**LAB 11**

**BASIC CONFIGURATION OF A Cisco ROUTER**

1. Open the file “Lab 11 – Basic configuration of a Cisco router.pka” in Packet Tracer. There, a router is used in the network. Hover with your mouse over the router. What is the router model?

Model 1941

1. This router has 2 Ethernet ports. Click on the Config tab to see what maximum speed can be achieved on these ports.

2 GigabitEthernet ports. Each port has a max speed of 1 Gbps

1. As admin IT guy/girl, we want to configure that router for the first time. Add your laptop (a ‘laptop-PT’ end device) to the workspace and make a connection with a console cable to the router. Open the Terminal on the laptop to access the CLI of the router.
2. You might see the question “Continue with configuration dialog”, in that case, answer with “no”. After all, you're supposed to run the initial configuration using commands yourself manually to understand better what is configured.



Using a command, request an overview of the current configuration of the router.

sh ru

Using a command, now request the contents of the boot configuration file. What do you see? Explain it!

The boot configuration file is not present. The user clicked no to the question we had before

1. Now, let’s make some initial configuration settings on the router (have a look at the previous lab about configuring a switch if you forgot some commands):

* Give the router your first name as its host name.

Conf t

Hostname chris

* Make sure that all passwords in the configuration file will be encrypted.

Service password-encryption

* Secure access to Privileged EXEC mode using the password (lower case): “icandothis”

Enable secret icandothis

* Previously, when configuring a switch, we set one generic password to access the CLI via the console (with the commands ***password*** and ***login***). Now, let’s take it one step further and set a password for specific users, rather than using one generic password for all.   
  We’ll make sure that only a user with login name “bumba” can access the console. First define the user and his password with the following command:

***username bumba secret studio101***

Note that behind the user’s name you can also opt for keyword *password* (instead of *secret*) but then you have no strong password protection! Thus don’t do it that way.

* Then make sure that the console is only made accessible to that user. That’s done in the specific configuration mode for the console line with the command:

***login local***

* Provide a suitable message of the day message: ”Forbidden access for unauthorized persons”

banner motd #Forbidden access for unauthorized persons#

* Save your current configuration to the router’s startup configuration (which is in NVRAM memory), using the default name.

Copy running-config startup-config

* Check that your current configuration was stored in the router’s NVRAM memory (thus check the startup-config)

Show startup-config

* Also copy a backup of the current configuration file to the flash memory of your router (using the default name) and then check if this has been achieved by using an IOS command to request the contents of your router’s flash memory.

Copy running-config flash

Sh flash

1. So far, most of the things you did were the same or similar as you’ve done on a switch. Let’s do some things which are specific for a router now. First, request the **route** **table** of your router via your console connection. What do you notice? Explain!

Sh ip route

It is empty. There are no interfaces configured yet.

1. Now, ping from PC0 to PC2. This will work because these PCs belong to the same network. All we had to do is connect these to the same switch.
2. Now, ping from PC0 to PC1. This will not work because these PCs do not belong to the same network .Explain why they don't belong to the same network.

You can see the 3th quad of the IP is different.

In order for PC0 (and PC2) to communicate with PC1, you have to configure the router. You do this by assigning IP addresses to the interfaces of the router using IOS commands (see lecture slides). A few things to consider:

* Hover with your mouse over the red triangle to know what interfaces are used for that connection
* The left interface of the router can have IP address 192.168.0.1
* The right interface of the router can have IP address 192.168.1.1
* Remember that router interfaces (as opposed to switch interfaces) are disabled by default. So you will have to explicitly enable them with a command!
* The left interface of the router is the default gateway for PC2 and PC0. The right interface of the router is the default gateway for PC1. Set the correct default gateway in each of the three PCs.

Int gi0/0

Ip addr 192.168.0.1 255.255.255.0

No shutdown

Exit

Int gi0/1

Ip addr 192.168.1.1 255.255.255.0

No shutdown

Exit

Then test your configuration by pinging from PC0 to PC1.This should be done after the configuration and should work.

Ping works now

1. Use some commands to verify your configuration (see slides):

* Request using the command **ip** information about all interfaces on your router
* request a summary of all IP information on all interfaces.

Sh ip interface

Sh ip interface brief

By hovering your mouse over the router, you can access this information in the GUI.

1. Now, request the router’s route table again. What kind of routes are included here?

Sh ip route: directly connected routes are visible

1. OK, so the network is configured properly. Just one more thing about the router: we want to get rid of the console cable and access the CLI via the network itself (thus via the IP address of the router).   
   Make sure you can login via telnet to your router from 2 different computers at the same time, so that you can access the router remotely from PC0, PC1 or PC2 (instead of having to use a direct console connection). When telnet’ing, don’t set one generic password, but you need to ask for a username and password (just like you did to secure access to the console line).

line vty 0 1

transport input telnet

login local

exit

1. Check if you can indeed start a Telnet session with the router from any of the PCs. You can use the ‘**telnet**’ command, with an IP address as parameter, in the command prompt of a PC. (Note that on an actual Windows 10 PC telnet is no longer installed by default).

At the CLI, you can check which sessions are active (console and/or vty) via the ***show users*** command.

Note that Telnet sessions are not safe and you will use SSH in practice. However, that’s quite specific in IOS to set up, which is why we don’t do that here (the focus is on understanding networks).

1. Now, let’s turn our attention towards our switches. We also like to manage these via the network, rather than via a console cable. Although a switch doesn’t use IP addresses for its operation, it is still possible to assign one to it for remote management.

Therefore, we will assign an IP address to a virtual interface of Switch0 (referred to as **SVI = Switch Virtual Interface**). Because the default SVI of a Cisco switch is called vlan1*,* you need to configure this with the commands below. Make a console connection, and type these commands at the CLI of Switch0.

***en***

***conf t***

***interface vlan 1***

***ip address 192.168.0.2 255.255.255.0***

***no shutdown***

Please note: As you can see, the *ip address* command used requires you to enter 2 parameters: first, an IP address (here:192.168.0.2), and second, a corresponding **subnet mask** (here: 255.255.255.0). Both parameters should be separated by a space.

The command *no shutdown* is mandatory and ensures that the virtual interface is enabled (because it is off by default).

1. Check if you can ping from PC0 to Switch0.

Works

Of course you also need to be able to ping backwards from Switch0 to PC0. Test this out too!

Works

Take a close look at the output of your IOS ping command. What is the difference with the Windows ping command (which is emulated on the PC0)?

The IOS ping sends 5 packets of 100 bytes, the PC0 ping sends 4 packets of 32 bytes

If you don't see this right away, you could do this in the Simulation Mode.

1. Now that Switch0 has an IP address, we can try to manage it remotely. Try to telnet from a command prompt at PC0 to your switch, similar as you did to Router0 in a previous question.

***telnet 192.168.0.2***

You will notice that your Telnet connection is automatically aborted by the switch.

The reason for this is that for security reasons a Cisco switch does not allow Telnet connections by default.

1. Sign in to Switch0 from PC0 and make sure that 3 simultaneous Telnet connections with the switch are allowed and secure these connections with a generic password “stayout”. Similar as you did for the router thus (but now without defining users). Use the following commands

***line vty 0 2***

***password stayout***

***login***

***transport input telnet***

The first line indicates that you can select the “virtual lines” vty0, vty1 and vty2 with a single command. The space in the command between the 0 and the means from 0 to 2.

The last line allows Telnet to be allowed on those “virtual lines”.

1. Try telnet’ing (or pinging) from PC1 to Switch0. You’ll notice that doesn’t work. Obvious, because they are on different networks and the switch doesn’t know what to use as default gateway to go beyond the network it is in. There define the default gateway on Swith0. In that network, the 192.168.0.1 IP address of Router0 is the gateway. Thus, on Switch0:

***ip default-gateway 192.168.0.1***

1. Now, verify that pinging/telnet’ing from PC1 to Switch0 works
2. Note that you could also protect your Switch0 console access and Privileged EXEC mode, as you did in a previous lab. And you could do all the same things on Switch1 (but then give it 192.168.1.2 as IP address). Feel free to do so as recap exercise, but it’s not obligatory for this lab. You’ve already done enough work 😊.
3. Save your pka file and upload it via Leho.