Cisco Config Elements

Notes:

* Commands preceded with "$" imply that you should execute the command as a general user - not as root.
* Commands preceded with "#" imply that you should be working as root.
* Commands with more specific command lines (e.g. "rtr>" or "mysql>") imply that you are executing commands on remote equipment, or within another program.
* If a command line ends with "\" this indicates that the command continues on the next line and you should treat this as a single line.

Exercises Part I

0. Work in a group

For this exercise you need to work in groups. Assign one person to type on the keyboard. There should be 4 people in group. For instance, members of Group 1 are those on pc1-pc4, Group 2 use pc5-pc8, Group 3 use pc9-12, etc…

If you are unsure of what group you are in refer to the Network Diagram on the classroom wiki by going to http://noc.ws.nsrc.org/ and clicking on the Network Diagram link.

1. Connect to your router

Log in to your vm/pc image and install Telnet:

```
$ sudo apt-get install telnet
```

If it is already installed that is fine.

Connect to router in your group. If you are not sure remember to review the classroom network diagram. Click on the Network Diagram link on the main NOC web page:

```
http://noc.ws.nsrc.org/
```

Now connect to your router:

```
$ telnet 10.10.N.254

username: cisco
password: cisco
```

Display information about your router

```
routerN>enable
    Password:               (default pw "cisco")
RouterN#show run            (space to continue)
RouterN#show int FastEthernet0/0
RouterN#show ?              (lists all options)
RouterN#exit                (log off router)
```

2. Configure your router to only use SSH

These steps will do the following:

    * Create an ssh key for your router
    * Create an encrypted password for the user cisco
    * Encrypt the enable password (cisco)
    * Turn off telnet (unencrypted) access to your router
    * Turn on SSH (version 2) access to your router

You need to work in groups of 4. Get together with the members of your router group and assign one person to enter commands. To start connect to one of the PCs in use by your group. From that PC image telnet to your router:

```
$ telnet rtrN.ws.nsrc.org   (or "telnet 10.10.N.254")

username: cisco
password: cisco

rtrN> enable                        (en)
password: cisco
rtrN# configure terminal            (conf t)
rtrN(config)# aaa new-model
rtrN(config)# ip domain-name ws.nsrc.org
rtrN(config)# crypto key generate rsa

How many bits in the modulus [512]: 2048
```

Wait for the key to generate. You can now specify passwords and they will be encrypted. First let's remove our cisco user temporarily, then we'll recreate the user:

```
rtrN(config)# no username cisco
rtrN(config)# username cisco secret 0 <CLASS PASSWORD>
```

Now the cisco user's password (of <CLASS PASSWORD>) is encrypted. Next let's encrypt the enable password as well:

```
rtrN(config)# enable secret 0 <CLASS PASSWORD>
```

Now we'll tell our router to only allow SSH connections on the 5 defined consoles (vty 0 through 4):

```
rtrN(config)# line vty 0 4
rtrN(config-line)# transport input ssh
rtrN(config-line)# exit
```

This drops us out of the "line" configuration mode and back in to the general configuration mode. Now we'll tell the router to log SSH-related events and to only allow SSH version 2 connections:

```
rtrN(config)# ip ssh logging events
rtrN(config)# ip ssh version 2
```

Now exit from configuration mode:

```
rtrN(config)# exit
```

And, write these changes to the routers permament configuration:

```
rtrN# write memory              (wr mem)
```

Ok. That's it. You can no longer use telnet to connect to your router. You must connect using SSH with the user "cisco" and password <CLASS PASSWORD>. The enable password is, also, "cisco" - Naturally in a real-world situation you would use much more secure passwords.

Let's exit from the router interface and reconnect using SSH:

```
    rtrN# exit
```

First, try connection again with telnet:

```
    $ telnet rtrN.ws.nsrc.org
```

What happens? You should see something like:

```
    Trying 10.10.N.254...
    telnet: Unable to connect to remote host: Connection refused
```

Now try connecting with SSH:

```
    $ ssh cisco@rtrN.ws.nsrc.org
```

You should see something looks similar to this:

```
    The authenticity of host 'rtr2.ws.nsrc.org (10.10.2.254)' can't be
        established. RSA key fingerprint is 93:4c:eb:ad:5c:4a:a6:3e:8b:9e:
    4f:e4:e2:eb:e4:7f. Are you sure you want to continue connecting
    (yes/no)?
```

Enter in "yes" and press ENTER to continue...

Now you'll see the follwoing:

```
    Password: <CLASSS PASSWORD>
    rtrN>
```

Type "enable" to allow us to execute privileged commands:

```
    rtrN> enable
    Password: <CLASS PASSWORD>
    rtrN#
```

Now let's view the current router configuration:

```
    rtrN# show running                  (sh run)
```

Press the space bar to continue. Note some of the entries like:

```
    enable secret 5 $1$p4/E$PnPk6VaF8QoZMhJx56oXs.
    .
    .
    .
    username cisco secret 5 $1$uNg1$M1yscHhYs..upaPP4p8gX1
    .
    .
    .
    line vty 0 4
     exec-timeout 0 0
     transport input ssh
```

You can see that both the enable password and the password for the user cisco have been encrypted. This is a good thing.

Now you should exit the router interface to complete this exercise:

```
    rtrN# exit
```

NOTES

1.) If you are locked out of your router after this exercise let your instructor know and they can reset your router's configuration back to its

    original state.
2.) Please only do this exercise once. If multiple people do this exercise
    it's very likely that access to the router will be broken.

3. Configure your router to send logging messages to your PCs

The routers are able to send syslog messages to multiple destinations,

Configure sending of syslog

Configure your virtual router to send syslog messages to every server
in your group.

Everyone in your group should log into your group's router and do the
following:

```
$ ssh cisco@rtrX
rtrX> enable
rtrX# config terminal

rtrX(config)# logging 10.10.X.Y
```

... where X.Y is the IP of your PC (group + number).

```
rtrX(config)# logging facility local0
rtrX(config)# logging userinfo
rtrX(config)# exit
rtrX# write memory
```

Now run `show logging` to see the summary of the log configuration.

```
rtrX# show logging
```

The other participants in your group will be doing the same thing,
so you should not be surprised if you see other destinations as well
in the output of "show logging"

Logout from the router (exit):

```
rtrX# exit
```

That's it. The router should now be sending UDP SYSLOG packets to your PC
on port 514.

To verify this log in on your PC and do the following:

```
$ sudo bash
# tcpdump -s0 -n -i eth0 udp port 514
```

Then have one person in your group log back in on the router and do the
following:

```
$ ssh cisco@rtrX
rtrX> enable
rtrX# config terminal
rtrX(config)# exit
rtrX> exit
```

You should see some output on your PC's screen from `tcpdump`. It should look
something like:

```
02:20:24.942289 10.10.1.254.63515 > 10.10.1.1.514: SYSLOG local0.notice, length: 102
02:20:24.944376 10.10.1.254.53407 > 10.10.1.1.514: SYSLOG local0.notice, length: 102
```

When you have seen this, hit Ctrl-C to exit tcpdump.

(Aside: tcpdump would also show you the *content* of the syslog messages if
you add `-v` to the command line)

Now you could configure logging software to receive messages from your routers and act accordingly.