

Module 1 – Basic Topology, OSPF and iBGP

Objective: Create a basic physical lab interconnection with one OSPF 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.

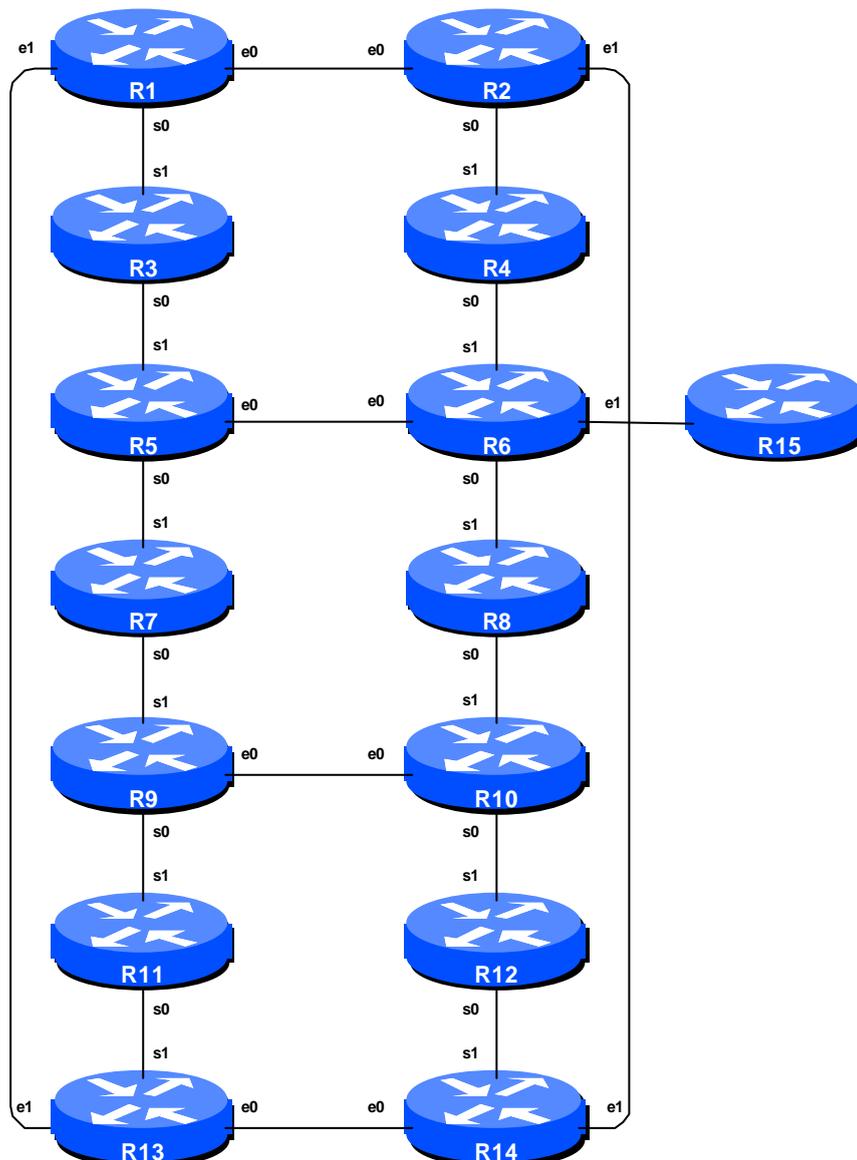


Figure 1 – ISP Lab Basic Configuration

Sunday, January 12, 2003

Lab Notes

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** (OSPF is chosen for this workshop). There is no purpose to building BGP while the chosen IGP (in this case OSPF) is not functioning properly. BGP relies on OSPF to find its neighbours and next hops, and an improperly or non-functioning OSPF 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 phrase used to inform engineers that the answer is 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 Cisco Documentation CD 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.

To explain Figure 1 in more detail:

- e0 and e1 refer to the 1st and 2nd ethernet interfaces on the router. Depending on the model, these may actually appear on the router as *ethernet 0* or as *ethernet 0/1*.
- s0 and s1 refer to the 1st and 2nd synchronous serial interfaces on the router. Again it depends on which model as to the actual interface naming scheme per the configuration.
- Each router has an RJ-45 style console port, as well as an auxiliary port – the console cable to the PC needs to be connected to the console port of the router.
- Router15 is the router used by the lab instructors during various exercises throughout the workshop. The instructors will tell you if any configuration is required.

Lab Exercise

1. **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)#
```

2. **Turn Off Domain Name Lookups.** Cisco routers will always try to look up the DNS for any name typed on 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
```

3. **Username 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 password cisco
Router1 (config)# enable secret cisco
Router1 (config)# service password-encryption
```

Sunday, January 12, 2003

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).

- 4. Enabling login access for other teams.** In order to let other teams telnet into your router, 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.

- 5. (For IOS < 12.0). CIDRise the router.** Make sure the router is configured for CIDR. Check the current configuration to ensure that the following commands appear. If they do not (IOS prior to 12.0), you will need to enter the following:

```
Router1 (config)# ip subnet-zero
Router1 (config)# ip classless
Router1 (config)#
```

Q: Why is *ip subnet-zero* necessary?

A: RFC791, historically, prohibited the use of the all-zero and all-one subnet number as part of the definition of class A, B and C address space. RFC1812, the router requirement RFC, lifts these restrictions. *ip subnet-zero* permits the use of the zero subnet in IOS (all-one subnet has always been usable on Cisco routers).

Note: *ip classless* is the default setting as from IOS 11.3. It allows the forwarding of packet destined to unknown subnets (of a known majornet) to follow a less specific prefix such as a default route.

Note: *ip subnet-zero* is the default setting as from IOS 12.0.

- 6. 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
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.

7. IP Addresses.

Each router is assigned a block of IP addresses.

R1	200.200.4.0/22	R8	220.220.4.0/22
R2	200.200.8.0/22	R9	220.220.8.0/22
R3	200.200.16.0/22	R10	220.220.16.0/22
R4	210.210.4.0/22	R11	222.222.4.0/22
R5	210.210.8.0/22	R12	222.222.8.0/22
R6	210.210.16.0/22	R13	222.222.16.0/22
R7	210.210.32.0/22	R14	222.222.32.0/22

Make a separate note of the address block assigned to your router – reference will be made to it several times throughout the rest of this workshop.

8. Back to Back Serial Connections.

Connect the serial connections as in Figure 1. IP addresses for the point to point links will be worked out between each *Router Team*¹. For instance, Router Team 1 and Router Team 3 will need to discuss with each other which part of which team’s address block will be used for the serial point to point link between them. (See the addressing plan in the Appendices for some suggestions.)

The DCE side of a back to back serial connection is configured with the **clock rate** command that drives the serial circuit. 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, **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 and other recommended BCP commands that are recommended for each ISP’s Interface:

¹ The Router Team is the group of students working on a router.

Sunday, January 12, 2003

```
Router2(config)# interface serial 0
Router2(config-if)# ip address 200.200.9.1 255.255.255.252
Router2(config-if)# description 2 Mbps Link to Router4 via DTE/DCE Serial
Router2(config-if)# bandwidth 2000
Router2(config-if)# clockrate 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
```

When the IP addresses are assigned, please annotate them and inform the instructor. A large network map will have been drawn on the white-board – 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).

9. Ethernet Connections. Use the Catalyst 2924XLs to interconnect the routers using ethernet. *Straight* RJ-45 cables will be used to connect the routers to the switches. If more than one collision domain is needed on the switch, configure the switch for multiple vLANs. IP subnets will be worked out between the Router Teams (again see the Appendices for hints). 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 more than sufficient.

Note: it is quite likely that the switched will have been configured as 3 vLANs already. The lab instructors will tell you. Each vLAN occupies one group of 8 ports on the switch.

10. 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
```

11. 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. Each Router Team will break apart their IP address block and assign part of it for use for loopback interfaces. Here

are suggestions for the network blocks to assign to each router (these lab notes assume these assignments in all examples):

R1	200.200.7.224/27	R8	220.220.7.224/27
R2	200.200.11.224/27	R9	220.220.11.224/27
R3	200.200.19.224/27	R10	220.220.19.224/27
R4	210.210.7.224/27	R11	222.222.7.224/27
R5	210.210.11.224/27	R12	222.222.11.224/27
R6	210.210.19.224/27	R13	222.222.19.224/27
R7	210.210.35.224/27	R14	222.222.35.224/27

For example, Router Team 1 would take the last /27 block (200.200.7.224/27) from their allocated /22 address block and select the first IP address in that /27 block for the loopback0 interface:

```
Router1(config)#interface loopback 0
Router1(config-if)#ip address 200.200.7.224 255.255.255.255
```

Router 1 200.200.4.0/22 network block

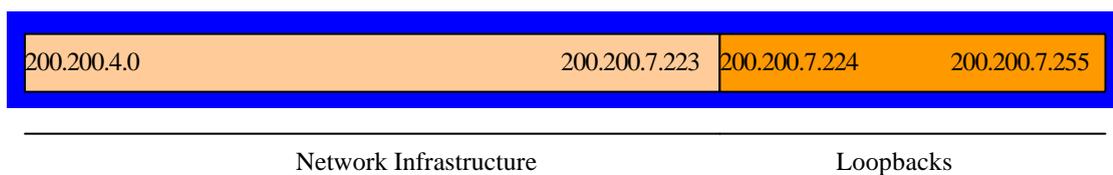


Figure 2 – Loopback Addressing Scheme: Picking the last /27 in the allocated block of /22

A /27 has been chosen by way of example only. Obviously the size of the address block should be assigned according to the extent of the infrastructure being configured. For example, if there are only 12 routers or other devices requiring a loopback interface in the network, only a /28 should be reserved for the loopback space (a /28 allows a maximum of 16 loopback interfaces to be configured). The example in Figure 3 might show how the loopback block fits into an ISP network-addressing scheme.

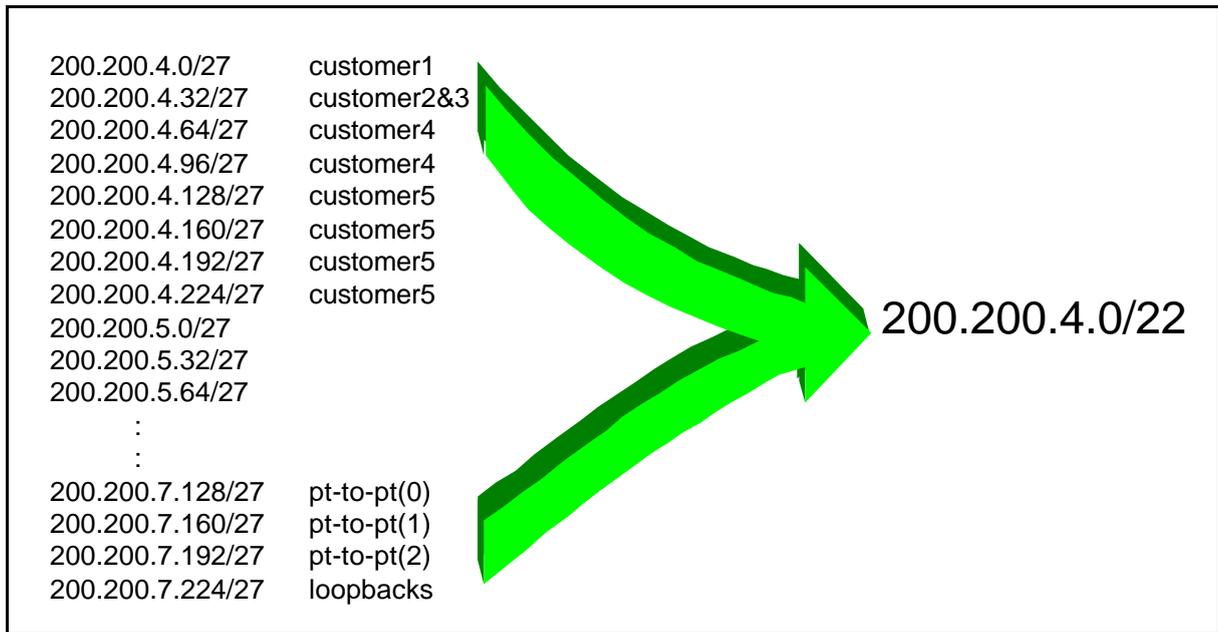


Figure 3 – Extract from ISP addressing plan

12. OSPF within the same AS. Each router Team should enable OSPF on their router. The OSPF process identifier should be *100* (see example). There needs to be a network statement for each interface over which OSPF adjacencies (exchanging OSPF routing information with neighbouring OSPF speakers) will be established on your router.

```
Router1(config)#router ospf 100
Router1(config-router)#network 200.200.4.0 0.0.0.3 area 0
```

All subnets for connected point to point and shared ethernet segments need to be configured into OSPF. Otherwise you will not be able to set up OSPF adjacencies.

Q: Why must you have each subnet from connected connection entered into OSPF?

Also, you want the subnets of connected interfaces which are not being used for OSPF adjacencies to appear in OSPF. This is so that the other routers can see the loopback interfaces and any other configured interfaces on your router which are not required to run OSPF:

```
Router2(config)#router ospf 100
Router2(config-router)#network 200.200.11.224 0.0.0.0 area 0
Router2(config-router)#passive-interface Loopback 0
```

Notice the “passive-interface” statement. This suppresses routing updates on the specified interface and is required for every interface which is not used to form OSPF adjacencies. For the loopback, this saves

router CPU, but it is more important that the “passive-interface” directive is applied to externally facing interfaces, for example customers, external networks, etc. If not, then it may be possible for external networks to form OSPF adjacencies with the internal network, and the potential problems this may bring.

- 13. OSPF Adjacencies.** Enable logging of OSPF adjacency changes. This is so that a notification is generated every time the state of an OSPF neighbour changes, and is useful for debugging purposes:

```
Router2(config)#router ospf 100
Router2(config-router)#ospf log-adjacency-changes
```

Note: from IOS 12.0 onwards, the command has been changed to drop the “ospf” directive. And from release 12.1 onwards, the “ospf” directive is not supported when enabling “log-adjacency-changes”.

- 14. (Optional). Enable DNS name and address resolution on the routers.** If the workshop instructors have set up the nameserver in the workshop at this stage, all router teams should now enable DNS lookups on their routers. OSPF is carrying all the prefixes, including the network connecting to Router15, around the classroom, so all routers should be able to see Router15.

```
Router2(config)#ip domain-lookup
Router2(config)#ip name-server 192.168.1.4
Router2(config)#ip domain-name workshop.net
```

These commands undo what was configured in step 2 at the beginning of the module. Make sure that you can ping the nameserver before you do this. If you can’t ping the nameserver, investigate why.

- 15. (Optional). Enable OSPF name lookups on the routers.** Following from the previous step, now enable OSPF name lookups on the router.

```
Router2(config)#ip ospf name-lookup
```

This command enables the display of the OSPF router-ids as domain names. So, rather than displaying the following output with name lookups disabled:

```
router2>sh ip ospf neigh
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
200.200.7.224	1	FULL/BDR	00:00:36	200.200.4.1	Ethernet0/0
210.210.7.224	1	FULL/ -	00:00:32	200.200.9.2	Serial0/0
222.222.35.224	1	FULL/DR	00:00:38	200.200.10.2	Ethernet0/1

the router will display the following:

```
router2#sh ip ospf neigh
```

Sunday, January 12, 2003

Neighbor ID	Pri	State	Dead Time	Address	Interface
router1.worksho	1	FULL/BDR	00:00:33	200.200.4.1	Ethernet0/0
router4.worksho	1	FULL/ -	00:00:39	200.200.9.2	Serial0/0
router14.worksh	1	FULL/DR	00:00:35	200.200.10.2	Ethernet0/1

which is much more informative.

16. Ping Test #2. Ping all loopback interfaces in the classroom. This will ensure the OSPF 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 ip ospf	: see general OSPF information
show ip ospf interface	: Check if OSPF is enabled on all intended interface
show ip ospf neighbor	: see a list of OSPF neighbours that the router sees

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.

17. Configuring iBGP Neighbours. All Routers will be in Autonomous System (AS) 100 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 100
Router2 (config-router)#neighbor 200.200.7.224 remote-as 100
Router2 (config-router)#neighbor 200.200.7.224 update-source loopback 0
Router2 (config-router)#neighbor 200.200.7.224 description iBGP with Router1
Router2 (config-router)#
Router2 (config-router)#neighbor 200.200.19.224 remote-as 100
Router2 (config-router)#neighbor 200.200.19.224 update-source loopback 0
Router2 (config-router)#neighbor 200.200.19.224 description iBGP with Router3
Router2 (config-router)#
Router2 (config-router)#neighbor 210.210.7.224 remote-as 100
Router2 (config-router)#neighbor 210.210.7.224 update-source loopback 0
Router2 (config-router)#neighbor 210.210.7.224 description iBGP with Router4
Router2 (config-router)#
Router2 (config-router)#neighbor 210.210.11.224 remote-as 100
Router2 (config-router)#neighbor 210.210.11.224 update-source loopback 0
Router2 (config-router)#neighbor 210.210.11.224 description iBGP with Router5
Router2 (config-router)#
Router2 (config-router)#neighbor 210.210.19.224 remote-as 100
Router2 (config-router)#neighbor 210.210.19.224 update-source loopback 0
Router2 (config-router)#neighbor 210.210.19.224 description iBGP with Router6
Router2 (config-router)#
```

```

Router2 (config-router)#neighbor 210.210.35.224 remote-as 100
Router2 (config-router)#neighbor 210.210.35.224 update-source loopback 0
Router2 (config-router)#neighbor 210.210.35.224 description iBGP with Router7
Router2 (config-router)#
Router2 (config-router)#neighbor 220.220.7.224 remote-as 100
Router2 (config-router)#neighbor 220.220.7.224 update-source loopback 0
Router2 (config-router)#neighbor 220.220.7.224 description iBGP with Router8
Router2 (config-router)#
Router2 (config-router)#neighbor 220.220.11.224 remote-as 100
Router2 (config-router)#neighbor 220.220.11.224 update-source loopback 0
Router2 (config-router)#neighbor 220.220.11.224 description iBGP with Router9
Router2 (config-router)#
Router2 (config-router)#neighbor 220.220.19.224 remote-as 100
Router2 (config-router)#neighbor 220.220.19.224 update-source loopback 0
Router2 (config-router)#neighbor 220.220.19.224 description iBGP with Router10
Router2 (config-router)#
Router2 (config-router)#neighbor 222.222.7.224 remote-as 100
Router2 (config-router)#neighbor 222.222.7.224 update-source loopback 0
Router2 (config-router)#neighbor 222.222.7.224 description iBGP with Router11
Router2 (config-router)#
Router2 (config-router)#neighbor 222.222.11.224 remote-as 100
Router2 (config-router)#neighbor 222.222.11.224 update-source loopback 0
Router2 (config-router)#neighbor 222.222.11.224 description iBGP with Router12
Router2 (config-router)#
Router2 (config-router)#neighbor 222.222.19.224 remote-as 100
Router2 (config-router)#neighbor 222.222.19.224 update-source loopback 0
Router2 (config-router)#neighbor 222.222.19.224 description iBGP with Router13
Router2 (config-router)#
Router2 (config-router)#neighbor 222.222.35.224 remote-as 100
Router2 (config-router)#neighbor 222.222.35.224 update-source loopback 0
Router2 (config-router)#neighbor 222.222.35.224 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.

18. BGP Adjacencies. 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 100
Router2(config-router)# bgp log-neighbor-changes

```

Note: this command is only available on 11.1CC and from IOS 12.0 onwards. From 12.1 onwards, the *bgp* directive has been dropped from *log-neighbor-changes*.

Sunday, January 12, 2003

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

<code>show ip ospf</code>	: see general OSPF information
<code>show ip ospf interface</code>	: see a list of OSPF interfaces that the router sees
<code>show ip ospf neighbor</code>	: see a list of OSPF neighbours that the router sees
<code>show ip ospf database</code>	: see OSPF link state database that the router has learned
<code>show ip bgp summary</code>	: see a list of BGP peers that the router sees
<code>show ip bgp</code>	: see a list of BGP paths that the router sees
<code>show ip route</code>	: 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*?

20. Add Networks via BGP. Each Router Team will advertise the CIDR block assigned to them via BGP. For example, Router Team 1 would add:

```
Router1 (config)#router bgp 100
Router1 (config-router)#network 200.200.4.0 mask 255.255.252.0
```

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 200.200.4.0 255.255.252.0 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?

21. Add the following commands to BGP:

```
Router1 (config)#router bgp 100
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?

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

22. 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 222.222.11.224
```

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, OSPF will “load share” traffic between those paths.

```
Router1>trace router12
```

Type escape sequence to abort.

```
Tracing the route to router12.workshop.net (222.222.11.224)
```

```
 1 eth0-0.router2.workshop.net (200.200.4.2) 4 msec
  eth0-1.router13.workshop.net (200.200.6.2) 0 msec
  eth0-0.router2.workshop.net (200.200.4.2) 0 msec
 2 eth0-0.router14.workshop.net (222.222.16.2) 4 msec
  eth0-1.router14.workshop.net (200.200.10.2) 4 msec
  eth0-0.router14.workshop.net (222.222.16.2) 0 msec
 3 ser0-0.router12.workshop.net (222.222.32.2) 4 msec * 4 msec
```

```
Router1>
```

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

24. 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.

Sunday, January 12, 2003

Review Questions

1. What IP Protocol does Ping and Traceroute use?
2. Ping the IP address of your neighbour's router (for example 200.200.9.2). 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 200.200.9.1). 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 forward table?
4. What IOS show command(s) will display the router's OSPF database?
5. What IOS show command(s) will display the router's BGP route table?

CONFIGURATION NOTES

Documentation is critical! You should record the configuration at each *Checkpoint*, as well as the configuration at the end of the module.