Module 11 – Advanced Router Configuration

Objective: Create a basic physical lab interconnection with two autonomous systems. Each AS should use ISIS, iBGP and eBGP appropriately to construct a working network.

Prerequisites: Basic ISP Workshop (at least Modules 1 and 6 to 8)

The following will be the common topology used.

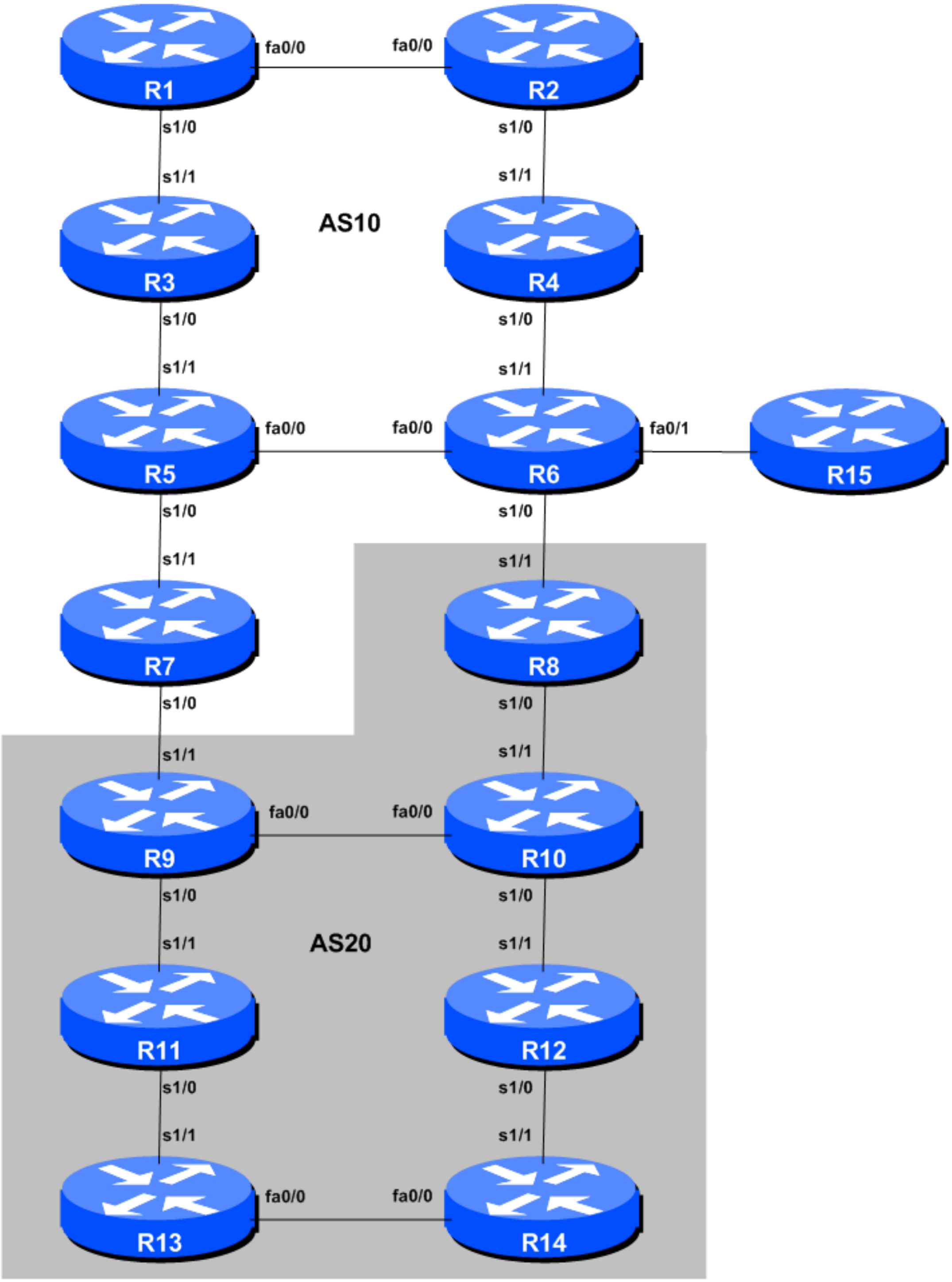


Figure 1 – ISP Lab Basic Configuration

Lab Notes

The purpose of this module is to construct the workshop lab and serve as a reminder of the basic principles of building a network, introducing an IGP, properly function iBGP, and the basics of eBGP:

- 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. Note that Router15 is the Workshop Instructor's router and it will be used at various instances throughout the workshop.

- 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.

- 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

The following list is typical for what needs to be done to bring up the lab configuration:

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.

   ```
   hostname Router1
   ```

2. **Set Domain name and 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. Unless the Workshop Instructor specifically tells you that there is a nameserver configured for the lab, we will turn this lookup off for the labs to speed up traceroutes. We will set a domain-name though, as this is required to set up SSH support later in the lab.

   ```
   no ip domain-lookup
   ip domain-name workshop.net
   ```

3. **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.

```
line con 0
 transport preferred none
line vty 0 4
 transport preferred none
```

4. **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).

```
no ip source-route
```

5. **Usernames and Passwords.** All router usernames should be ***isplab*** and all passwords should be ***lab-PW***. 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.

```
username isplab secret lab-PW
enable secret lab-PW
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).

There is the temptation to simply have a username of *cisco* and password of *cisco* as a lazy solution to the username/password problem. Under no circumstances must any service provider operator ever use easily guessable passwords as these on their live operational network[1].

6. **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.

```
aaa new-model
aaa authentication login default local
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.

[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.

7. **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. We will retain this functionality for the workshop, but it 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.

   ```
   logging buffered 8192 debugging
   ```

   which records all logs in a 8192byte buffer set aside on the router. Note that outside this workshop environment, console logging should normally be disabled as such:

   ```
   no logging console
   ```

8. **CIDRise the router.** Make sure the router is configured for CIDR. These two commands are now default in 12.0S and from 12.3 and more recent releases, but it is good practice to check just in case:

   ```
   ip subnet-zero
   ip classless
   ```

9. **Set up timestamps for all logs on the router.** 12.0S has made basic timestamping on the logs the default but ISPs should enable the complete detail on their logs as follows:

   ```
   service timestamps debug datetime localtime show-timezone msec
   service timestamps log datetime localtime show-timezone msec
   ```

   Refer to IOS Essentials or the router's on-line help system if you have forgotten what these options mean.

10. **Set up a login banner.** Next, set up a login banner. Use an appropriate greeting – consult IOS Essentials document for appropriate and inappropriate greetings. If you use an inappropriate greeting, expect the lab instructors to ask you to change it. For example

    ```
    banner login ^
    BGP Multihoming Workshop Lab
    ^
    ```

11. **Using SSH for router access.** If the router software images have SecureShell support available in them, this step will enable SSH support for access to and from the routers. You can recognise an image which has SSH in it as it will have either "k4" or "k9" in the name, signifying 3DES crypto support; for example, c2801-ik9s-mz.124-8 is a crypto IP PLUS image for the 2801 series routers.

    To enable support for SSH on the router, first the key needs to be set. To do this enter the following IOS command in configuration mode:

    ```
    crypto key generate rsa
    ```

    which will generate an RSA crypto key for the router. When the router prompts for key size, ensure that it is at least 768 bits so that SSH version 2 can be used. This key will be automatically stored in a file in NVRAM – this file is not readable by any user on the router. Next, explicitly set SSH to version 2:

```
ip ssh version 2
```

SSH is now available for use on the router.

Note: Don't use Secure Shell version 1 any more – it is no longer considered secure.

12. **Set SSH source interface.** Any network device, by default, will use the egress interface as the source IP address for outgoing traffic originated by the router. But most connectivity and monitoring of routers is targeted on the loopback interface. So we will now change the source address for SSH traffic to be that of the loopback interface address.

```
ip ssh source-interface loopback 0
```

13. **Tidy up the vty and console interface configuration.** In the real world we'd now add access-lists to the vty ports on the router. However, this lab is not connected to any external network or the Internet so these will not be required. However, we need to make some changes to the defaults. Basically, only ssh will be the supported mechanism to connect to the routers, and only ssh will be the permitted mechanism to connect from one router to the next. (Most ISPs completely disable telnet support on their routers – industry best practice considers telnet an obsolete and archaic protocol, ill-suited for use for management of public network infrastructure equipment.)

```
line vty 0 4
 transport input ssh
 transport output ssh
```

14. **Create a loopback interface.** Loopback interfaces will be used in this workshop for many things. They are an essential and fundamental requirement for any ISP backbone:

```
interface loopback 0
 description Loopback Interface for RouterXX
```

15. **Disable pad, finger and bootp servers.** The pad, finger and bootp servers are running by default in IOS. These should be disabled on any Internet router. Finger is a security risk, bootp and pad are simply unnecessary.

```
no service pad
no ip finger
no ip bootp
```

16. **Remove unneeded SNMP configuration.** IOS versions prior to 12.0S install a default SNMP configuration when the router first starts with an unconfigured NVRAM. As we will not be using SNMP to access the routers in the workshop, check if the SNMP configuration is there and remove it if it is. (Unless configured correctly SNMP is a security risk in the Internet.) Example:

```
no snmp-server community public
```

17. **Disable built-in http server.** IOS now comes with a built-in http server which is enabled by default (assists with simple installation for non-technical users). This server is completely useless

for ISPs, and being activated by default is a serious security risk. Disable it before the router receives any IP address configuration:

```
no ip http server
no ip https server
```

18. **Enable CEF.** While this is now enabled by default on most Cisco IOS router platforms (from IOS 12.4 onwards), it is still worth including the command in your configuration template.

    ```
    ip cef
    ```

    Entering the *sh cef interface* command will show the status of CEF on the router interfaces. And the command *sh ip cef* will show the forwarding table – currently empty.

19. **Path MTU discovery.** The default MTU for all communications originating from the router is 576 bytes – while this may be sufficient for most light use purposes, ISP networks tend to place larger stresses on routers. Enabling path MTU discovery on the router will ensure that the router will use the optimum (i.e. largest) MTU possible for a communication. For example, a router with several BGP neighbours and exchanging the full Internet routing table with each neighbour will be able to transfer this routing table almost 3 times faster over Ethernet or serial connections with path MTU discovery enabled (allowing 1500 byte packets) than with using the default MTU of 576 bytes.

    Enable path MTU discovery on your router:

    ```
    ip tcp path-mtu-discovery
    ```

    While there may not be much visible difference in router performance in the workshop lab, participants are encouraged to add this command to their default router configuration.

20. **IP unreachables.** When implementing BGP in an ISP network, the classic and recommended way of inserting a prefix into the BGP table is by configuring a network statement in BGP and a matching static route to the Null0 interface (the so-called pull-up route). We saw this used in Modules 1 and 6 and will see it again later on in this module.

    The benefit delivered to the ISP network by using this method is that any traffic destined for any IP address covered by that address block will have a final destination, regardless as to whether the IP address is routed on the network or not. For example, if a customer is using a /25 address range out of the ISPs /20 address block, and that customer disconnects from the Internet to allow maintenance on their connection, traffic trying to reach the /25 address block will be "caught" by the aggregate's null route. This means the traffic doesn't traverse the ISP's backbone before dying on the aggregation router, but is caught "early" on as it enters the backbone. This is operationally tidier for many ISPs, and can be less confusing for Internet users as well.

    (The static route to Null0 has many uses, and is one of the tools used frequently in helping with defeating denial of service attacks on service provider and end user networks.)

    The side effect from doing this is that the router has to send a response that the packet has reached a destination – this response is that the destination is "unreachable". Each packet generates one response – an ICMP unreachable message. For a stream of packets, this can introduce some burden on the router CPU, so many ISPs configure the Null0 interface to not send ICMP unreachables –

the packets end up at the Null0 interface and are silently discarded. This is much lighter on the router CPU.

Now disable the sending of ICMP unreachables on your router's Null0 interface:

```
interface Null 0
 no ip unreachables
```

21. **Saving the configuration.** With the basic configuration in place, save the configuration by using "write memory". Then 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. Don't forget to frequently save the configuration to NVRAM after each configuration change.

    IMPORTANT NOTE: Each router team is strongly recommended to make a copy of the basic router configuration at this stage. It will be assumed throughout this workshop that the above configuration will **ALWAYS** be present on the router. If it is not, each router team will be requested to restore it as a matter of urgency.

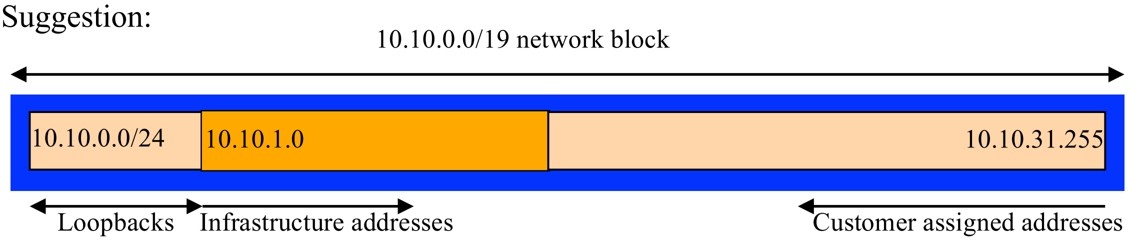
Checkpoint #1: *call lab assistant to verify the connectivity. Save the configuration as it is on the router either on the worksheet on the end of this hand out, or own your own laptop, or on the classroom tftp server if it is available. It will be required again several times throughout this workshop.*

22. **IP Addresses.** Each AS is assigned a block of IP addresses.

```
AS10      10.10.0.0/19                 AS20      10.20.0.0/19
```

    Decide among your team what the addressing plan for you AS should be.
    Hint One: point to point links only require /30 blocks.
    Hint Two: loopbacks only require a /32 host address.
    Hint Three: number your backbone sequentially, from either the start or finish of the range.

    Suggestion:



    Note: When the IP addresses are assigned, they **MUST** be annotated on the **WHITE-BOARD** at the front of the workshop room. A large network map will have been drawn on the white-board – all the IP address assignments need to be annotated there so that other Router Teams can document and understand the links and routing in this and future modules.

23. **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.

24. **Create vty filters.** Set up filters on the vty interfaces restricting vty access to your router to those addresses you would like to permit. For the purpose of this lab, even though it is not connected to the Internet, set up filters so that only the address space in the lab has access. The following example shows an extended access-list (standard and numbered access-lists can be used too):

```
ip access-list extended ipv4-vty
 remark VTY filters
 permit ip 10.10.0.0 0.0.31.255 any
 permit ip 10.20.0.0 0.0.31.255 any
 deny   ip any any log
!
line vty 0 4
 access-class ipv4-vty in
```

25. **ISIS within the same AS.** Each router Team should enable ISIS on their router. All the routers in one AS will be in ISIS level 2. The NET should be *49.0001.x.x.x.x.00*, where *x.x.x.x* is the loopback IP address. For this lab, serial interfaces need to have metric 20, fastethernet interfaces need to have metric 2. Remember to use wide metrics, and don't forget to mark the loopback interface as a passive interface – for example:

```
router isis as20
 net 49.0001.0100.2000.0001.00
 is-type level-2-only
 metric-style wide level-2
 log-adjacency-changes all
 passive-interface Loopback0
!
interface serial 1/0
 ip router isis as20
 isis circuit-type level-2 only
 isis metric 20 level-2
!
```

26. **Setting ISIS default metric for all interfaces.** Many network operators recommending setting the default ISIS metric away from the IOS value of 10 to something much larger, for example, 100000. (See the Advanced ISIS presentation.) This avoids situations where a new router would be introduced into the network and becomes the best transit path before the operators has fully configured all the links to it from other routers. Here is a configuration example:

```
router isis as20
 metric 100000
!
```

27. **DMZ between AS10 and AS20.** ISIS must **NOT** run on the demarcation links between AS10 and AS20. So Routers 6, 7, 8 and 9 must configure the serial links between each other to be passive. This is a very important point, and a mistake frequently made by many ISPs. Also, do not put any `ip isis` configuration on external facing interfaces – again see iBGP discussion at step 35. Router 6 should have two adjacencies only – with Router 4 and 5. Router 7 should have one adjacency only – with Router 9. And so on.

28. **Point-to-Point Ethernet Links.** Because Ethernet is more and more commonly used for point to point links within the PoP and on the WAN, we want to try and make the establishment and failover of ISIS just a little more efficient on these links. ISIS can be configured in point to point mode on these sorts of connections. This mode prevents the election of a DIS, prevents flooding

on the LAN (pointless as there is only one other device), and speeds up the SPF computations. To configure point to point mode, simply go to the Ethernet interface in question and configure:

```
interface fastethernet 0/0
 isis network point-to-point
!
```

29. **Intra Area Authentication – Part 1.** ISIS supports neighbour authentication. Even though ISIS runs alongside IP on the wire, some ISPs still consider neighbour authentication to be necessary. Authentication helps prevent the introduction of improperly configured or unintended equipment.

    This first step will be to set up the authentication key chains:

    ```
    key chain lab-key
     key 1
      key-string cisco
    ```

    This sets up a key chain called *lab-key* with the key string “cisco”. Obviously on a production network a key other than “cisco” should be used!

30. **Intra Area Authentication – Part 2**. Now that the key chain has been set up, the final step is to actually configure the authentication on the interface. MD5 encryption should be used rather than exchanging keys in plain text – to do this, use the *md5* sub-interface command.

    An example configuration might be:

    ```
    interface fastethernet0/0
     isis circuit-type level-2 only
     isis metric 2 level-2
     isis authentication mode md5 level-2
     isis authentication key-chain lab-key level-2
    ```

    Notice now that the ISIS adjacencies do not come up unless the neighbouring router has also entered the same configuration and key. Notice also how the ISIS adjacencies were reset as the configuration was entered – security is being introduced, so the adjacencies are reset.

31. **ISIS metrics and dependencies on BGP.** One very important feature for an ISP network is to ensure that a router starting up after maintenance does not advertise itself as a best transit path through the backbone until iBGP is up and operational. This protects the network from blackholing transit traffic, a common customer complaint during ISP maintenance.

    The way to ensure this is for when ISIS to start, it advertises all networks known to it with the maximum possible metric – this makes them highly undesirable. Once iBGP is up and running (at least one adjacency is established and converged), the ISIS metric is then returned to normal.

    ```
    router isis as20
     set-overload-bit on-startup wait-for-bgp
    ```

32. **Ping Test #2.** Ping all loopback interfaces in your AS. They should all respond. 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
```

<u>Checkpoint #2:</u> *call lab assistant to verify the connectivity. Save the configuration as it is on the router either on the worksheet on the end of this hand out, or own your own laptop, or on the classroom tftp server if it is available.*

STOP AND WAIT HERE

33. **BGP distance.** 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>
    ```

    Note: This should be included in all future BGP configurations in this workshop. Set the BGP protocol distance so that BGP is always less preferred than any IGP. So:

    ```
    router bgp 10
     distance bgp 200 200 200
    ```

34. **Passwords on BGP sessions.** It is now considered very good practice to use passwords on the BGP sessions on the router. When BGP is set up in the next step, don't forget to include a password on the BGP peering.

    The password used for this module will be *cisco* – obviously on a real operational network operators will use a password which follows their normal password rules, and not something which is easily guessable. An example configuration might be:

    ```
    router bgp 10
     neighbor 1.2.3.4 password cisco
    ```

    Note: Passwords should be included in all future BGP configurations in this workshop.

35. **Configuring next-hop-self on iBGP Neighbours.** So that BGP has a valid next-hop for external destinations, we introduce the `next-hop-self` BGP configuration. This changes the iBGP default by replacing the next-hop address for external sites from that of the external neighbour address to

[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.

the loopback address of the local router. The local router knows how to get to the external destinations because it is connected to the LAN that leads there – the rest of the network internal to the AS is told simply to go via this router. Note that because we do this, we no longer need to quote the external point to point link in our ISIS (or OSPF) configuration – see steps 24 & 26 earlier. For example:

```
router bgp 10
 neighbor 1.2.3.4 next-hop-self
```

Note that the use of *next-hop-self* on all iBGP sessions is considered industry best practice, and its use from now on in the workshop is strongly recommended.

36. **Configuring iBGP Neighbours.** Configure iBGP peers within each autonomous system. Use a full iBGP mesh. Don't forget that iBGP peering is configured to be between the loopback interfaces on the routers. Also, it is good practice to use a peer-group. For example:

```
router bgp 10
 neighbor ibgp-peers peer-group
 neighbor ibgp-peers remote-as 10
 neighbor ibgp-peers description iBGP v4 peergroup
 neighbor ibgp-peers update-source loopback 0
 neighbor ibgp-peers next-hop-self
 neighbor ibgp-peers password cisco
 neighbor ibgp-peers send-community
 neighbor 10.10.0.1 peer-group ibgp-peers
 neighbor 10.10.0.2 peer-group ibgp-peers
 neighbor 10.10.0.3 peer-group ibgp-peers
..etc..
```

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. Note: get into the habit of using peer-groups and configuring them fully, including the "send-community" directive (Cisco IOS by default does not send BGP communities to BGP neighbours). This workshop makes extensive use of communities, and making them part of your configuration is good practice.

Note: Router6 should also include the network connecting to Router15 in the iBGP configuration. This is so that the network connected to Router15 can be accessed – it has the DNS server and NTP server located on it.

37. **Add Prefixes to BGP.** Each Router Team will advertise the CIDR block assigned to them via BGP. AS10 would advertise 10.10.0.0/19 and AS20 would advertise 10.20.0.0/19:

```
router bgp 10
 no synchronization
 no auto-summary
 bgp log-neighbor-changes
 network 10.10.0.0 mask 255.255.224.0
!
ip route 10.10.0.0 255.255.224.0 null0
```

Don't forget the static route to Null0. This ensures that the prefix has an entry in the routing table, and therefore will appear in the BGP table. Also, don't forget to disable synchronisation and auto-summarisation – these are also mandatory requirements for ISP routers connecting to the Internet.

(Note that a distance of 250 could be applied to the static route to ensure that routing protocols announcing this exact prefix will override the static (if this is required/desired).)

Checkpoint #3: *call the lab assistant to verify the connectivity.*

38. **Enable new format of BGP communities.** It is also worth getting into the habit of changing the BGP community format from the default 32-bit integer to colon separated 16-bit integers, as used in RFC1998. Example:

```
ip bgp-community new-format
```

39. **Configure eBGP peering.** Now that iBGP is functioning, it is time to configure eBGP. External BGP will be set up between AS10 and AS20, specifically between Routers 6 and 8, and Routers 7 and 9 only. The remaining lab teams should monitor the BGP table they see on their routers.

    Firstly, agree on what IP addresses should be used for the point-to-point links between the ASes. Put the /30 networks used for the DMZ links into OSPF (network statement and passive interface). Then configure eBGP between the router pairs, for example:

```
router bgp 10
 neighbor 10.20.2.2 remote-as 20
 neighbor 10.20.2.2 password cisco
 neighbor 10.20.2.2 description eBGP with RouterXX
```

    Use the BGP show commands to ensure that you are receiving prefixes from your neighbouring AS.

40. **eBGP filters.** The final step of setting up the eBGP sessions is to apply filters on the BGP sessions. The teams operating Routers 6, 7, 8 and 9 should now set up filters to allow their neighbour AS prefixes into their network; and set up filters to allow their own AS prefixes out of their network. For example:

```
router bgp 10
 neighbor 10.20.2.2 prefix-list AS20peer-in in
 neighbor 10.20.2.2 prefix-list AS20peer-out out
!
ip prefix-list AS20peer-in permit 10.20.0.0/19
ip prefix-list AS20peer-out permit 10.10.0.0/19
!
```

    This only allows 10.20.0.0/19 in from an AS20 eBGP neighbour, and allows 10.10.0.0/19 out to an AS20 eBGP neighbour. Don’t forget that you will also need to allow the address of the NTP timeserver through the BGP filters as well.

41. **BGP Communities on eBGP sessions.** Network operators only send communities to their eBGP peers if they have a policy requirement to do so. Cisco IOS also does not send communities on the eBGP sessions by default, so we will now add this to eBGP peerings. The teams operating Routers 6, 7, 8 and 9 should add the “send-community” directive to their eBGP peerings. For example:

```
router bgp 10
 neighbor 10.20.2.2 send-community
!
```

42. **Check the network paths and the routing table.** Run traceroutes between your router and other routers in the classroom. Ensure that all routers are reachable. If any are not, work with the other router teams to establish what might be wrong. Make sure that you can see Router15. The lab instructor will have written the addresses and network up on the whiteboard. (The network is 192.168.1.0/24, the address of Router6 on that LAN is 192.168.1.254, and the address of Router15 is 192.168.1.1.)

43. **Saving the configuration. For software releases from 12.0 onwards**, the commands to save the configuration are of the format *copy <source> <destination>* where the source and destinations can be any of the following options: *ftp, lex, null, nvram, rcp, running-config, startup-config, system, tftp.* To save the configuration to the TFTP server, use the "*copy system:/running-config tftp:*" command sequence. If the TFTP server is unreachable, "."s followed by an error message will be displayed rather than "!"s. (Note that the "*write net*" command of earlier releases is still supported but may be removed at a future release.)

    An example of saving the configuration for Router 1 might be:

    ```
    Router1#copy system:running-config tftp:
    Address or name of remote host[]? 192.168.1.4
    Destination filename [running-config]? router1-confg
    !!
    2259 bytes copied in 2.920 secs (1129 bytes/sec)
    Router1#
    ```

Checkpoint #4: *call the lab assistant to verify the connectivity.*

44. **Summary.** This module has covered most of the fundamental configuration topics required to construct an ISP network. It has covered basic router configuration, configuration Best Current Practices, ISIS configuration, iBGP configuration, and finally simple eBGP configuration. No routing policy has been implemented. **Each Router team is strongly recommended to make a copy of their configuration, as most of the configuration concepts will be required throughout the remainder of the workshop.**