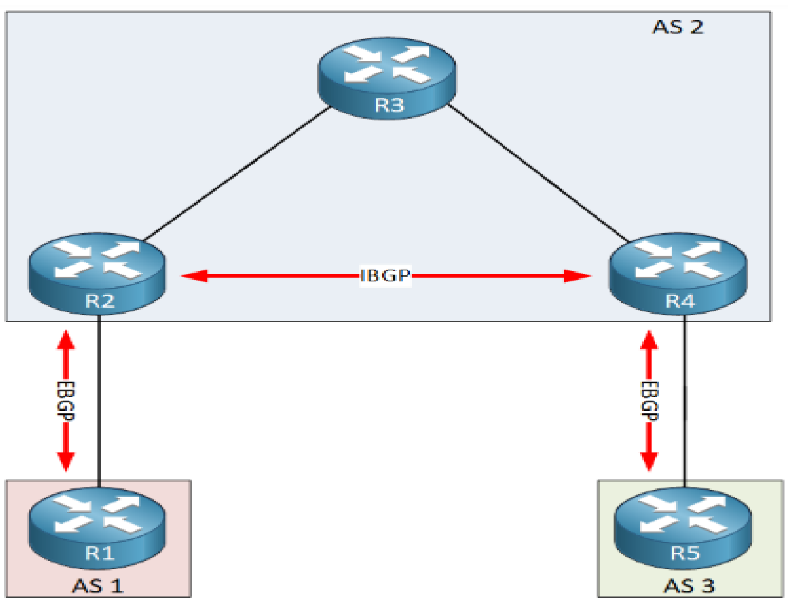
✓ Let's Get Technical

Now that the background is (hopefully) clearer, let's start with an example topology and I'll explain a couple of things:



Above you see 3 autonomous systems and 5 routers. When AS1 wants to reach AS3 we have to cross AS2, this makes AS2 our transit AS. This is a typical scenario where AS1 and AS3 are customers and AS2 is the ISP.

In our scenario, AS1 has a loopback interface (a logical, virtual interface) with network 1.1.1.0 /24 and AS3 wants to reach this network. This means we'll have to advertise this network through BGP. Here's what it looks like:



<u>So what is going on here? Let me explain it step-by-step:</u>

We need EBGP between AS1 and AS2 because these are two different autonomous systems. This allows us to advertise a prefix on R1 in BGP so that AS2 can learn it.

We also need EBGP between AS2 and AS3 so that R5 can learn prefixes (composed of a path of AS numbers, indicating which networks the packet must pass through, and IP the block that is being routed) through BGP.

We need to get the prefix that R2 learned from R1 somehow to R5. We do this by configuring IBGP between R2 and R4, this allows R4 to advertise it to R5.

So that's the first reason why we need IBGP, so you can **advertise a prefix from one autonomous system to another**. You might have a few questions after reading this:

Why don't we use OSPF (or EIGRP) on AS2 instead and redistribute the prefix on R2 from BGP into OSPF and on R4 from OSPF back into BGP?

- Doesn't IBGP have to be directly connected?
- How are R2 and R4 able to reach each other through IBGP if we don't have any routing protocol within AS2?
- What about R3? Do we need IBGP?

These are some of the questions we get all the time, here are the answers:

Technically this is possible… we can run OSPF (or EIGRP) within AS2 and use redistribution between BGP and OSPF. In our example R1 will only have a single prefix so it's no problem but what if R1 had a full internet routing table? (Over 500.000 prefixes (2014)). IGPs like OSPF or EIGRP are not able to handle that many prefixes so you'll need BGP for this.

IBGP does not have to be directly connected, this might be a little confusing when you only know about OSPF or EIGRP since they always form adjacencies on directly connected links.

They are not! This is why we **need an IGP within the AS**. Since R2 and R4 are not directly connected we'll configure an IGP so that they can reach each other.

Let's see what will go wrong if we don't configure R3.

Enough reading for now, let's get some configuration. We'll start with BGP between R1/R2, R2/R4 and R4/5 like we just described.

✓ Configuration

First we'll configure R1 and R2. I am also advertising a prefix (on a loopback interface) in BGP:

```
R1(config)#interface loopback 0
R1(config-if)#ip address 1.1.1.1 255.255.255.0
R1(config-router)#neighbor 192.168.12.2 remote-as 2
R1(config-router)#network 1.1.1.0 mask 255.255.255.0
```

```
R2(config-router)#neighbor 192.168.12.1 remote-as 1
```

Our next step will be to configure IBGP between R2 and R4. What IP addresses are we going to use for this? Let's look at our options:

```
R2#show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        192.168.12.2    YES NVRAM  up                    up
FastEthernet1/0        192.168.23.2    YES NVRAM  up                    up
```

```
R4#show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        192.168.34.4    YES NVRAM  up                    up
FastEthernet1/0        192.168.45.4    YES NVRAM  up                    up
```

We can use any of these IP addresses but we need connectivity. That's why we need an IGP like we talked about earlier. So which IP addresses will we select? In this particular scenario it really doesn't matter since there is only 1 path between R2 and R4. What if we had multiple paths between R2 and R4?

When there are **multiple paths it's better to use a loopback interface** with an IP address and to advertise that into your IGP. We will use the loopback interface as the source for our BGP session. Why?

A physical interface can go down which means the IP address on the interface is no longer reachable. A **loopback interface will never go down** unless the router crashes or when you "shut" it. This is why it's best practice to use loopback interfaces when configuring IBGP.

We'll add a loopback interface on R2 and R4 and use these for IBGP, first we'll have to configure an IGP (We'll use OSPF) to advertise them:

```
R2(config)#interface loopback 0
R2(config-if)#ip address 2.2.2.2 255.255.255.0
```

```
R4(config)#interface loopback 0
R4(config-if)#ip address 4.4.4.4 255.255.255.0
```

That takes care of the loopback interfaces, now we can enable OSPF:

```
R2(config)#router ospf 1
R2(config-router)#network 192.168.23.0 0.0.0.255 area 0
R2(config-router)#network 2.2.2.0 0.0.0.255 area 0
```

```
R3(config)#router ospf 1
R3(config-router)#network 192.168.23.0 0.0.0.255 area 0
R3(config-router)#network 192.168.34.0 0.0.0.255 area 0
```

```
R4(config)#router ospf 1
R4(config-router)#network 192.168.34.0 0.0.0.255 area 0
R4(config-router)#network 4.4.4.0 0.0.0.255 area 0
```

R2 and R4 will now be able to reach each other's loopback interfaces. It's not a bad idea to test this though:

```
R2#ping 4.4.4.4 source 2.2.2.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 4.4.4.4, timeout is 2 seconds:
Packet sent with a source address of 2.2.2.2
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 48/52/60 ms
```

Alright we are now prepared for IBGP between R2 and R4. Here's what it looks like:

```
R2(config)#router bgp 2
R2(config-router)#neighbor 4.4.4.4 remote-as 2
R2(config-router)#neighbor 4.4.4.4 update-source loopback 0
```

```
R4(config)#router bgp 2
R4(config-router)#neighbor 2.2.2.2 remote-as 2
R4(config-router)#neighbor 2.2.2.2 update-source loopback 0
```

This takes care of our IBGP session. Note that we have to use the **update-source** command to specify that we will use the loopback interfaces as the source for the IBGP session.

Last but not least, let's configure EBGP between R4 and R5:

```
R4(config)#router bgp 2

R4(config-router)#neighbor 192.168.45.5 remote-as 3
```

```
R5(config)#router bgp 3

R5(config-router)#neighbor 192.168.45.4 remote-as 2
```

That takes care of that. Whenever you configure BGP you will see a message on the console that shows you that the neighbor adjacency has been established. You can also check it with the **show ip bgp summary** command.

✓ Verification

If everything went OK, all routers should have learned about the 1.1.1.0 /24 prefix that I advertised on R1. Let's see if that is true:

First we'll check R1:

```
R1#show ip bgp

BGP table version is 2, local router ID is 1.1.1.1

Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,

              r RIB-failure, S Stale

Origin codes: i - IGP, e - EGP, ? - incomplete


   Network          Next Hop            Metric LocPrf Weight Path

*> 1.1.1.0/24       0.0.0.0                  0         32768 i
```

✓ **<u>Some useful command for testing and troubleshooting:</u>**

- Check BGP (any other protocol) route from one router to another: **show ip route**
- Check BGP (any other protocol) protocol is configure or not: **show ip protocol**
- Show BGP status: **show bgp summary**
- Show bgp neighbor status: **show ip bgp neighbours**