Module 1 – Basic Topology, ISIS and iBGP

Objective: Create a basic physical lab interconnection with one ISIS Area and one BGP AS number. Ensure that all routers, interfaces, cables and connections are working properly.

Prerequisites: Knowledge of Cisco router CLI, previous hands on experience.

The following will be the common topology used for the first series of labs.

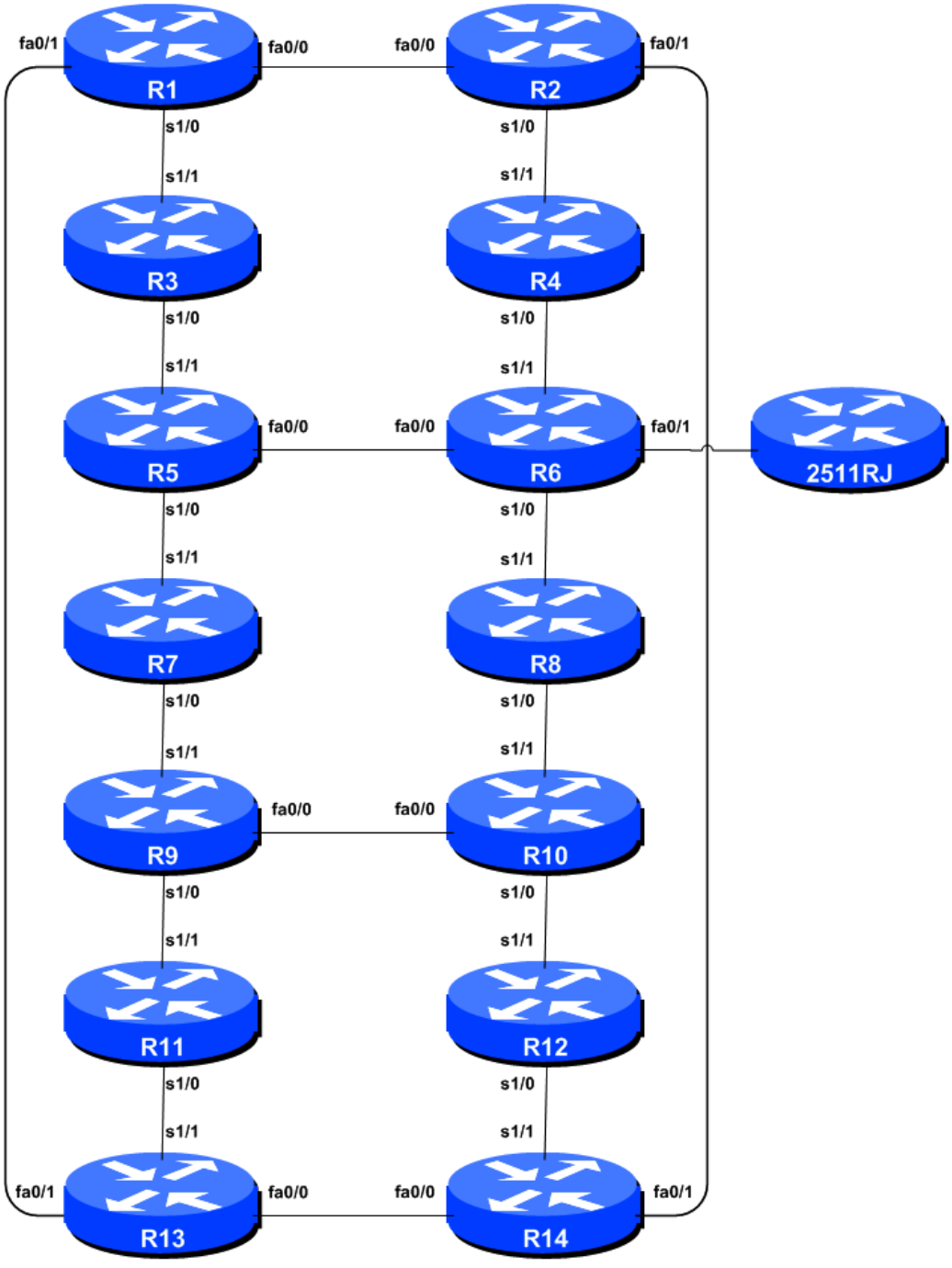

Figure 1 – ISP Lab Basic Configuration

Lab Notes

This workshop is intended to be run on a Dynamips server with the appropriate lab topologies set up. The routers in the Dynamips environment are using service provider IOS. The configurations and configuration principles discussed below will work on all Cisco IOS Release 12.4 onwards. Earlier Cisco IOS releases are not supported but will mostly work using the notes below; they will miss some of the features covered.

The purpose of this module is to construct the workshop lab and introduce everyone to the basic principles of constructing and configuring a network. An important point to remember, and one that will be emphasised time and again through out this workshop, is that there is a distinct sequence to building an operational network:

- After the **physical design** is established, the connections between the hardware should be built and verified.

- Next, the routers should have the **base configuration** installed, and basic but sufficient security should be set up.

- Next the **basic IP connectivity** be tested and proven. This means assigning IP addresses on all links which are to be used, and testing the links to the neighbouring devices.

- Only once one router can see its neighbour does it make sense to start configuring routing protocols. And **start with the IGP** (ISIS is chosen for this workshop). There is no purpose to building BGP while the chosen IGP (in this case ISIS) is not functioning properly. BGP relies on ISIS to find its neighbours and next hops, and an improperly or non-functioning ISIS will result in much time wasted attempting to debug routing problems.

- Once the IGP is functioning properly, the **BGP configuration** can be started, first internal BGP, then external BGP.

- **Remember to RTFM.** What is RTFM? It is critical that ISP Network Engineers fully utilise all information resources. The #1 source is the documentation. ***Read The F#$% Manual (RTFM)*** is the traditional phase used to inform engineers that the answer is in the documentation and go read it. We will use *RTFM* through out these exercises to highlight areas where the student should use the documentation for further deepening. There will be many new commands. *Please refer to the IOS Command Reference on Cisco Connection Online (CCO.cisco.com) for details on each of these commands.*

- Finally, **documentation**. Documentation is often overlooked or forgotten. It is an ongoing process in this workshop. If the instructor asks you to document something, either on the whiteboard in the class, or at the back of this booklet, it is in your best interests to do so. There can never be too much documentation, and documentation at the time of network design and construction can usually saves much frustration at a future date or event.

Lab Exercise

1. **Routers and the Workshops participants.** This workshop is laid out such that a group of two students will operate a single router. 14 routers generally imply at least 28 participants. For workshops with larger numbers of participants, groups of three should configure a single router. The Workshop Instructors will divide the routers amongst the workshop participants. In the following notes, a "router team" refers to the group assigned to one particular router.

2. **Router Hostname.** Each router will be named according to the table location, Router1, Router2, Router3, etc. Documentation and labs will also refer to *Router1* as R1. At the router prompt, first go into enable mode, then enter "config terminal", or simply "config" by itself:

```
Router> enable
Router# config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# hostname Router1
Router1(config)#
```

3. **Turn Off Domain Name Lookups.** Cisco routers will always try to look up the DNS for any name or address specified in the command line. You can see this when doing a *trace* on a router with no DNS server or a DNS server with no in-addr.arpa entries for the IP addresses. We will turn this lookup off for the labs for the time being to speed up traceroutes.

```
Router1 (config)# no ip domain-lookup
```

4. **Disable Command-line Name Resolution.** The router by default attempts to use the various transports it supports to resolve the commands entered into the command line during normal and configuration modes. If the commands entered are not part of Cisco IOS, the router will attempt to use its other supported transports to interpret the meaning of the name. For example, if the command entered is an IP address, the router will automatically try to connect to that remote destination. This feature is undesirable on an ISP router as it means that typographical errors can result in connections being attempted to remote systems, or time outs while the router tries to use the DNS to translate the name, and so on.

```
Router1 (config)# line con 0
Router1 (config-line)# transport preferred none
Router1 (config-line)# line vty 0 4
Router1 (config-line)# transport preferred none
```

5. **Disable Source Routing.** Unless you really believe there is a need for it, source routing should be disabled. This option, enabled by default, allows the router to process packets with source routing header options. This feature is a well-known security risk as it allows remote sites to send packets with different source address through the network (this was useful for troubleshooting networks from different locations on the Internet, but in recent years has been widely abused for miscreant activities on the Internet).

```
Router1 (config)# no ip source-route
```

6. **Usernames and Passwords.** All router usernames and passwords should be ***cisco***. Please do **not** change the username or password to anything else, or leave the password unconfigured (access to

vty ports is not possible if no password is set). It is essential for a smooth operating lab that all participants have access to all routers.

```
Router1 (config)# username cisco secret cisco
Router1 (config)# enable secret cisco
Router1 (config)# service password-encryption
```

The *service password-encryption* directive tells the router to encrypt all passwords stored in the router's configuration (apart from *enable secret* which is already encrypted).

Note A: while we use username *cisco* and many instances of a password of *cisco* in these workshops, under no circumstances must any service provider operator ever use easily guessable passwords as these on their live operational network[1].

Note B: for IOS releases prior to 12.3, the username/secret pair is not available, and operators will have to configure username/password instead. The latter format uses type 7 encryption, whereas the former is the more secure md5 based encryption.

7. **Enabling login access for other teams.** In order to let other teams telnet into your router in future modules of this workshop, you need to configure a password for all virtual terminal lines.

```
Router1 (config)# aaa new-model
Router1 (config)# aaa authentication login default local
Router1 (config)# aaa authentication enable default enable
```

This series of commands tells the router to look locally for standard user login (the username password pair set earlier), and to the locally configured enable secret for the enable login. By default, login will be enabled on all vtys for other teams to gain access.

8. **Configure system logging.** A vital part of any Internet operational system is to record logs. The router by default will display system logs on the router console. However, this is undesirable for Internet operational routers, as the console is a 9600 baud connection, and can place a high processor interrupt load at the time of busy traffic on the network. However, the router logs can also be recorded into a buffer on the router – this takes no interrupt load and it also enables to operator to check the history of what events happened on the router. In a future module, the lab will configuration the router to send the log messages to a SYSLOG server.

```
Router1 (config)# no logging console
Router1 (config)# logging buffer 8192 debug
```

which disables console logs and instead records all logs in a 8192byte buffer set aside on the router. To see the contents of this internal logging buffer at any time, the command "`sh log`" should be used at the command prompt.

9. **Save the Configuration.** With the basic configuration in place, save the configuration. To do this, exit from enable mode by typing "end" or "<ctrl> Z", and at the command prompt enter "write memory".

```
Router1(config)#^Z
```

[1] This sentence cannot be emphasized enough. It is quite common for attackers to gain access to networks simply because operators have used familiar or easily guessed passwords.

```
Router1# write memory
Building configuration...
[OK]
Router1#
```

It is highly recommended that the configuration is saved quite frequently to NVRAM, especially in the workshop environment where it is possible for power cables to become dislodged. If the configuration is not saved to NVRAM, any changes made to the running configuration will be lost after a power cycle.

Log off the router by typing exit, and then log back in again. Notice how the login sequence has changed, prompting for a "username" and "password" from the user. Note that at each checkpoint in the workshop, you should save the configuration to memory – remember that powering the router off will result in it reverting to the last saved configuration in NVRAM.

10. **IP Addresses.** This Module will introduce the basic concepts of putting together a sensible addressing plan for an ISP backbone. We are building one autonomous system out of the 14 routers we have in the lab. The RIRs are typically handing out IPv4 address space in /20 chunks (depends on which RIR region) – we assume for the purposes of this lab that our ISP has received a /20. Rather than using public address space, we are going to use a portion of 10/8 (RFC1918 or private address space) for this lab. In the real world Internet, we would use public address space for our network infrastructure.

    The typical way that ISPs split up their allocated address space is to carve it into three pieces. One piece is used for assignments to customers, the second piece is used for infrastructure point-to-point links, and the final piece is used for loopback interface addresses for all their backbone routers. The schematic in Figure 2 shows what is typically done.

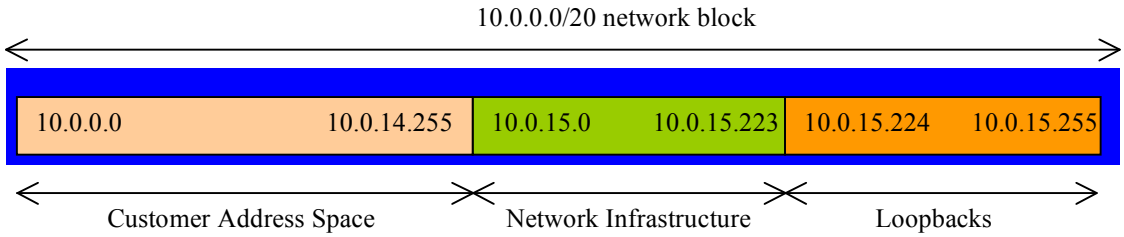

Figure 2 – Dividing allocated block of /20 into Customer, Infrastructure and Loopbacks

Study the address plan which was handed out as an addendum to this workshop module. Notice how the infrastructure addressing starts at 10.0.15.0 and carries on up to 10.0.15.70 – this leave us room to grow the network by more point-to-point links, up to 10.0.15.223 in fact. Notice how we have set a side just a single /27 for the router loopbacks – but we have only used the 14 addresses from 241 up to 254 for our network, leaving some spare for future growth (not that we have future growth planned for the workshop), an entirely realistic proposition for an ISP backbone. Indeed, ISPs tend to document their addressing plans in flat text files or in spreadsheets – Figure 3 below shows an extract from a typical example (using our addressing scheme here).

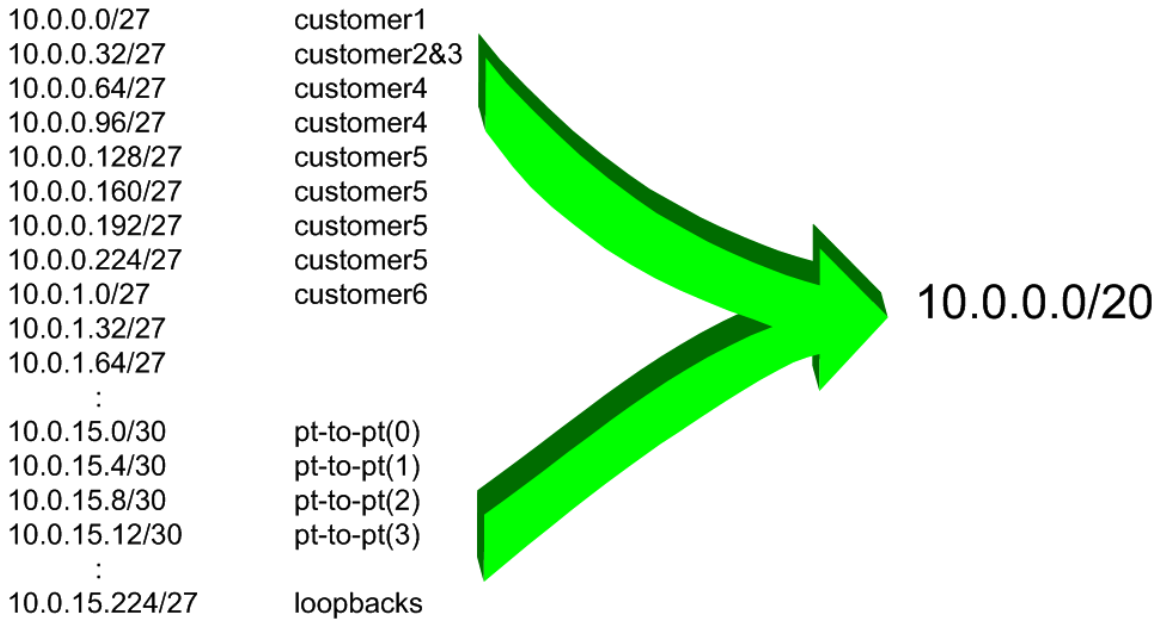

Figure 3 – Extract from an ISP addressing plan

11. **Back-to-Back Serial Connections.** Connect the serial connections as in Figure 1. The DCE side of a back-to-back serial connection is configured with the ***clock rate*** command that drives the serial circuit. (Older versions of IOS used the ***clockrate*** command, now hidden but still functional.) Physically check the cable to see which side is DCE and which is DTE. On some routers, the command ***show controller <interface>*** will show DCE/DTE status. For example, on a Cisco 3620 router, ***show controllers serial 0/0*** will produce a result that will display whether the cable connected to serial 0/0 is a DTE or DCE.

    Once the DTE and DCE cables have been determined and the ***clock rate*** command has been applied, configure the IP address (as per the addressing plan discussed earlier) and other recommended BCP commands that are recommended for each ISP's Interface:

```
Router2(config)# interface serial 0/0
Router2(config-if)# ip address 10.0.15.17 255.255.255.252
Router2(config-if)# description 2 Mbps Link to Router4 via DTE/DCE Serial
Router2(config-if)# clock rate 2000000
Router2(config-if)# no ip redirects
Router2(config-if)# no ip directed-broadcast
Router2(config-if)# no ip proxy-arp
Router2(config-if)# no shutdown
```

    NOTE: The lab instructors will have drawn a large network map on the white-board in the workshop lab. When the IP addresses are assigned, please annotate them and inform the instructor. All the point to point links **MUST** be annotated there so that other Router Teams can document and understand the links and routing in this and future modules.

    Q: What network mask should be used on point-to-point links?

    A: On serial interfaces, the network mask should be /30 (or 255.255.255.252 in dotted quad format). There is no point in using any other size of mask as there are only two hosts on such a link. A 255.255.255.252 address mask means 4 available host addresses, of which two are usable (the other two representing network and broadcast addresses).

12. **Ethernet Connections.** The Ethernet links between the routers will be made using *cross-over* RJ-45 cables – these will directly connect the Ethernet ports on the two routers without the requirement for an Ethernet switch. IP subnets will again be taken from the Addressing Plan. Don't make the mistake of assigning a /24 mask to the interface address – there are only two hosts on the Ethernet connecting the two routers, so a /30 mask should be entirely sufficient.

13. **Ping Test #1.** Ping all physically connected subnets of the neighbouring routers. If the physically connected subnets are unreachable, consult with your neighbouring teams as to what might be wrong. Don't ignore the problem – it may not go away. Use the following commands to troubleshoot the connection:

```
show arp                                  : Shows the Address resolution protocol
show interface <interface> <number>       : Interface status and configuration
show ip interface                         : Brief summary of IP interface status and configuration
```

14. **Create Loopback Interfaces.** Loopback interfaces will be used in this workshop for many things. These include generating routes (to be advertised) and configuring some BGP peerings. As discussed earlier in Step 10, we will use part of the allocated IP address block for loopback interfaces. Most ISPs tend to set aside a contiguous block of addresses for use by their router loopbacks. For example, if an ISP had 20 routers, they would need a /27 (or 32 host addresses) to provide a loopback address for each router. We have 14 routers in our lab – to be prudent and allow for growth, we will set aside a /27 (allows us 32 loopbacks) but only use 14 of them. The assigned loopback addresses are:

R1	**10.0.15.241/32**	**R8**	**10.0.15.248/32**
R2	**10.0.15.242/32**	**R9**	**10.0.15.249/32**
R3	**10.0.15.243/32**	**R10**	**10.0.15.250/32**
R4	**10.0.15.244/32**	**R11**	**10.0.15.251/32**
R5	**10.0.15.245/32**	**R12**	**10.0.15.252/32**
R6	**10.0.15.246/32**	**R13**	**10.0.15.253/32**
R7	**10.0.15.247/32**	**R14**	**10.0.15.254/32**

For example, Router Team 1 would assign the following address and mask to the loopback on Router 1:

```
Router1(config)#interface loopback 0
Router1(config-if)#ip address 10.0.15.241 255.255.255.255
```

Q: Why do we use /32 masks for the loopback interface address?

A: There is no physical network attached to the loopback so there can only be one device there. So we only need to assign a /32 mask – it is a waste of address space to use anything else.

15. **ISIS with one area and one level (level-2) within the same AS.** Each router team should enable ISIS on their router, and use *workshop* as the ISIS ID in the configuration. In this module, we use level-2 in one area (*49.0001*) and use wide metrics (IOS default is the historical narrow metric and is not considered good practice). The NET should be *49.0001.x.x.x.x.00*, where *x.x.x.x* represents the router loopback IP address. For example, the loopback for Router1 is 10.0.15.241 which will make the NSAP address *49.0001.0100.0001.5241.00*.

```
Router1(config)# router isis workshop
Router1(config-router)#net 49.0001.0100.0001.5241.00
Router1(config-router)#is-type level-2-only
```

Q: Why do you have *`is-type level-2-only`* configured? Write your answer here:

Hint: A nice trick for converting the loopback interface address into the NSAP address is to take the loopback address and put the missing leading zeroes in. For example, Router 5 loopback address is 10.0.15.245; this is rewritten to 010.000.015.245 putting in the missing zeroes. Then rather than having the dot after every third character, move it to be after every fourth character. So 010.000.015.245 becomes 0100.0001.5245.

16. **Setting Wide Metrics.** We also set the metric-style to wide. ISIS supports two types of metric, narrow (historic now and not suitable for modern networks) and wide. IOS still defaults to narrow metrics, so we need to enter explicit configuration to change this to wide.

```
Router1(config)# router isis workshop
Router1(config-router)#metric-style wide
```

17. **Activating ISIS on each interface.** Now that the ISIS process is configured, all connected point to point and shared ethernet interfaces need to be configured with ISIS. Else, you will not be able to see network advertisements via ISIS from routers two or more hops away. Here is an example configuration as would be used on Router1:

```
Router1(config)# interface fastethernet 0/0
Router1(config-if)# ip router isis workshop
!
Router1(config)# interface fastethernet 0/1
Router1(config-if)# ip router isis workshop
!
Router1(config)# interface serial 0/1
Router1(config-if)# ip router isis workshop
```

Note: the ISIS ID on the interfaces must be matched with the router's ISIS ID.

18. **ISIS Metrics.** Now each team needs to set the ISIS metric on each physical interface. The default ISIS metric for all interface types is 10. Unlike OSPF in IOS, ISIS has no automatic scheme to convert the interface bandwidth into a metric value. ISPs deploying ISIS have to come up with their own scheme (as in fact many ISPs using OSPF now also do).

    In the lab we will use metric 2 for the Ethernet interfaces and metric 20 for the Serial interfaces. For example:

```
Router1(config)# interface fastethernet 0/0
Router1(config-if)# isis metric 2 level-2
!
Router1(config)# interface fastethernet 0/1
Router1(config-if)# isis metric 2 level-2
!
Router1(config)# interface serial 0/0
Router1(config-if)# isis metric 20 level-2
```

19. **Announcing the Loopback /32.** We do not need to set up ISIS adjacencies on the loopback interface as there are no neighbours there, so we mark it as passive:

```
Router1(config)# router isis workshop
Router1(config-router)# passive-interface Loopback0
```

Note that this will tell ISIS to install the loopback interface address in the ISIS RIB. We do NOT need to add an `ip router isis` statement onto the loopback interface itself. This is different from the required OSPF configuration, and often catches many engineers out, especially those who are learning ISIS after gaining experience with OSPF.

20. **ISIS Adjacencies.** Enable logging of ISIS adjacency changes. This is so that a notification is generated every time the state of a CLNS neighbor changes, and is useful for debugging purposes.

(**Note:** From IOS 12.4 onwards, *log-adjacency-changes* is activated by default when ISIS is first configured.)

```
Router1(config)#router isis workshop
Router1(config-router)#log-adjacency-changes
```

21. **Ping Test #2.** Ping all loopback interfaces. This will ensure the ISIS IGP is connected End-to-End. If there are problems, use the following commands to help determine the problem:

`show ip route`	: see if there is a route for the intended destination
`show clns neighbor`	: see a list of CLNS-IS neighbors that the router sees
`show clns interface`	: see if ISIS is configured and see the IS type
`show isis database`	: see ISIS link state database that the router has learned

Checkpoint #1: *call lab assistant to verify the connectivity. Save the configuration as it is on the router – use a separate worksheet, or the workspace at the end of this Module. You will require this configuration several times throughout the workshop.*

22. **Configuring iBGP.** Before we set up iBGP with our neighbours in our AS, we need to do some basic preparation on the router. The IOS defaults are not optimised for Service Provider networks, so before we bring up BGP sessions, we should set the defaults that we require.

The default distance for eBGP is 20, the default distance for iBGP is 200, and the default distance for ISIS is 115. This means that there is a potential for a prefix learned by eBGP to override the identical prefix carried by ISIS. Recall from the Routing presentation that there is a distinct separation between BGP and ISIS processes – prefixes present in ISIS will never be found in BGP, and vice-versa. To protect against accidents[2], the eBGP distance is set to 200 also. The command to do this is the bgp distance subcommand, syntax is:

```
distance bgp <external-routes> <internal-routes> <local-routes>
```

[2] There have been several incidents in the past where denial of service attacks on ISP networks have been successful because ISPs have omitted basic routing protocol security. Setting the BGP distances to be greater than any IGP is one of the mitigation methods available.

Note: This should be included in all future BGP configurations in this workshop. For example, for Router2, the configuration might be:

```
Router2(config)#router bgp 10
Router2(config-router)#distance bgp 200 200 200
```

23. **Logging of BGP Adjacency State.** Enable logging of BGP neighbour changes. This is so that a notification is generated every time the state of a BGP neighbour changes state, and is useful for debugging purposes:

```
Router2(config)#router bgp 10
Router2(config-router)# bgp log-neighbor-changes
```

Note: From IOS 12.3 onwards, *bgp log-neighbor-changes* is activated by default when BGP is first configured.

24. **Configuring iBGP Neighbours.** All Routers will be in Autonomous System (AS) 10 for this first lab. Use the *show ip bgp summary* to check the peering. The BGP peering will be established using the loopback interfaces' IP address.

```
Router2(config)#router bgp 10
Router2 (config-router)#neighbor 10.0.15.241 remote-as 10
Router2 (config-router)#neighbor 10.0.15.241 update-source loopback 0
Router2 (config-router)#neighbor 10.0.15.241 description iBGP with Router1
Router2 (config-router)#
Router2 (config-router)#neighbor 10.0.15.243 remote-as 10
Router2 (config-router)#neighbor 10.0.15.243 update-source loopback 0
Router2 (config-router)#neighbor 10.0.15.243 description iBGP with Router3
Router2 (config-router)#
Router2 (config-router)#neighbor 10.0.15.244 remote-as 10
Router2 (config-router)#neighbor 10.0.15.244 update-source loopback 0
Router2 (config-router)#neighbor 10.0.15.244 description iBGP with Router4
Router2 (config-router)#
Router2 (config-router)#neighbor 10.0.15.245 remote-as 10
Router2 (config-router)#neighbor 10.0.15.245 update-source loopback 0
Router2 (config-router)#neighbor 10.0.15.245 description iBGP with Router5
Router2 (config-router)#
Router2 (config-router)#neighbor 10.0.15.246 remote-as 10
Router2 (config-router)#neighbor 10.0.15.246 update-source loopback 0
Router2 (config-router)#neighbor 10.0.15.246 description iBGP with Router6
Router2 (config-router)#
Router2 (config-router)#neighbor 10.0.15.247 remote-as 10
Router2 (config-router)#neighbor 10.0.15.247 update-source loopback 0
Router2 (config-router)#neighbor 10.0.15.247 description iBGP with Router7
Router2 (config-router)#
Router2 (config-router)#neighbor 10.0.15.248 remote-as 10
Router2 (config-router)#neighbor 10.0.15.248 update-source loopback 0
Router2 (config-router)#neighbor 10.0.15.248 description iBGP with Router8
Router2 (config-router)#
Router2 (config-router)#neighbor 10.0.15.249 remote-as 10
Router2 (config-router)#neighbor 10.0.15.249 update-source loopback 0
Router2 (config-router)#neighbor 10.0.15.249 description iBGP with Router9
Router2 (config-router)#
Router2 (config-router)#neighbor 10.0.15.250 remote-as 10
Router2 (config-router)#neighbor 10.0.15.250 update-source loopback 0
Router2 (config-router)#neighbor 10.0.15.250 description iBGP with Router10
Router2 (config-router)#
Router2 (config-router)#neighbor 10.0.15.251 remote-as 10
```

```
Router2 (config-router)#neighbor 10.0.15.251 update-source loopback 0
Router2 (config-router)#neighbor 10.0.15.251 description iBGP with Router11
Router2 (config-router)#
Router2 (config-router)#neighbor 10.0.15.252 remote-as 10
Router2 (config-router)#neighbor 10.0.15.252 update-source loopback 0
Router2 (config-router)#neighbor 10.0.15.252 description iBGP with Router12
Router2 (config-router)#
Router2 (config-router)#neighbor 10.0.15.253 remote-as 10
Router2 (config-router)#neighbor 10.0.15.253 update-source loopback 0
Router2 (config-router)#neighbor 10.0.15.253 description iBGP with Router13
Router2 (config-router)#
Router2 (config-router)#neighbor 10.0.15.254 remote-as 10
Router2 (config-router)#neighbor 10.0.15.254 update-source loopback 0
Router2 (config-router)#neighbor 10.0.15.254 description iBGP with Router14
```

Q. Why is *update-source loopback 0* necessary on iBGP?

Use *show ip bgp summary* to check the status of the iBGP neighbour connections. If the iBGP session is not up and/or no updates are being sent, work with the Router Team for that neighbour connection to troubleshoot the problem.

25. **Sanity Check.** Remember to use the following commands to ensure you are getting the information you are suppose to be getting:

`show clns neighbor`	: see a list of CLNS-IS neighbours that the router sees
`show isis database`	: see ISIS link state database that the router has learned
`show ip bgp summary`	: see a list of BGP **peers** that the router sees
`show ip bgp`	: see a list of BGP **paths** that the router sees
`show ip route`	: see all the routes that the router has installed

Q. Are there routes seen via ***show ip bgp***? If not, why not? Are there any routes tagged "B" when you do a ***show ip route***?

26. **Add Networks via BGP.** Each Router Team will use BGP to advertise the address block used for the Module. For example, Router Team 1 would add:

```
Router1 (config)#router bgp 10
Router1 (config-router)#network 10.0.0.0 mask 255.255.240.0
```

Use *show ip bgp* on neighbour's router to see if you are advertising your network via BGP.

Q. Does the network show up via BGP? If not, why?

Enter a static route for the CIDR block. For example, Router 1 would use:

```
Router1 (config)#ip route 10.0.0.0 255.255.240.0 Null0
```

Q. Does the network show up via a neighbour's BGP? Use the command ***show ip bgp neighbor <neighbour's IP address> advertised-routes*** to see what you are exporting to the other router. Physically go to one of your neighbour's routers and check their BGP Table. Explain what you see.

Q. Does the network appear in the router's forwarding table? Use the command ***show ip route*** to check the local forwarding table. If not, why not?

27. For Routers with IOS prior to 12.3 add the following commands to BGP:

```
Router1 (config)#router bgp 10
Router1 (config-router)# no synchronization
Router1 (config-router)# no auto-summary
```

Q. Does the network appear in the router's forwarding table? Use the command ***show ip route*** to check the local forwarding table. What does the *no synchronisation* command do in BGP? How does it effect the router's forwarding table?

Note: As from IOS 12.3, synchronization and auto-summarisation are disabled by default and do not appear in the default BGP configuration. These two features have not been required in Service Provider networks since the classless routing system was introduced to the Internet in 1994.

Checkpoint #2 : *call the lab assistant to verify the connectivity.*

28. Adding a "customer" route into BGP (Background & Example). We are now going to add a "customer" route into BGP on each router. Now in the lab we don't have any "customers" as such connected to our routers, so we are going to simulate the connectivity by simply using a Null0 interface. **As an example**, in real life, the configuration to connect a customer would look something like this.

```
ip route 172.16.4.0 255.255.255.128 Serial 0/5/2 permanent
!
router bgp 64509
 network 172.16.4.0 mask 255.255.255.128
!
```

This would add a static route pointing 172.16.4.0/25 towards Serial 0/5/2 – the latter interface would be a fixed link connecting to the customer site. 172.16.4.0/25 would be the address space that the ISP had assigned to the customer. The BGP network statement would then add the customer address block into the ISP's iBGP.

Note: the **`permanent`** keyword ensures that the static route is always in the routing table, even if the interface physically goes down. Many ISPs use this to ensure they don't have iBGP churn when their customer links go down.

29. Adding a "customer" route into BGP. The "customer" address space that each router team will introduce into the iBGP is listed below – we will each use a /26, for simplicity's sake.

R1	**10.0.0.0/26**	**R8**	**10.0.1.192/26**
R2	**10.0.0.64/26**	**R9**	**10.0.2.0/26**
R3	**10.0.0.128/26**	**R10**	**10.0.2.64/26**
R4	**10.0.0.192/26**	**R11**	**10.0.2.128/26**
R5	**10.0.1.0/26**	**R12**	**10.0.2.192/26**
R6	**10.0.1.64/26**	**R13**	**10.0.3.0/26**
R7	**10.0.1.128/26**	**R14**	**10.0.3.64/26**

Each team should now set up a static route pointing to the **NULL0** interface for the /26 that they are to originate. Once the static is set up, the team should then add an entry into the BGP table. Here is an example for Router11:

```
Router11 (config)# ip route 10.0.2.128 255.255.255.192 Null0
Router11 (config)# router bgp 10
Router11 (config-router)# network 10.0.2.128 mask 255.255.255.192
```

30. **Check the BGP table.** Are there routes seen via ***show ip bgp***? If not, why not? Once every team in the class has done their configuration, each team should see the aggregate as well as the fourteen /26s introduced in the previous step. If this is not happening, work with your neighbours to fix the problem.

Checkpoint #3 : *call the lab assistant to demonstrate the current BGP table.*

31. **Traceroute to all routers.** Once you can ping all the routers, try tracing routes to all the routers using *trace x.x.x.x* command. For example, Router Team 1 would type:

```
Router1# trace 10.0.15.252
```

to trace a route to Router R12. If the trace times out each hop due to unreachable destinations, it is possible to interrupt the *traceroute* using the Cisco break sequence CTRL-^.

Q. Why do some trace paths show multiple IP addresses per hop?

A. If there are more than one equal cost paths, ISIS will "load share" traffic between those paths.

```
Router1>trace router12

Type escape sequence to abort.
Tracing the route to router12.workshop.net (10.0.15.224)

  1 fe0-0.router2.workshop.net (10.0.15.2) 4 msec
    fe0-1.router13.workshop.net (10.0.15.6) 0 msec
    fe0-0.router2.workshop.net (10.0.15.2) 0 msec
  2 fe0-0.router14.workshop.net (10.0.15.54) 4 msec
    fe0-1.router14.workshop.net (10.0.15.26) 4 msec
    fe0-0.router14.workshop.net (10.0.15.54) 0 msec
  3 ser0-0.router12.workshop.net (10.0.15.69) 4 msec *  4 msec
Router1>
```

32. **Other Features in ISIS and BGP.** Review the documentation or use command line help by typing *?* to see other *show* commands and other ISIS and BGP configuration features.

33. **Advanced Configuration.** Those router teams who have completed this module should refer to Module 11 of the Advanced BGP Workshop. The set-up steps have been extended to include all the basic requirements of a router being used in an ISP backbone. While waiting for the module to complete, now would be a good time to review the advanced Module and incorporate the additions to the configuration used here.

Review Questions

1. What IP Protocol does Ping and Traceroute use?

2. Ping the IP address of your neighbour's router (for example 10.0.15.2). Look at the time it took for the ping to complete. Now Ping the IP address of your router on the same segment (for example 10.0.15.1). Look at the time it took to complete a ping. What are the results? Why is there a difference?

3. What IOS show command(s) will display the router's forwarding table?

4. What IOS show command(s) will display the router's ISIS database?

5. What IOS show command(s) will display the router's BGP route table?

6. Why change the eBGP distance from the default 20 to 200?

7. Why is the static pull up route necessary in relation to inserting a prefix into iBGP?