# Module 1 - Basic Topology & Router Setup

## Lab Topology

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

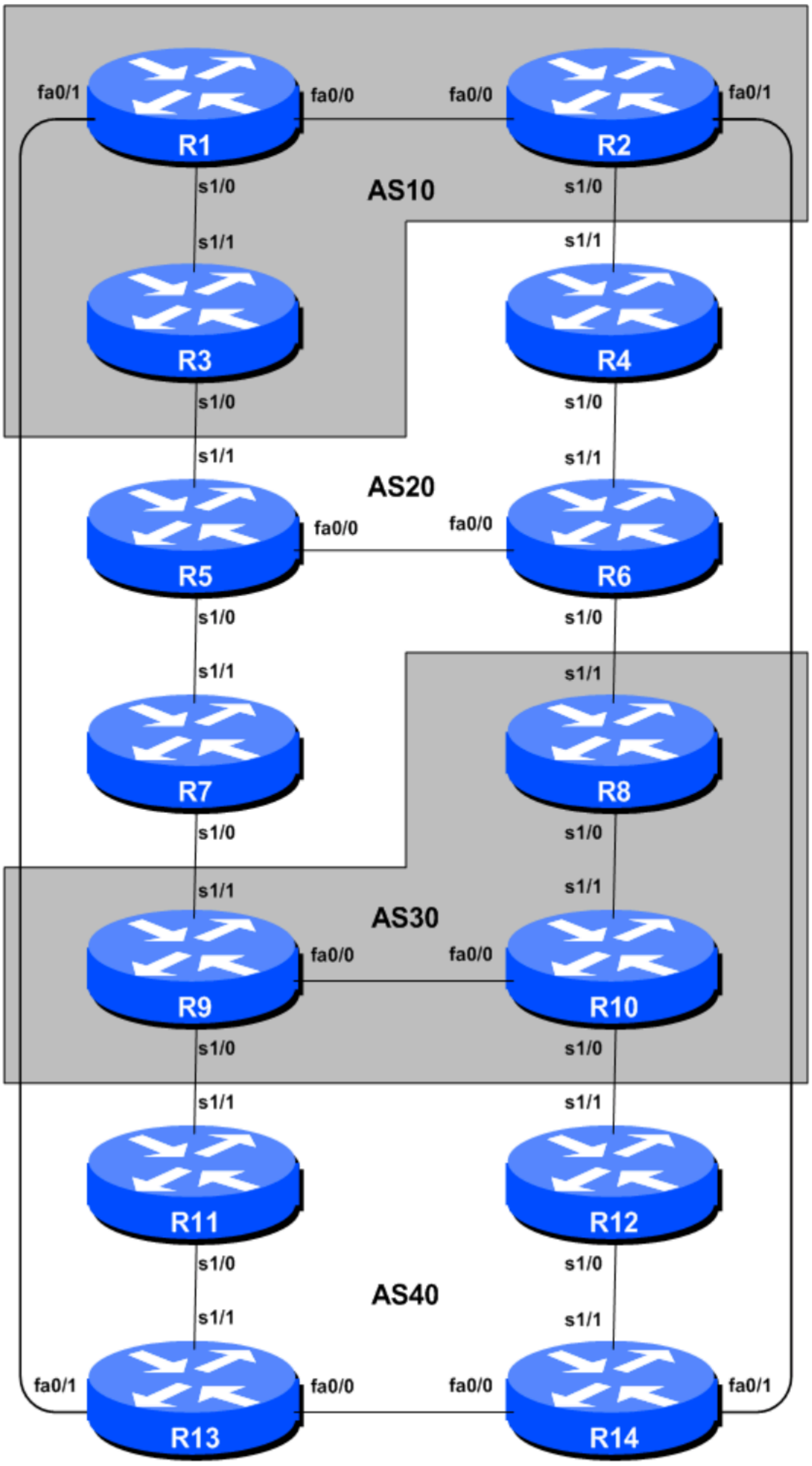

**Figure 1 – BGP AS Numbers**

## Lab Notes

This workshop is run on a Dynamips server with the appropriate lab topologies set up. The routers in the Dynamips environment are using service provider IOS.
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*** should 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 a router can see its neighbour does it make sense to start configuring routing protocols. And ***start with IGP***. There is no purpose to building BGP while the chosen IGP is not functioning properly. BGP relies on the IGPto find its neighbours and next hops, and an improperly or non-functioning IGPwill 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.

## Lab Exercise

1. **Introducing the lab**. This workshop uses Cisco IOS routers running IOS, but on the Dynamips systems – Dynamips translates the Cisco 7200 router PowerPC processor instructions in IOS to those of the host system, allowing Cisco IOS images, and therefore network configurations, to be run on a host PC system (usually Linux or MacOS based).

2. **Accessing the lab.** Make note of the IP address (IPv4, as Dynamips only supports IPv4 access) of the Dynamips server. Access to Dynamips will be by telnet, to a high port ( `2001...2014` ), which the instructor will specify.

   Telnet to the router you have been assigned:

   ```
   telnet 192.168.30.254 2001 [R1]
   telnet 192.168.30.254 2002 [R2]
   ......
   telnet 192.168.30.254 2013 [R13]
   telnet 192.168.30.254 2014 [R14]
   ```

   - If you see the initial config during router bootup, enter `no` :

     ```
     Would you like to enter the initial configuration dialog? [yes/no]:no
     ```

3. **Configure Router Hostname.** Documentation and labs will also refer to Router1 as R1.

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

4. **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. We will turn this lookup off for the labs for the time being to speed up traceroutes.

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

5. **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
```

6. **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
   ```

7. **Disable Console Logging.** The following command disables console logs and instead records all logs in a 8192 byte 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.

   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.

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

8. **Save the Configuration.**

   ```
   Router1(config)#^Z
   Router1# write memory
   Building configuration...
   [OK]
   Router1#
   ```

9. **Configure Interface IP addresses.** Study the address plan which was handed out as an addendum to this workshop module, and configure each interface.

   Example for Router2's serial interface to Router4:

```
Router2(config)# interface serial 1/0
Router2(config-if)# ip address 10.10.15.9 255.255.255.252
Router2(config-if)# description 2 Mbps Link to Router4
Router2(config-if)# no shutdown
```

10. **Verify connectivity** to each directly connected neighbouring routers.

11. **Loopback Interface Addressing.** Refering the address plan, configure loopback interfaces.

    Example for Router1:

```
Router1(config)#interface loopback 0
Router1(config-if)#ip address 10.10.15.224 255.255.255.255
```

12. **Save your configurations.**

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