# Overview

This lab will use a Cisco 1841 router to connect 3 or 4 networks together. All the Cisco 1841 routers in the lab have 2 Ethernet ports built into the base router. Half the routers have a 1 port Ethernet WIC expansion card, the other half have a 2 port Ethernet WIC. This gives a capability of 3 or 4 networks per router.

This lab will be done with teams of 3 or 4, each on their own VM on separate workstations. The number per team depends on the ports available on the Cisco router. All members of the team should install, configure and test the serial terminal on their VM. After everyone has gotten their own serial port working, one team member can use their serial port to do the router configuration.

After the router is configured, the 3 or 4 VMs (each on a network the same as configured on the port in the router will connect to the router via Ethernet and see if all VMs can ping each other.

Note in your report any questions asked and anything noted with a \*\*.

# Part1 –Set up a serial terminal

## Enable your Debian VM to use the serial port on the workstation

**Important:** The serial port feature must be enabled ***before*** starting your VM

1. Start VirtualBox
2. Modify your VM to enable the serial port
   1. Select your VM but do NOT start it!!!!
   2. Click *Settings*
      1. Select *Serial Ports*
         1. Ensure tab *Port 1* is selected
         2. Ensure *Enable Serial Port* is checked
         3. Port Number: Select *COM1*
         4. Port Mode: Select *Host Device*
         5. Port/File Path: Enter */dev/ttyS0*
      2. Click OK
3. Click the *Start* arrow

## Install Minicom on your VM

* Use Synaptic or CLI to install the Minicom Serial Communication Program
  1. Synaptic: figure it out yourself
  2. CLI: open a root enabled terminal and use the command: apt-get install minicom
* Document which method you used and why. \*\*
* If you have trouble installing a package see the *Installing Packages* notes in the *Addenda*

## Start Minicom

1. Open a terminal as root (or with root authority)
2. The command *minicom* is normally used to start the terminal emulator. However, for the first use minicom will need to be initialized with the –s option:
   1. On a root enabled command line enter: minicom –s
      1. Do steps 3.a.i, 3.a.ii, and 3.c below to configure
      2. Restart minicom and continue

## Configure Minicom

**Hint:** The sequence <Ctrl>+A followed by the character Z will give a “help screen” at any time

1. Minicom will need to be configured to communicate with the serial port on the router.
2. Terminal emulation: VT102 (should be default)
3. Serial Port Setup: 9600,8,n,1 where 9600 is the speed in bits per second, 8 is the number of bits per character, n is the parity (none), and 1 is the number of stop bits.
4. <Ctrl>+A then O
   1. Select serial port setup
   2. Ensure the Serial Device is set to /dev/ttyS0 under option A
      * See *Note A* below if it needs changing
   3. Option E
   4. Set Speed: 9600, Parity: None, Data: 8, Stop bits: 1
      * Note: Option Q will set the 8N1 for you in one keystroke
5. Enable Text wrapping (<Ctrl>+A then W will toggle wrapping on and off)
   1. You should do this every time you start minicom
6. When exiting [configuration] select “Save setup as dfl” to save these settings so they will be used the next time you start minicom
7. Then exit….

**Note A**: if the serial device needs changing select option A, edit the line to read to be: /dev/ttyS0, next hit <enter>, then Save setup as dfl, and finally restart minicom.

**Note B:** Sometimes the VM will use /dev/ttyS1 instead of /dev/ttyS0

# Part 2 – Connect to and configure a router

## Connect the configuring VM to Cisco Router

* Get a Cisco Serial cable
  + Blue flat cable with DB9 on one end and RJ-45 on the other end
* Connect the DB9 Connector to the serial port on the Workstation
  + If there are two, be careful which one to choose.
* Connect the RJ45 connector to the Cisco 1841 router
  + The blue labeled console port on the back of the router
* Start minicom on your VM
* Turn the router on, you should see some dialog on the minicom terminal as the router initializes.
* If there is no dialog something is wrong, start troubleshooting…

Note: make sure all your workstation VMs can communicate to the router using the serial port. The rest of the lab can be done by a single VM.

## Reset the router to the factory default configuration

The Cisco route you get for the lab will be in an unknown configuration, with unknown IDs and passwords. To reset the router to factory presets go through the following procedure. To start the procedure you must restart the router by turning it off, then on again. While it is initializing you need to send the router a ***<break>*** signal (done in Minicom by doing a ***<Ctrl>+A*** then ***F***)

* Start a terminal session with the router via the Console port (telnet will not work for this). Physically reload the router, and from the console repeatedly hit the "Break" key, normally on the upper right hand side of the keyboard.
  + The break is done with a <Ctrl>+A then F in minicom
  + Do the break while the router is initializing
* Continue the “break” sequence until you see a prompt of “rommon 1 >”. You should now be in ROMMON mode.
  + Type "confreg 0x2142" without quotes and hit <enter>.
  + Now type "reset" without quotes and hit <enter>.
* Allow the router to restart.
  + Wait for the prompt of “Would you like to enter… …[yes/no]”.
  + Type "no" followed by <enter> to stop the startup wizard.
* When you see “Press RETURN to get started”, press <enter> and continue.
* When you see the prompt, type the following command sequence (each line has an <enter>):
  + enable  
    conf t  
    config-register 0x2102  
    exit
  + Hit <enter> if needed to get a prompt back, then enter:  
    write erase  
    <enter>  
    reload
  + Answer “yes” to the question and then confirm (<enter>)
* The router will now boot normally, with a factory default configuration, even if you did not know any of the password(s) securing the router.
  + You should see a bunch of ##### then some text as the router reboots.
  + Note: be sure your serial terminal is in line-wrap mode so you can see all the messages as the router restarts
* Wait for activity to stop. You will see the message “Press RETURN to get Started!” again followed by some text. Press <enter>
* You should see a “Router>” prompt.

## Restart the router using the configuration wizard

* Turn off and restart the router. While the router reboots send the <break> signal again (<Ctrl>+A then F)
* Type "confreg 0x2142" without quotes and hit <enter>.
* Now type "reset" without quotes and hit <enter>. Allow the router to restart.
  + This time when you see “Would you like to enter… …[yes/no]”.
  + Type "yes" followed by <enter> to start the startup wizard.
* Use the following **Example** as a guideline to configure your router:

### Example dialog of setting up the router.

Use the following example dialog as *guide* to set up your router. When setting the IP addresses for the ports on the router use 192.168.nn.1; where nn is the same as the number of the workstation that will connect to it. The example shows 192.168.10.1, 192.168.20.1, and 192.168.30.1. ***Change names and IP addresses as needed***.

--- System Configuration Dialog ---

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

At any point you may enter a question mark '?' for help.

Use ctrl-c to abort configuration dialog at any prompt.

Default settings are in square brackets '[]'.

Basic management setup configures only enough connectivity

for management of the system, extended setup will ask you

to configure each interface on the system

Would you like to enter basic management setup? [yes/no]: no

First, would you like to see the current interface summary? [yes]: no

Configuring global parameters:

Enter host name [Router]: ajkRouter1

The enable secret is a password used to protect access to

privileged EXEC and configuration modes. This password, after

entered, becomes encrypted in the configuration.

Enter enable secret: secretpass

The enable password is used when you do not specify an

enable secret password, with some older software versions, and

some boot images.

Enter enable password: enablepass

The virtual terminal password is used to protect

access to the router over a network interface.

Enter virtual terminal password: vtpass

Configure SNMP Network Management? [yes]: no

Configure IP? [yes]: yes

Configure RIP routing? [no]: no

Configure CLNS? [no]: no

Configure bridging? [no]: no

Configuring interface parameters:

Do you want to configure FastEthernet0/0 interface? [yes]: yes

Operate in full-duplex mode? [no]: yes

Configure IP on this interface? [yes]: yes

IP address for this interface: 192.168.10.1

Subnet mask for this interface [255.255.255.0] :

Class C network is 192.168.10.0, 24 subnet bits; mask is /24

Do you want to configure FastEthernet0/1 interface? [yes]: yes

Operate in full-duplex mode? [no]: yes

Configure IP on this interface? [yes]: yes

IP address for this interface: 192.168.20.1

Subnet mask for this interface [255.255.255.0] :

Class C network is 192.168.20.0, 24 subnet bits; mask is /24

Do you want to configure FastEthernet0/0/0 interface? [yes]: yes

Operate in full-duplex mode? [no]: yes

Configure IP on this interface? [yes]: yes

IP address for this interface: 192.168.30.1

Subnet mask for this interface [255.255.255.0] :

Class C network is 192.168.30.0, 24 subnet bits; mask is /24

Would you like to go through AutoSecure configuration? [yes]: no

AutoSecure dialog can be started later using "auto secure" CLI

The following configuration command script was created:

hostname ajkRouter1

enable secret 5 $1$AK3K$MvVl10j0X9qN4pBVcOuSl0

enable password enpass

line vty 0 4

password vtpass

no snmp-server

!

ip routing

no clns routing

no bridge 1

!

interface FastEthernet0/0

full-duplex

ip address 192.168.10.1 255.255.255.0

no mop enabled

!

interface FastEthernet0/1

full-duplex

ip address 192.168.20.1 255.255.255.0

no mop enabled

!

interface FastEthernet0/0/0

full-duplex

ip address 192.168.30.1 255.255.255.0

no mop enabled

dialer-list 1 protocol ip permit

!

end

[0] Go to the IOS command prompt without saving this config.

[1] Return back to the setup without saving this config.

[2] Save this configuration to nvram and exit.

Enter your selection [2]: 2

Building configuration...

Use the enabled mode 'configure' command to modify this configuration.

Press RETURN to get started!

…

After done with the above, type the following command sequence (each line has an <enter>):

* + enable  
    conf t  
    config-register 0x2102  
    exit

## Manually setting up router (optional to using the above configuration wizard)

This section is a brief overview for manual setup and is for informational purposes only. It is an alternative to the above procedure. It is the preferred way when extensive custom configuration is desired. It is more flexible but requires more knowledge and skill to set the router up.

Enter privileged mode and enter the command “setup”, then hit <enter>

* The first pass will be to configure the basics to just get the router going (e.g. configure only one port
  + Just set up one of the ports
* The second setup will be able to configure the whole router.
  + Redo setup and configure the rest of the ports

## Modes

The router has modes to do common command (unprivileged) and “dangerous” commands that can configure the router (privileged). The router’s prompt will show you which mode you are in. The prompt is the router’s name followed by a special character. The special character shows the mode. E.g.: myrouter> is the prompt for unprivileged mode and myrouter# is the prompt for privileged mode.

* enable command enters privileged mode
* disable command leaves privileged mode

**Note:** you can use <Ctrl>-Z to stop commands or return to parent modes.

## Basic Commands

Try some basic commands in privileged and unprivileged modes and notice the results:

* show interfaces
* show protocols
* show ip route
* show ip arp

***Note 1: show several lines from each of the above commands in you lab report and comment on them.\*\****

The “?” command will show what commands are available in your current mode. Try the ? in both unprivileged and privileged modes and note what commands are available in each mode.

Next, try going into configuration sub-mode:

* Ensure you are in privileged mode (with the # in the prompt)
* Type config and hit <enter>
  + Note that hitting <enter> selects the default mode of terminal
* Your prompt will change to show the mode you are in e.g.: ajkrouter0(config)#
* Change the name of your router: hostname NewName
* Note the prompt has changed to show the new name
  + You may change the name back if you like
* Set a DNS server for the router
  + Ensure you are in privileged mode
  + Enter config sub-mode by typing:
    - ip name-server 172.16.1.245
    - <enter>
    - <Ctrl>-z
  + The <Ctrl>-z forces the execution of the command typed before it and exits the config sub-mode. The above ip command has set the hades.lab DNS server as for this router

***Note 2: in your lab report document the name of your router.\*\****

# Part 3 – Use the configured router

## Test the router working

At this point the router should be configured and ready for routing.

Connect three or four (depending on the number of ports on the router) workstation with a Debian or CentOS VM to a port on the router. Make sure the network address of the VM matches the network address of the address of the port on the router.

The router should do some auto-configuration of the routing tables and allow the ports to communicate. Test it by trying to ping among the VMs.

When you try the pings from VM to VM nothing will happen. The router should be properly configured and ready to route. The problem is with the configuration of the network interface on the VMs. There is one small but critical item missing. Think about what might be missing from the VMs network configuration and fix it. After a reasonable effort you may ask the TA for assistance on the missing item.

***Note 3: document in your lab report that the VMs can ping each other!\*\****

After some successful communications across the VMs check the contents of the router’s routing table (show ip route).

***Optional:*** After the pings are working on your router, hook 2 routers together by disconnecting 1 workstation from each of the routers and connecting the routers together with a cable. What changes need to be made to get the routers to communicate?

Hints:

* use config to change the router ports
* use interface FastEthernet0/1 to select the port
* use ip 192.168.nn.xx 255.255.255.0 to change the address
* use <Ctrl>-z to effect the change

Check to see it the pings work across the routers. Check contents of the routing tables on the routers now

***Bonus points: document this step working***

# Part 4 – Save the configuration

## Save the current setup

The above changes have only set the current configuration in volatile memory. If the router is shut off and restarted all changes are lost. To have the current data valid next startup the current configuration must be copied to the startup configuration.

1. Ensure you are in privileged mode
2. Check the current configuration: show running-config
3. Save the current configuration: copy running-config startup-config

## Test the setup

Turn the router off, and then back on again. Check that the router has the configuration from when you last saved it.

***Note 4: Comment in the lab report if the save worked or not.***

# Deliverables

* Lab write-up with introduction and summary paragraphs.
* Copy of lab with comments
  + Comment as appropriate on the steps involved on the lab.
* Be sure to document ***Note 1*** and ***Note 2*** in the lab report!
* **Bonus:** document the step where 2 routers are used to hook 4 or more networks together

# Addenda

## Installing Packages

* To install a package the VM must be in the same network as hades.lab (172.16.0.0) so it can access the mirror
* Easiest way to ensure the VM is in the same network is to use DHCP

## Minicom notes

* Make sure Minicom has word-wrap on so you can see all the messages
* <Ctrl>+A then Z gets you to the help screen
  + From there you may access functions by typing the appropriate command (letter)
* The commands may also be directly reached by <Ctrl>+A then the command letter
* <Ctrl>+F does a <break>
* <Ctrl>+W toggles line-wrap

## Serial port notes

* The serial port uses /dev/ttyS0 or /dev/ttyS1
* To use ttyS0 or ttyS1 the user must be a member of the dialout group
* If the user is not a member of the dialout group change the permissions to 777 (must be root)
  + This will work until the user logs out, the next time the user logs in the permissions will be changed back to 660

# Other Comments/Notes

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_