**250 CT Portfolio Exercise 1**

Academic year 2020/2021

**Read me 1st**

This activity is part of the coursework portfolio for this module. You must do portfolio exercises in the order provided. Ensure each activity is completed prior to attempting the next one. Ideally students should work in **groups of two or three**.

**1. Basic connectivity**

As part of this activity students are required to connect the computers with networking devices using the Cisco Packet Tracer.

**1.1 Connect a PC to a console port of a router**

For this exercise you need to establish a link between the **serial port** of a PC and a **console port** of a router in the equipment. To achieve this you will need deploy a **roll-over** (also known as console) cable to connect from the PC to the **console port** of the router.

Below list the settings of the serial communication:

|  |  |
| --- | --- |
| Bits per-second: | **9600** |
| Data-bits: | **8** |
| Parity: | **no parity** |
| Stop bits: | **1** |
| Flow control: | **no flow control** |

If the connection was successful, a command prompt or a login request should appear on the PC. Please provide **screenshot(s)** as the evidence of your achievement.

|  |
| --- |
|  |

**1.2 Simple point to point network between two PCs**

Set up a simple point to point network by direct connection using a crossover cable to connect **the Ethernet adaptors** of the two PCs together. Configure the PCs using the following addresses, where **X** is selected so that PC1 and PC2 don’t share the same IP address:

IP address for PC 1: 192.10.2.X

IP address for PC 2: 192.10.2.X

Subnet mask: 255.255.255.0

To complete this task you need to **evidence via screeshots** that:

1. The PCs are configured properly by typing “**ipconfig**” in the command prompt.
2. The PCs can communicate with each other by “pinging” PC1 from PC2 and vice versa using the “**ping <ip address of other PC>”** command in the command prompt.

|  |
| --- |
| PC1 > PC0    PC0 > PC1 |

**1.3 Simple network using a hub/switch**

Set up a simple network using a hub/switch via the Cisco Packet Tracer,

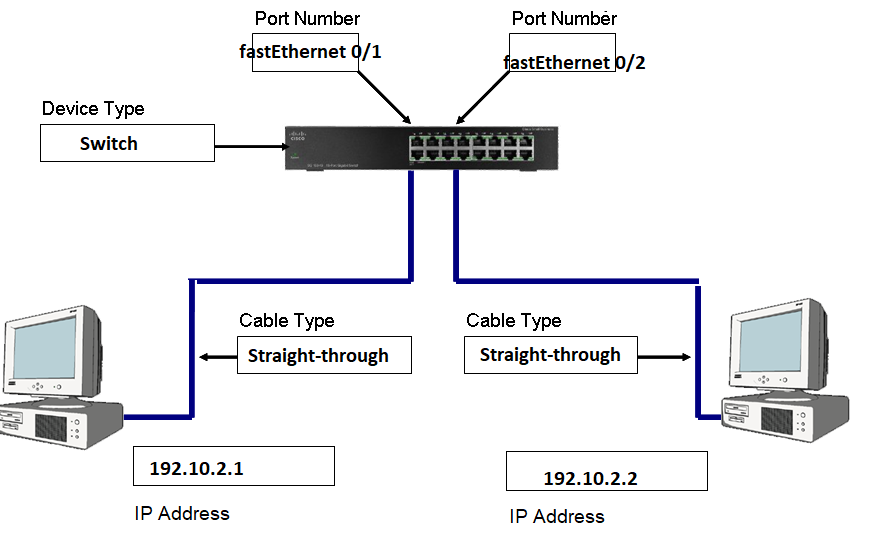
Please demonstrate to the instructor that your network works as you did in the previous exercise (use the same IP addresses as in 1.2), and further provide the evidences of your work as shown below:

* Please provide, as evidence, the screenshot(s) of your system on the Cisco Packet Tracer.

|  |
| --- |
| PC-0 > PC-1    PC-1 > PC-0 |

* Please fill in the diagram provided.

**Note:** Please fill in the missing information in the following diagram – this will be marked and needs to be submitted alongside the reset of the portfolio exercises.



Wiring Diagram for Activity 3 – Please fill in the gaps

**250 CT Portfolio Exercise 1**

**Submission Instructions**

**Group Members**

Ideally these tasks should be completed in groups of two. Groups of three are allowed but not recommended. Please indicate the group members for this portfolio exercise.

Name: James Thomas Student ID: 9195071

Name: Alexander Collins Student ID: 9442540

Name: Liam Smith Student ID: 8452270

**Portfolio Submission**

You are required to **provide the evidence and fill in the portfolio exercise sheets**. Those are **to be submitted later** (all portfolio exercises to be submitted together) as indicated in the coursework description. The completion level and correctness of the submitted portfolio exercise sheets contributes towards you mark.

**250 CT Portfolio Exercise 2**

Academic year 2020/2021

**Read me 1st**

This activity is part of the coursework portfolio for this module. You must do portfolio exercises in the order provided. Ensure each activity is completed prior to attempting the next one. Ideally students should work in **groups of two or three**.

**1. Addressing Scheme**

As a network administrator you are required to design the addressing scheme for the network shown in diagram 1. Subnet the given address (**192.168.10.0/24**) using **normal** (as opposed to variable) subnet masking and assign a subnet to each part of the network. Finally decide which IP address will be allocated to each port of each device.

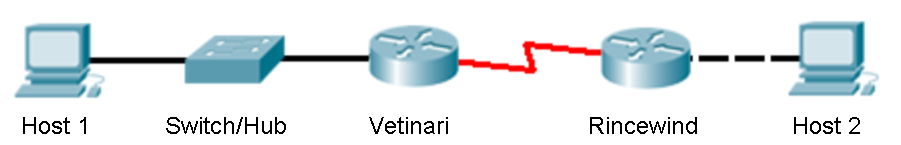


Diagram 1: The Network

* 1. **Subnets:** Calculate the subnets and fill in the table below. Remember that for this exercise you are only allowed to use **normal** subnetting.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Subnet** | **First host** | **Last host** | **Broadcast** | **Subnet mask** |
| 1 | 192.168.10.0 | 192.168.10.1 | 192.168.10.30 | 192.168.10.31 | 255.255.255.224 |
| 2 | 192.168.10.32 | 192.168.10.33 | 192.168.10.62 | 192.168.10.63 | 255.255.255.224 |
| 3 | 192.168.10.64 | 192.168.10.65 | 192.168.10.94 | 192.168.10.95 | 255.255.255.224 |
| 4 | 192.168.10.96 | 192.168.10.97 | 192.168.10.126 | 192.168.10.127 | 255.255.255.224 |
| 5 | 192.168.10.128 | 192.168.10.129 | 192.168.10.158 | 192.168.10.159 | 255.255.255.224 |
| 6 | 192.168.10.160 | 192.168.10.161 | 192.168.10.190 | 192.168.10.191 | 255.255.255.224 |

* 1. **Device IP addresses:** Decide which subnet will be allocated to each network part, and allocate IP addresses to device ports.

|  |  |  |  |
| --- | --- | --- | --- |
| **Device** | **Port** | **IP address** | **Subnet mask** |
| Vetinari | Ethernet 0/0 | 192.168.10.1 | 255.255.255.224 |
| Vetinari | Serial 0/2/0 | 192.168.10.33 | 255.255.255.224 |
| Rincewind | Ethernet 0/0 | 192.168.10.65 | 255.255.255.224 |
| Rincewind | Serial 0/2/0 | 192.168.10.34 | 255.255.255.224 |
| Host 1 | Ethernet | 192.168.10.2 | 255.255.255.224 |
| Host 2 | Ethernet | 192.168.10.66 | 255.255.255.224 |

**2. Cabling**

For this task you are required to wire up the devices in the way illustrated by the network diagram (diagram 1). As this is similar to what we did in the **portfolio 1 exercise**, step by step instructions will **not** be provided. However the following few bullet points provide some additional help.

* The link between the two routers (Vetinary and Rincewind) is a **serial** cable – one side has to be the **DCE**, meaning that it will provide the clock rate. Connect the DCE side (should be labelled as such) to Rincewind.
* Routers are **specialised computers** and as such, a **crossover** cable must be used for a direct computer to router connection (such as the one between Host 2 and Rincewind).

In addition to the network connections shown in diagram 1 we also need **two extra connections** for the purpose of configuring the routers. For those two connections you are required to connect from the **serial port** of the computer (not the Ethernet port) to the **console port** of the router (from Host 1 to Vetinari, and from Host 2 to Rincewind). Detailed instructions of how to do this were given as part of the first portfolio exercise.

After finishing cabling, please provide **screenshot(s)** as the evidence of your achievement.

|  |
| --- |
| Evidence 1: |

**3. Basic router commands**

Assuming that the last part of the previous task was completed successfully you should be able to access the router prompt using the **Serial** port on a PC.

Cisco routers provide a number of **different modes** (states). Depending on those modes there are different things you can or can’t do. The initial mode is called the **user-Exec mode** and provides only a limited set of commands. This mode is indicated by using the character “>” as part of the prompt. For instance:

Router>

Typing “?” in the prompt should display a list of the commands available. Typing the name of a command followed by “?” displays the parameters this command expects. Copy some of the commands available along with the explanations provided in the following table.

|  |  |  |
| --- | --- | --- |
|  | Command | Function |
| 1 | connect | Open a terminal connection |
| 2 | disable | Turn off privileged commands |
| 3 | disconnect | Disconnect an existing network connection |
| 4 | enable | Turn on privileged commands |
| 5 | exit | Exit from the EXEC |
| 6 | logout | Exit from the EXEC |
| 7 | ping | Send echo messages |
| 8 | resume | Resume an active network connection |

To do anything useful we need access to the **privilege mode** (basically akin to logging in to a computer as admin). We can do that using the enable command. Type “enable” and press enter. If a password is asked for use either “cisco” or “class”. If successful the prompt should change somehow, with the character “#” shown to indicate we are in privilege mode.

Router#

Type “?” to see the commands available to you now (the list should be longer). Copy some of those and their explanations to the table provided.

|  |  |  |
| --- | --- | --- |
|  | Command | Function |
| 1 | Clock | Manage the system clock |
| 2 | Dir | List files on a filesystem |
| 3 | Copy | Copy from one file to another |
| 4 | Erase | Erase a filesystem |
| 5 | More | Display the contents of a file |
| 6 | No | Disable debugging information |
| 7 | Reload | Halt and perform a cold restart |
| 8 | Clear | Reset functions |

The configuration of the router is stored in the startup-config file. When booting the router it is copied to memory. Typing “show running-config” in the privilege mode displays the current configuration the router has loaded in its memory (note that since it might have been changed since the router last booted, the running-config is not necessarily the same as the startup-config). Try this command and inspect the configuration (pressing space to go to the next page). What sort of useful information can be found there?

We know that the router has no service password-encryption, the hostname is ‘Router’, the IP addresses of each interface that’s been configured, whether the interface is up or down, the clock rate of the serial connection and the duplex mode of interfaces.

Before we start configuring routers it is a good idea to delete the previous configuration. Type “erase startup-config” to delete the configuration file from the secondary storage. Then restart the router using the “reload” command. After reboot the router might ask you if you want to enter the initial configuration dialog. Answer **NO** to this question.

**Some tips:** Writing enough characters for the command to be uniquely identified is enough – “sh run” will work just as well as “show running-config”. Pressing the tab character while typing a command will try to guess what you are trying to type and complete the command for you. The up and down arrow keys allow you to access commands previously typed – so no need to type the same command again and again.

**4. Basic router configuration**

Before you start this part make sure that the router **has no previous configuration set**. This can be achieved by erasing the startup-config and reloading the device (last part of the previous task). For this part you are required to configure the two routing devices (Vetinari and Rincewind). In order to configure routers you need to be in the router **global configuration mode** – the prompt will look like this:

Router(config)#

You can achieve that by typing “enable” to get into privilege mode, and then typing “configure terminal” to enter the global configuration mode. Moving back to the previous mode (if you need to do so) can be achieved by typing exit.

**4.1 Router names:** First task is to configure the router names. This can be easily achieved with the hostname command:

Router(config)# hostname *name*

**Where** *name* is the name of the device (Vetinary or Rincewind depending on which router you are configuring). If successful the prompt should change accordingly (can also verify by typing “**exit**” to go back to the privilege mode and using the “show running-config” there).

**4.2 Interfaces:** The next part is configuring the router interfaces. This can be accomplished by entering **interface configuration mode** from the global configuration mode and using the “IP address” command followed by the address and subnet mask you wish to configure. Use whatever IP address and subnet masks you assigned to these interfaces in the first part of the exercise. Write those down on the blank spaces provided in the following command sequences.

// Ethernet port for **Vetinari**:

Vetinari(config)# interface *fa0/0*

Vetinari(conf-if)# ip address 192.168.10.1 255.255.255.224

Vetinari(conf-if)# no shutdown

// Ethernet port for **Rincewind**:

Rincewind(config)# interface fa0/0

Rincewind(conf-if)# ip address 192.168.10.65 255.255.255.224

Rincewind(conf-if)# no shutdown

// Serial port for **Vetinari**:

Vetinari(config)# interface *s0/2/0*

Vetinari(conf-if)# ip address 192.168.10.33 255.255.255.224

Vetinari(conf-if)# no shutdown

// Serial port for **Rincewind**:

Rincewind(config)# interface *s0/2/0*

Rincewind(conf-if)# ip address 192.168.10.34 255.255.255.224

Rincewind(conf-if)# clock rate 56000

Rincewind(conf-if)# no shutdown

Notice that the above command sequences make some **assumptions** regarding router port names. Those might not be entirely accurate. You can find out exactly how many ports each router has and how they are called (s0/2/0 or s0/0 or whatever) by typing the “show protocols” command in the privilege mode. For instance:

Vetinari# show protocols

This command is also useful for **troubleshooting** as it will tell you if the router ports are now operational or not (in which case check you cabling).

**4.3 Setting up passwords**: Clearly in a real environment we do not want unauthorised personnel been able to change our configurations. The following commands can be used to setup some basic security in our routers. Apply them to **both Vetinari and Rincewind**.

// Passwords for telnet session (enter **line configuration mode**)

Router(config)# line vty 0 4

Router(config-line)# password cisco

Router(config-line)# login

// Console and privilege mode passwords

Router(config)#enable password cisco

Router(config)#enable secret class

After you finish you might want to inspect the router’s running config files again in order to see the changes.

Please provide the result of “show running-config” as the evidence of your achievement.

|  |
| --- |
| Evidence 2-1  **Vetinari:**  Building configuration...  Current configuration : 796 bytes  !  version 12.4  no service timestamps log datetime msec  no service timestamps debug datetime msec  no service password-encryption  !  hostname Vetinari  !  !  !  enable secret 5 $1$mERr$9cTjUIEqNGurQiFU.ZeCi1  enable password cisco  !  !  !  !  !  !  no ip cef  no ipv6 cef  !  !  !  !  !  !  !  !  !  !  !  !  spanning-tree mode pvst  !  !  !  !  !  !  interface FastEthernet0/0  ip address 192.168.10.1 255.255.255.224  duplex auto  speed auto  !  interface FastEthernet0/1  no ip address  duplex auto  speed auto  shutdown  !  interface Serial0/1/0  no ip address  clock rate 2000000  shutdown  !  interface Serial0/1/1  ip address 192.168.10.33 255.255.255.224  !  interface Vlan1  no ip address  shutdown  !  ip classless  !  ip flow-export version 9  !  !  !  !  !  !  !  !  line con 0  !  line aux 0  !  line vty 0 4  password cisco  login  !  !  !  end  **Ricewind:**  Building configuration...  Current configuration : 817 bytes  !  version 12.4  no service timestamps log datetime msec  no service timestamps debug datetime msec  no service password-encryption  !  hostname Ricewind  !  !  !  enable secret 5 $1$mERr$9cTjUIEqNGurQiFU.ZeCi1  enable password cisco  !  !  !  !  !  !  no ip cef  no ipv6 cef  !  !  !  !  !  !  !  !  !  !  !  !  spanning-tree mode pvst  !  !  !  !  !  !  interface FastEthernet0/0  ip address 192.168.10.65 255.255.255.224  duplex auto  speed auto  !  interface FastEthernet0/1  no ip address  duplex auto  speed auto  shutdown  !  interface Serial0/1/0  ip address 192.168.10.34 255.255.255.224  clock rate 56000  !  interface Serial0/1/1  no ip address  clock rate 2000000  shutdown  !  interface Vlan1  no ip address  shutdown  !  ip classless  !  ip flow-export version 9  !  !  !  !  !  !  !  !  line con 0  !  line aux 0  !  line vty 0 4  password cisco  login  !  !  !  end |

Please provide the result of “show protocols” as the evidence of your achievement.

|  |
| --- |
| Evidence 2-2  **Vetinari:**  Global values:  Internet Protocol routing is enabled  FastEthernet0/0 is up, line protocol is up  Internet address is 192.168.10.1/27  FastEthernet0/1 is administratively down, line protocol is down  Serial0/1/0 is administratively down, line protocol is down  Serial0/1/1 is up, line protocol is up  Internet address is 192.168.10.33/27  Vlan1 is administratively down, line protocol is down  **Ricewind:**  Global values:  Internet Protocol routing is enabled  FastEthernet0/0 is up, line protocol is up  Internet address is 192.168.10.65/27  FastEthernet0/1 is administratively down, line protocol is down  Serial0/1/0 is up, line protocol is up  Internet address is 192.168.10.34/27  Serial0/1/1 is administratively down, line protocol is down  Vlan1 is administratively down, line protocol is down |

**5. Making it all work**

It is time to configure the hosts and the routing protocol on the routers and test our network. We start by configuring the two hosts.

**5.1 Configuring the computers:** Setup the IP address for the two hosts (like you did for the first exercise) and use the “ipconfig” command in the DOS prompt (start->run-> cmd). Note that the **default gateway** for the two hosts will be the IP address for the Ethernet port of the router they are connected to (if I do not know where a network is, then I send the message to the router).

After you configured the computers try to ping your default gateway from each host. Was it successful?

**Yes**

Now try to ping the other host on the other end of the network. Was it successful?

**No**

**Note** that you can also use the ping command in the privilege mode of the routers. Try to ping the serial port of Vetinari from Rincewind and vice versa. Was it successful?

**Yes**

**5.2 Configuring RIP:** It is time to configure the routing protocol so that routers can exchange information about networks with each other. We are going to use the RIP routing protocol for this exercise. Type “show ip route” on the privilege mode of each router. What is the output of this command?

**Vetinari:**

Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP

i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area

\* - candidate default, U - per-user static route, o - ODR

P - periodic downloaded static route

Gateway of last resort is not set

192.168.10.0/27 is subnetted, 2 subnets

C 192.168.10.0 is directly connected, FastEthernet0/0

C 192.168.10.32 is directly connected, Serial0/1/1

**Ricewind:**

Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP

i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area

\* - candidate default, U - per-user static route, o - ODR

P - periodic downloaded static route

Gateway of last resort is not set

192.168.10.0/27 is subnetted, 2 subnets

C 192.168.10.32 is directly connected, Serial0/1/0

C 192.168.10.64 is directly connected, FastEthernet0/0

Each router has a number of networks **directly connected** to it (it knows about them as they are connected to **its own interfaces**). Those are **the networks** it needs to ***advertise*** to other routers, therefore the ones that should be used when configuring the routing protocol. Use this info to fill in the gaps provided in the instruction sequences that follow.

Vetinari(config)# router rip

Vetinari(config-router)# no auto-summary

Vetinari(config-router)# network 192.168.10.0

Vetinari(config-router)# network 192.168.10.32

Rincewind(config)# router rip

Rincewind(config-router)# no auto-summary

Rincewind(config-router)# network 192.168.10.32

Rincewind(config-router)# network 192.168.10.64

Type the “show ip route” command in the privilege mode again. What has changed (if nothing you done something wrong)?

The subnet from the other side of the network is now in the IP route, so the routers are now aware of each other’s subnets.

**6. Testing the network**

At this point the network **should be fully operational**. In order to verify it we need to make a number of tests.

**6.1** **Ping** Host 1 from Host 2 and vice versa.

…………………………………………………………………………………………………..

Please provide **screenshot(s)** as the evidence of your achievement.

|  |
| --- |
| Evidence 3  **Host 1:**    **Host 2:** |

**6.2** **Ping** both host from each router.

…………………………………………………………………………………………………..

Please provide **screenshot(s)** as the evidence of your achievement.

|  |
| --- |
| Evidence 4  **Vetinari:**    **Ricewind:** |

**6.3** **Trace the route** from host 1 to host 2 from the command prompt (tracert xx.xx.xx.xx) command (replace x with the host 2 IP address).

…………………………………………………………………………………………………..

Please provide **screenshot(s)** as the evidence of your achievement.

|  |
| --- |
| Evidence 5 |

**6.4** **Telnet** to Vetinari from Host 2

………………………………………………………………………………………………......

Please provide **screenshot(s)** as the evidence of your achievement.

|  |
| --- |
| Evidence 6 |

After you finish you might want to **save the configurations** for the devices to a flash drive for future reference. You can either do so by cut and pasting the output of the show run command to notepad or by using the **capture** facility offered by the HyperTerminal application. You might also want to reload (reboot the router) in order to erase your work and restore the router to factory settings.

**250 CT Portfolio Exercise 2**

**Submission Instructions**

**Group Members**

Ideally this task should be completed in groups of two. Groups of three are allowed but not recommended.

Name: James Thomas Student ID: 9195071

Name: Alex Collins Student ID: 9442540

Name: Liam Smith Student ID: 8452270

**Portfolio Submission**

You are required to **fill in the portfolio exercise sheets**. Those are **to be submitted later** (all portfolio exercises to be submitted together) as indicated by the module guide/schedule. The completion level and correctness of the submitted portfolio exercise sheets contributes towards you mark.

**250 CT Portfolio Exercise 3**

Academic year 2020/2021

**Read me 1st**

This activity is part of the coursework portfolio for this module. **This particular activity does not need to be signed off** – **the router configurations and screenshots that demonstrate end to end connectivity need to be submitted instead.** You must do portfolio exercises in the order provided. Ensure each activity is completed prior to attempting the next one. Ideally students should work in **groups of two or three**.

**The Problem**

As a first grade technician for the mining company Red Dwarf you have been assigned with the task of designing the addressing scheme for their network (see diagram). In order to do so you have to **subnet the network 192.168.42.0/24** ensuring that there will be enough addresses to accommodate the needs of each department. Note that the network connects to the outside world through network 172.17.0.0/16 – you can choose an IP address for this connection (ISP’s IP is 172.17.0.1) but it has to be within the same address space.

Lister

Rimmer



Officers

(28 Hosts)

Scavengers

(72 hosts)

Canaries

(49 hosts)

172.17.0.1/16

ISP

Network Diagram

**1. Addressing Scheme**

Design the addressing scheme for the network above using VLSM subnet masking. You are allowed to use subnet zero[[1]](#footnote-1).

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Subnet** | **First host** | **Last host** | **Broadcast** | **Mask** |
| Router link | 192.168.42.224 | 192.168.42.225 | 192.168.42.226 | 192.168.42.227 | 255.255.255.252 |
| Officers | 192.168.42.192 | 192.168.42.193 | 192.168.42.222 | 192.168.42.223 | 255.255.255.224 |
| Canaries | 192.168.42.128 | 192.168.42.129 | 192.168.42.190 | 192.168.42.191 | 255.255.255.192 |
| Scavengers | 192.168.42.0 | 192.168.42.1 | 192.168.42.126 | 192.168.42.127 | 255.255.255.128 |

**2. Basic Configuration**

For this task you are required to connect the devices using the appropriate cables and configure the interfaces. The commands to do that can be found in the previous portfolio exercise.

**3. EIGRP**

EIGRP[[2]](#footnote-2) routing protocol configuration is similar to that of RIP (see portfolio 2) but at the same time differs in a number of significant ways. Routers can support multiple instances of this protocol running at the same time, forming different autonomous systems. As such a positive number must be specified when configuring. This number has to be **the same for all routers** participating to that autonomous system (and thus can exchange information about the networks that form this system).

Router(config)# router EIGRP number

Another difference is that EIGRP expects hosts bits to be specified when adding networks during the routing protocol configuration. Host bit are indicated using a wildcard mask. Inverting the bits of the subnet mask can be used to derive the wildcard mask for each subnet. For instance: network 192.168.177.128 with subnet mask 255.255.255.192 will have 0.0.0.63 as its wildcard (192=110000002 while 63=001111112). This network could be added to the list of networks the router advertises using the following instruction:

Router(config-router)# network 192.168.177.128 0.0.0.63

Configure the EIGRP protocol for both routers and verify that the configuration using the “show ip route” command and/or pinging addresses in other, not directly connected parts of you network.

**4. Static route**

It is impossible for a router to have an entry for every network out there. A default gateway can be configured on router Rimmer to forward traffic towards unknown networks. This can be specified as a static route which redirects any network (0.0.0.0) with any subnet mask (0.0.0.0) we do not know about out of a particular interface (for instance fa0/1). The command to do that would look a bit like:

Router(config)# ip route 0.0.0.0 0.0.0.0 fa0/1

Following the configuration of that static route you might want to ensure that other routers in you autonomous system know about this route. This can be achieved by adding the “redistribute static” command to your router EIGRP configuration.

Router(config-router)# redistribute static

After configuring the gateway try to device a test to verify that this is working as expected.

|  |  |
| --- | --- |
| Ping from Canneries to Scavengers, Officers and ISP |  |
| Ping from Scavengers to Canneries, Officers and ISP |  |
| Ping from Officers to Scavengers, Canneries and ISP |  |
| Rimmer running config | Rimmer#show run  Building configuration...  Current configuration : 968 bytes  !  version 15.1  no service timestamps log datetime msec  no service timestamps debug datetime msec  no service password-encryption  !  hostname Rimmer  !  !  !  !  !  !  !  !  no ip cef  no ipv6 cef  !  !  !  !  license udi pid CISCO1941/K9 sn FTX15246IH6-  !  !  !  !  !  !  !  !  !  !  !  spanning-tree mode pvst  !  !  !  !  !  !  interface GigabitEthernet0/0  ip address 192.168.42.194 255.255.255.224  duplex auto  speed auto  !  interface GigabitEthernet0/1  ip address 172.17.0.2 255.255.0.0  duplex auto  speed auto  !  interface Serial0/1/0  ip address 192.168.42.226 255.255.255.252  clock rate 2000000  !  interface Serial0/1/1  no ip address  clock rate 2000000  shutdown  !  interface Vlan1  no ip address  shutdown  !  router eigrp 8  redistribute static  network 192.168.42.192 0.0.0.31  network 192.168.42.224 0.0.0.3  network 172.17.0.0    !  ip classless  ip route 0.0.0.0 0.0.0.0 GigabitEthernet0/1  !  ip flow-export version 9  !  !  !  !  !  !  !  !  line con 0  !  line aux 0  !  line vty 0 4  login  !  !  !  end |
| Lister Running config | Lister#show run  Building configuration...  Current configuration : 899 bytes  !  version 15.1  no service timestamps log datetime msec  no service timestamps debug datetime msec  no service password-encryption  !  hostname Lister  !  !  !  !  !  !  !  !  no ip cef  no ipv6 cef  !  !  !  !  license udi pid CISCO1941/K9 sn FTX15241X7V-  !  !  !  !  !  !  !  !  !  !  !  spanning-tree mode pvst  !  !  !  !  !  !  interface GigabitEthernet0/0  ip address 192.168.42.130 255.255.255.192  duplex auto  speed auto  !  interface GigabitEthernet0/1  ip address 192.168.42.2 255.255.255.128  duplex auto  speed auto  !  interface Serial0/1/0  ip address 192.168.42.225 255.255.255.252  !  interface Serial0/1/1  no ip address  clock rate 2000000  shutdown  !  interface Vlan1  no ip address  shutdown  !  router eigrp 8  network 192.168.42.128 0.0.0.63  network 192.168.42.0 0.0.0.127  network 192.168.42.224 0.0.0.3    !  ip classless  !  ip flow-export version 9  !  !  !  !  !  !  !  !  line con 0  !  line aux 0  !  line vty 0 4  login  !  !  !  end |

**Important:** You need to **submit screenshots that demonstrate connectivity**. You can either do so by cut and pasting the output of the show run command to notepad or by using the **capture** facility offered by the HyperTerminal application. Clearly label the configurations and do not forget to include you addressing scheme as part of your submission.

**Group Members**

Ideally this task should be completed in groups of two. Groups of three are allowed but not recommended.

Name: James Thomas Student ID: 9195071

Name: Alexander Collins Student ID: 9442540

Name: Liam Smith Student ID: 8452270

**Instructor signatures**

No instructor signatures are needed for this portfolio exercise. Submissions should include:

1. Group member composition
2. Addressing scheme
3. Router configurations (clearly labelled)
4. Evidence that end to end connectivity was achieved (i.e. ping and trace route screenshots)

1. The first subnet can be the one where all network bits are set to zero. This in effect makes the formula giving the number of subnets 2n-1. [↑](#footnote-ref-1)
2. Enhanced Internal Gateway Routing Protocol (EIGRP) is a CISCO proprietary vector routing protocol. It supports variable length subnet masking, uses multiple metrics to rate routes and only sends updates when changes are detected. This makes it a better choice than RIP, especially for bigger and more complex networks - EIGRP offers many advanced features, consumes less bandwidth and offers faster convergence (time taken before all routers get updated with the latest information). The main drawbacks of EIGRP are that it is a proprietary protocol not supported by all vendors and that it is more complex than RIP (requiring more effort to implement and more processing power to run). [↑](#footnote-ref-2)