

## Lab 9: BGP

*Or. BGP handles it all: IPv4 & IPv6 routing, Virtual Networks, Datacenters, and more*

### What You Will Do:

**Working in pairs**, combine your skills & knowledge to:

1. Leverage your knowledge of one routing protocol (OSPF) to understand another (BGP)
2. Learn "enough" about BGP from the following section (2.5 pages) to do this lab
3. Configure an eBGP session
4. Capture and examine output from show commands which reveal the state of BGP

### Things that you will need to know or learn:

1. All the details about BGP contained in the article in References (below).
2. All the details about configuring BGP contained in the 2.5 page info section (below).

### What you need to submit and when:

1. Complete the pre-lab quiz on BrightSpace, **before** the start of your lab period.
2. Complete the in-lab part of the exercise (see below), **before** the end of your lab period.
3. Complete post-lab exercise and quiz on BrightSpace, **before** your next lab.

### In-Lab Marks:

[1 mark] Demo the correct configuration of a single BGP router.

[1 mark] Demo full end-to-end from PC to PC connectivity with BGP routes.

The pre-lab is worth **33%**, the in-lab is worth **34%**, and the post-lab is worth **33%** of this lab, even though the number of points may differ between the three parts.

### Required Equipment:

- A laptop and/or a USB memory stick to save results for post-lab questions
- **Hard-cover lab notebook**, for reference during **SBA** at the end of the course.

### References and Resources:

- Notes on following pages
- Packet Tracer ver 8 (available from NetAcad)
- A short (2-3 min), simple explanation of BGP:  
<https://www.cloudflare.com/learning/security/glossary/what-is-bgp/>

## Comparison of OSPF and BGP

The purpose of this comparison is to start leveraging your knowledge of routing protocols (namely OSPF) in order to more easily understand another protocol i.e. Border Gateway Protocol (BGP).

Although there is *lots* of additional detail that you'll cover in later courses, the comparison below illustrates the differences for a few key points. It also illustrates that many common themes exist for most routing protocols. These themes appear in the first column of the table.

For this course, we'll stick with using BGP simply as a routing protocol; we'll leave its advanced features for later courses. The chart shows that the configuration starts off very simply: only **4 commands** for basic neighbouring, most of which should look somewhat familiar.

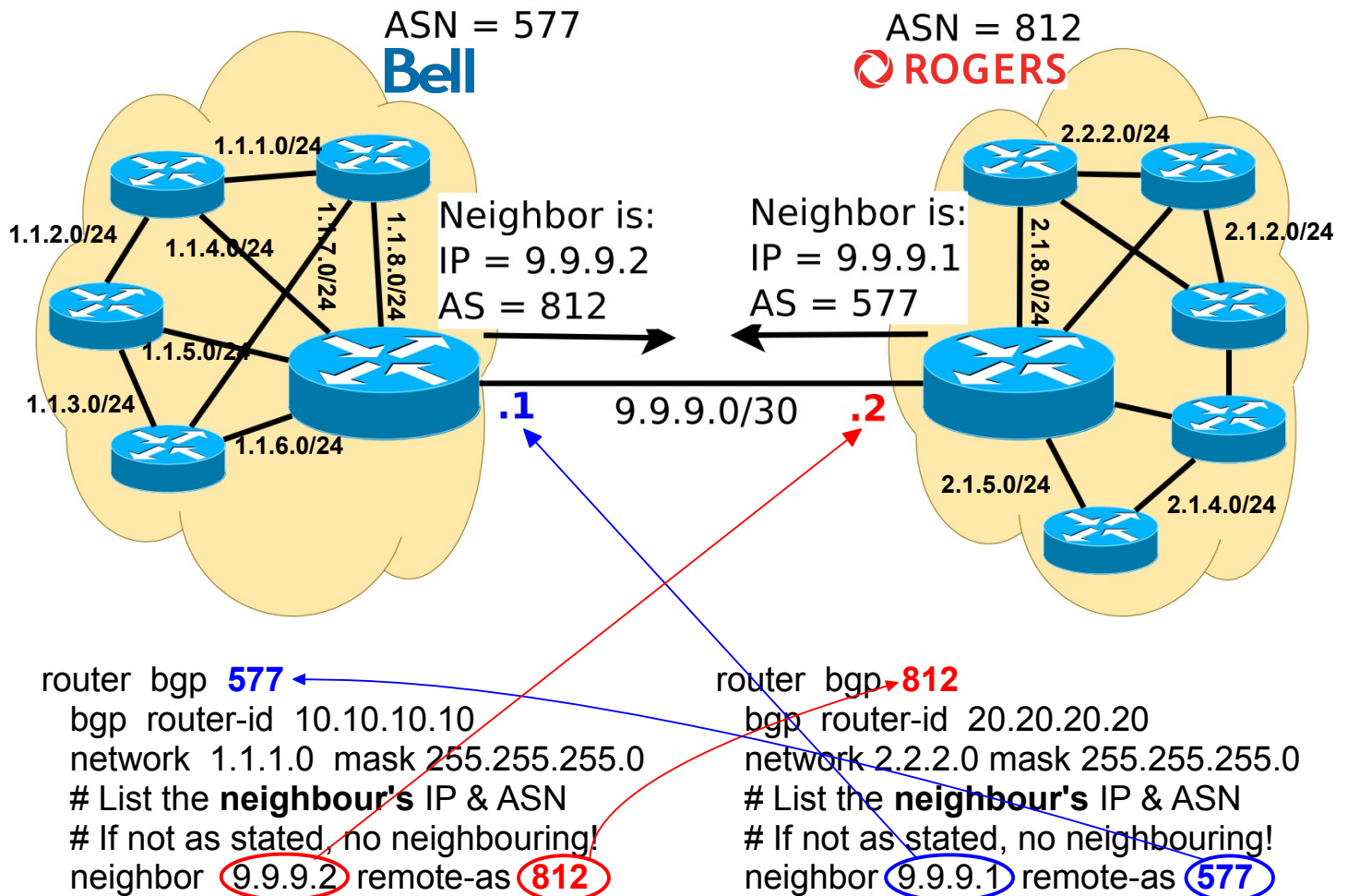
Item	OSPF	BGP	New BGP Config
Router grouping	by area	by ASN (Autonomous System Number)	—
Group numbering	<b>0</b> (backbone); 1 - 2 <sup>32</sup> : other areas	1 – 2 <sup>16</sup> (+ extended #'s) Private AS: 64512 – 65534	—
Config Context	router ospf {p-ID} - p-ID is purely local	router bgp {AS #} AS # matters between routers	router BGP {AS-#}
Router ID	Yes, must be unique	Yes, must be unique (32 bit)	bgp router-id x.x.x.x
<b>Neighbours</b>	<b>Always discovered, not configured</b>	<b>Always configured, never discovered</b>	<b>neighbor {ip} remote-as {ASN}</b>
Interface config	Yes with 2 purposes: 1. to reach neighbours 2. subnet to share	<b>No:</b> all config in <i>router bgp {X}</i>	—
Info shared	Link states (LSA 1-2) routes (LSA 3, 5)	<b>prefixes</b> (another name for subnets)	—
What routes are shared	- I/F's listed for OSPF <b>or</b> <i>network x.x.x.x</i>	- subnets (I/F's) identified by <i>network x.x.x.x</i>	<b>network x.x.x.x mask y.y.y.y</b> (y.y.y.y = <u>subnet mask!</u> )
Metric	Cost - simple sum of link costs	How many AS's the path goes through (no idea of cost!)	—
Other route adjustments	No. Only cost can be adjusted and used for route selection	YES! Many other parameters can be adjusted & used for route selection.	(... Covered in future courses such as NET3008 & NET3012 ...)
Timers	Yes: <b>Hello</b> & dead	Yes: <b>KeepAlive</b> , Hold time	timers bgp {KA} {Hold}
Transport	OSPFv2 = IPv4 <b>only</b> OSPFv3 = IPv6 <b>only</b>	IPv4 <b>or</b> IPv6 to neighbors (BGP is "bilingual")	—
Address Families	OSPFv2 = IPv4 <b>only</b> OSPFv3 = IPv4, IPv6	IPv4 IPv6 EVPN VPN-IPv4, VPN-IPv6 multicast-IPv4, multicast-IPv6 etc, etc	—

## Cisco IOS Configuration

Below is an example of eBGP peering: the two connected routers are in different Autonomous Systems. BGP peering is arranged by mutual cooperation – for a successful neighbouring:

- Bell must configure BGP on their router to reach out to Rogers
- Rogers must similarly configure their BGP to reach out to Bell

Since Bell has *no control* over the routers in Rogers' network, and vice versa, both providers must have an agreement or BGP will not succeed in neighbouring. (Not surprisingly, BGP neighbouring won't succeed either if there are any typos in the IP address or ASN!)



For the sake of clarity, the following simplifications have been made:

- the link address of 9.9.9.0/24 has been invented (although we could look up who owns it)
- only 1 (one) prefix is being shared by each provider (even though many subnets exist)
- there would be substantially more config for protection and security of each provider's network

The ASNs are real and correct:

- **Bell** Canada: <https://www.bigdatacloud.com/asn-lookup/AS577>
- **Rogers** Communications: <https://www.bigdatacloud.com/asn-lookup/AS812>

## Comparison of OSPF and BGP Commands

When looking at OSPF, we had 4 (four) 'show' commands that gave us pretty much all the info that we needed to confirm all the operational details.

What we'd like to start seeing is that there's plenty of common ground between most (all?) routing protocols in the 'show' commands we use to get the essential info. Consider the chart below, which shows comparable commands for OSPF and BGP and notice the common themes identified in the "Purpose" column.

One other point: you'll find that every network vendor (Cisco, Nokia, Juniper, MikroTik, etc) structures their commands so that they're similar across all routing protocols. For example:

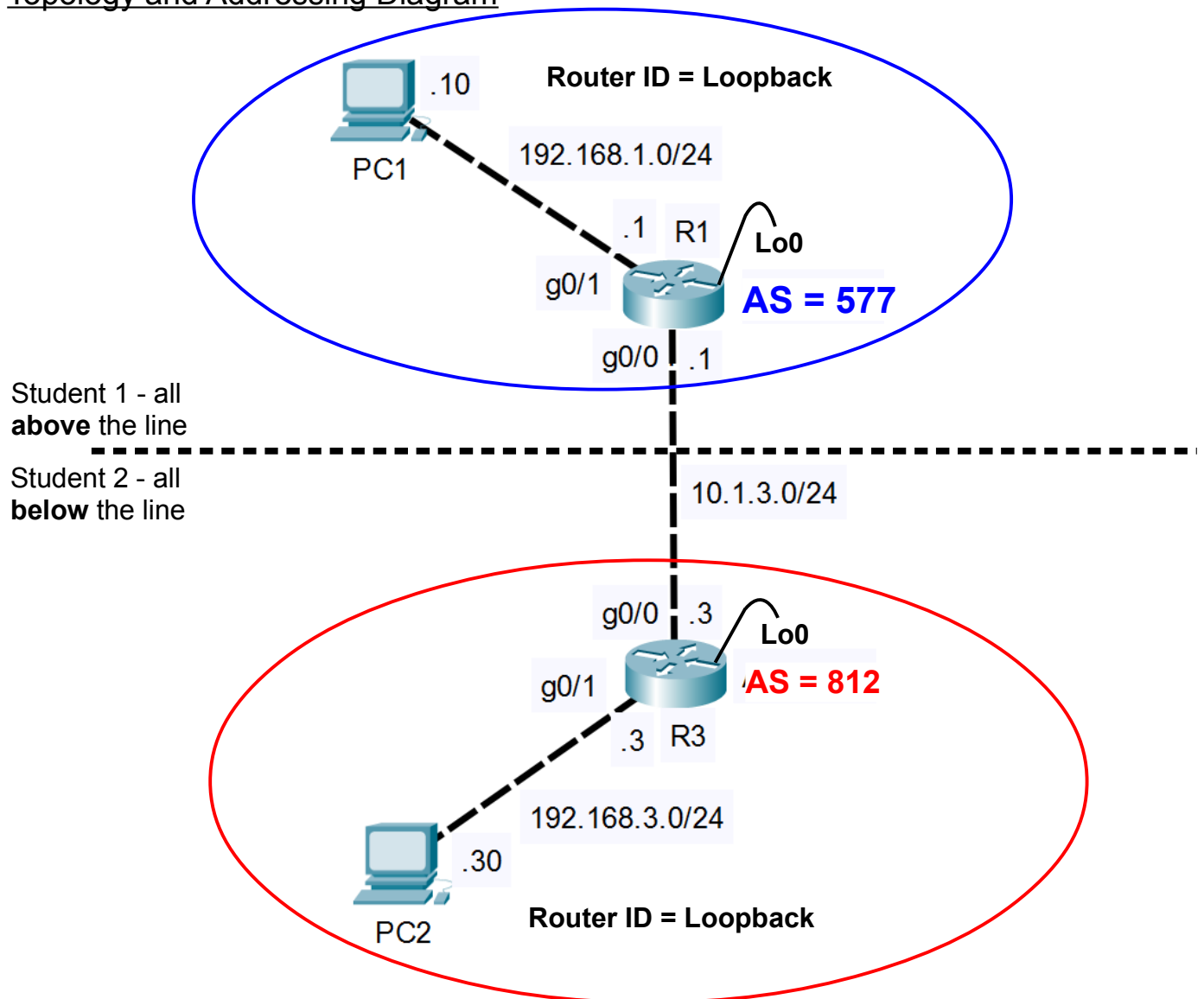
- Commands, whether for OSPF or BGP, mostly start the same way 'show ip [ospf | bgp] ...'
- Most commands can be converted from OSPF to BGP (or vice versa) by replacing "ospf" with "bgp"

This is hopefully encouraging: once you've learned one routing protocol, you've got a great head-start on learning any number of additional protocols!

<b>Purpose</b>	<b>OSPF (v2; use "ipv6" for v3)</b>	<b>BGP</b>
Show general status	show ip protocols	show ip protocols
Show list of known neighbours	show ip <b>ospf</b> neighbor	*show ip <b>bgp</b> neighbor
Show entries in Route Table	show ip route [ <b>ospf</b> ]	show ip route [ <b>bgp</b> ]
Show (potential) connections	show ip <b>ospf</b> interface brief	show ip <b>bgp</b> summary or show <b>bgp</b> summary
Fix mis-typed info (e.g. reset)	clear ip <b>ospf</b> process	clear ip <b>bgp</b> *
*Not so helpful (at this point); too much output to fully process	show ip <b>ospf</b> interface	show ip <b>bgp</b> neighbor

If you've carefully read and understood the above material, you should hopefully find this lab fairly straight-forward! :-)

## Topology and Addressing Diagram



Device	Interface	Address	Gateway
<b>R1</b> (student 1) LHS rack (odd # pods): any of R1 - R6	G0/0, G0/1	As per topology	(None used)
	<b>Lo0</b> – Loopback 0 and <b>router-ID</b>	<b>10.10.10.1/32</b>	
PC1 (student 1)	Realtek NIC	192.168.1.10/24	?
<b>R3</b> (student 2) RHS rack (even # pods): any of R1 - R6	G0/0, G0/1	As per topology	(None used)
	<b>Lo0</b> – Loopback 0 and <b>router-ID</b>	<b>10.10.10.3/32</b>	
PC2 (student 2)	Realtek NIC	192.168.3.30/24	?

## Task 1: Cable the Network

Cable the devices as shown in the diagram above, to create the required network topology.

## Task 2: Configure basic *settings for PCs*

Working in parallel with your partner, determine the correct gateway address for each PC. Configure the necessary settings: Realtek NIC address, mask, and gateway.

## Task 3: Configure your two Routers with the essentials

Still working in parallel with your partner, your two routers need to be configured with the four essential categories of basic config.

- As usual, starting with configs you saved from previous labs, edit them to use this lab's addressing scheme.
- Don't forget the "no ip domain-lookup" command to prevent long delays from a mis-typed command.
- Verify connectivity between your PC and its associated router. A successful ping from the PC to the router's loopback is rock-solid confirmation of full connectivity between PC + router.
- Verify connectivity between routers R1-R3.
- **Telnet or SSH** from your PC to its router, so that you can completely remove the console cable. You will need to be able to Telnet and SSH for the SBA. Remember to use '**terminal monitor**'.

## Task 4: Examine the Network Before BGP Neighbouring

Since this is a new protocol, we'll examine the network carefully for the status of BGP 'show ...' commands. This is really helpful for before-and-after comparisons.

- Step 1. Use the four 'show...!' commands listed in the chart on page 4. Capture, save, and note the output for any BGP details.
- Step 2. For troubleshooting purposes in the future, what helpful message is common to both 'show ip bgp neighbors' and 'show ip bgp summary'?
- Step 3. On **one router only** (doesn't matter which one), configure BGP:
  - enter the BGP router process context
  - assign a router ID
  - configure a BGP peer (the opposite router)
  - add a network statement to advertise a prefix for the loopback interface
- Step 4. Repeat step 1: run the 4 show commands on both routers, record & save the output, and compare to find the differences.

**Q.** Is there any BGP info now appearing on the router configured for BGP?

**Q.** Is there any BGP info now appearing on the router not-yet-configured for BGP?

**CHECK POINT:** Using 'show ...' output, prove that you have correctly configured the first (and only the first!) router for BGP.

## Task 5: Configure BGP Peering

Create a BGP peering by configuring BGP on the other router. Remember that you will not get successful BGP peering if there are any typos on either router. There's lots of numbers!

- Step 1. Similar to step 3 of the previous task, configure the other router for BGP. Both partners should be double-checking everything that's been typed!
- Step 2. If all the config is correct, you should see a Notification message indicating the peering succeeded. Note the format of the message:
  - the first character is '%'
  - the next field is the protocol identity
  - the next field is the severity (level 5 = simple Notification; not a problem)
  - the next field is the event or action
- Step 3. If you don't get successful peering, review your config. If you've put your name on the help-list and are waiting, you can try using debug commands:  
turn ON: `debug ip bgp out` or `debug ip bgp in` turn OFF: `undebug all`
- Step 4. Repeat the 'show ...' commands used previously. Save the results. What fields and items have changed since before the peering?
- Step 5. Confirm proper operation by pinging the neighbour router's loopback. If it doesn't succeed, check the details of your network statements.

Q. What is the full text of the notification message when successful BGP peering occurs?

Q. What changes do you see in the 'show ...' output once the BGP peering is successful?

## Task 6: Full End-to-End Connectivity

For end-to-end connectivity, both routers will need to advertise their PC subnets. You'll need to add network statements to make that happen.

- Step 1. Both partners should configure a network statement on their router so that the PC prefix (subnet) is advertised it's BGP neighbour.
- Step 2. Wait a few moments, and check the results of the 'show ...' commands. Do you see any changes, or is this the type of change which necessitates resetting the BGP connection? If necessary, and **only** if necessary, reset the connection.
- Step 3. Verify end-to-end connectivity between PCs.

Q. Does adding (or removing) a prefix (route) require resetting the routing protocol?

**CHECK POINT:** Show **both** the ping result (including TTL) and the routing table.

## Task 7: Backups and Clearing the equipment

You'll want/need a backup of the config for both routers for the post-lab. Make backups of your router, then exchange configs with your lab partner.

Be sure your equipment is cleared of all config before de-cabling and powering down.

`show startup-config` If the startup-exists, then: `erase startup-config`

Congratulations! You've learned the basics of a new routing protocol! :-)