Network Management & Monitoring

Using RANCID
============

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-GW>" or "mysql>")
  imply that you are executing commands on remote equipment, or within
  another program.

Exercises
----

1. Connect to your PC using ssh

2. Become root, and install the Subversion Version Control System:

        In addition to Subversion we will specify to install telnet and the mutt
        email client. Both these package may already be installed from prior
        exercises. If so, don't worry - the apt-get command will not reinstall
        them.

        $ sudo bash
          # apt-get install subversion telnet mutt

3. Install Rancid itself

          # apt-get install rancid

        - It will prompt with a warning - Select <OK> and press ENTER to continue.
        - It will give you another warning about making a backup copy of your
          rancid data. We have no data, so select <YES> and press ENTER to continue.

4. Add an alias for the rancid user in /etc/aliases file

          # editor /etc/aliases

          rancid-all: sysadm
          rancid-admin-all: sysadm

    Save the file, then run:

          # newaliases

5. Edit /etc/rancid/rancid.conf

          # editor /etc/rancid/rancid.conf

Find this line in rancid.conf:

        #LIST_OF_GROUPS="sl joebobisp"

And, underneath it add the following line:

        LIST_OF_GROUPS="all"

(with no '#' at the front of line, and aligned to the left)

We want to use Subversion for our Version Control System, and not CVS, so find the line with the parameter RCSSYS:

        RCSSYS=cvs; export RCSSYS

And, change it to:

      RCSSYS=svn; export RCSSYS

and the line with CVSROOT:

        CVSROOT=$BASEDIR/CVS; export CVSROOT

And, change it to:

      CVSROOT=$BASEDIR/svn; export CVSROOT

Note the lowercase "svn". Now exit and save the file.

6. Change to the rancid user

CRITICAL!

Pay very close attention to what userid you are using during the rest of these exercises.
If you are not sure simply type "id" on the command line at any time.

    From a root prompt ("#"), switch identity to become the 'rancid' user:

        # su -s /bin/bash rancid

    Check that you ARE the rancid user:

        $ id

    You should see something similar (numbers may be different):

        uid=104(rancid) gid=109(rancid) groups=109(rancid)

    ***** IF YOU ARE NOT USER RANCID NOW, do NOT continue ******

7. Create /var/lib/rancid/.cloginrc

        $ editor /var/lib/rancid/.cloginrc

    Add the following two lines to the file:

        add user *.ws.nsrc.org cisco
        add password *.ws.nsrc.org nsrc+ws nsrc+ws

    (The first 'cisco' is the username, the first and second 'nsrc+ws' are the
    password and enable password used to login to your router. The star in the name
    means that it will try to use this username and password for all routers
    whose names end .ws.nsrc.org)

    Exit and save the file.

    Now protect this file so that it cannot be read by other users:

        $ chmod 600 /var/lib/rancid/.cloginrc

8. Test login to the router of your group

     Login to your router with clogin. You might have to type yes to the first
warning, but should not need to enter a password, this should be automatic.

        $ /var/lib/rancid/bin/clogin rtrX.ws.nsrc.org

        (replace X with your group number. So, group 1 is rtr1.ws.nsrc.org)

     You should get something like:

        spawn ssh -c 3des -x -l cisco rtr2.ws.nsrc.org
        The authenticity of host 'rtr2.ws.nsrc.org (10.10.2.254)' can't be
        established.
        RSA key fingerprint is 73:f3:f0:e8:78:ab:49:1c:d9:5d:49:01:a4:e1:2a:83.
        Are you sure you want to continue connecting (yes/no)?
        Host rtr1.ws.nsrc.org added to the list of known hosts.
        yes
        Warning: Permanently added 'rtr1.ws.nsrc.org' (RSA) to the list of known
        hosts.
        Password:

        rtr1>enable
        Password:
        rtr1#

     Exit the from the router login:

        rtr2#exit

9. Initialize the SVN repository for rancid:

        Make sure you are the rancid user before doing this:

             $ id

        If you do not see something like
              "uid=108(rancid) gid=113(rancid) groups=113(rancid)"
        then DO NOT CONTINUE until you have become the rancid user. See exercise 6 for
        details.

        Now initialize the Version Control repository (it will use Subversion):

         $ /usr/lib/rancid/bin/rancid-cvs

     You should see something similar to this:

        Committed revision 1.
        Checked out revision 1.
        At revision 1.
        A   configs
        Adding   configs

        Committed revision 2.
        A   router.db
        Adding   router.db
        Transmitting file data .
        Committed revision 3.

******** Do this ONLY if you have problems *******

    If this does not work, then either you are missing the subversion package, or
    something was not properly configured during the previous steps. You should
    verify that subversion is installed and then before running the rancid-cvs
    command again do the following:

```
$ exit
# apt-get install subversion
# su -s /bin/bash rancid
$ cd /var/lib/rancid
$ rm -rf all
$ rm -rf svn
```

   Now try running the rancid-cvs command again:

```
$ /usr/lib/rancid/bin/rancid-cvs
```

****

10. Create the router.db file

```
$ editor /var/lib/rancid/all/router.db
```

    Add this line:

```
rtrX.ws.nsrc.org:cisco:up
```

    (remember to replace X as appropriate)

    Exit and save the file.

11. Let's run rancid!

```
$ /usr/lib/rancid/bin/rancid-run
```

    This will take a few moments so be patient.

    Run it again, since the first time it might not commit correctly:

```
$ /usr/lib/rancid/bin/rancid-run
```

12. Check the rancid log files:

```
$ cd /var/lib/rancid/logs
$ ls -l
```

    ... View the contents of the file(s):

```
$ less all.*
```

    NOTE! Using "less" - to see the next file press ":n". To see the Previous
    file press ":p". To exit from less press "q".

13. Look at the configs

```
$ cd /var/lib/rancid/all/configs
$ less rtrX.ws.nsrc.org
```

    Where you should replace "X" with your group number.

    If all went well, you can see the config of the router.

14. Let's change an interface Description on the router

        $ /usr/lib/rancid/bin/clogin rtrX.ws.nsrc.org

    Where you should replace "X" with your group number.

    At the "rtrX#" prompt, enter the command:

        rtrX# conf term

    You should see:

        Enter configuration commands, one per line. End with CNTL/Z.
        rtrX(config)#

    Enter:

        rtrX(config)# interface LoopbackXX   (replace XX with your PC no)

    You should get this prompt:

        rtrX(config-if)#

    Enter:

        rtr2(config-if)# description <put your name here>
        rtr2(config-if)# end

    You should now have this prompt:

        rtrX#

    To save the config to memory:

        rtrX# write memory

    You should see:

        Building configuration...
        [OK]

    To exit type:

        rtrX# exit

      Now you should be back at your rancid user prompt on your system:

15. Let's run rancid again:

        $ /usr/lib/rancid/bin/rancid-run

    Look at the rancid logs

        $ ls /var/lib/rancid/logs/

    You should see the latest rancid execution as a new log file with the date
    and time in the name.

16. Let's see the differences

        $ cd /var/lib/rancid/all/configs
        $ ls -l

You should see the router config file for your group:

```
$ svn log rtrX.ws.nsrc.org
```

(where X is the number of your router)

Notice the revisions. Let's view the difference between two versions:

```
$ svn diff -r 5:7 rtrX.ws.nsrc.org | less
```

... can you find your changes?

Notice that svn is the Subversion Version Control system command line
tool for viewing Subversion repositories of information. If you type:

```
$ ls -lah
```

You will see a hidden directory called ".svn" - this actually contains all the
information about the changes between router configurations from each time you
run rancid using /usr/lib/rancid/bin/rancid-run.

Whatever you do, don't edit or touch the .svn directory by hand!

17. Check your mail

   Now we will exit from the rancid user shell and the root user shell to go
   back to being the "sysadm" user. Then we'll use the "mutt" email client to
   see if rancid has been sending emails to the sysadm user.

```
$ exit                               (takes your from rancid to root user)
# exit                               (take you from root to sysadm user)
$ id
... check that you are now the 'sysadm' user again;

... if not, log out and in again as sysadm to your virtual host

$ mutt
```

(When asked to create the Mail directory, say Yes)

If everything goes as planned, you should be able to read the mails
sent by Rancid. You can select an email sent by "rancid@pcX.ws.nsrc.org"
and see what it looks like.

Notice that it is your router description and any differences from the last
time it was obtained using the rancid-run command.

Now exit from mutt.

(use 'q' return to mail index, and 'q' again to quit mutt)

18. Let's make rancid run automatically every 30 minutes from using cron

   cron is a system available in Linux to automate the running of jobs. First we
   need to become the root user again:

```
$ sudo bash
```

Now we will create a new job to run for the rancid user:

```
# crontab -e -u rancid
```

It will ask you for your favorite editor. Select whichever editor you have been
using in class.

Add this line at the bottom of the file (COPY and PASTE):

    */30 * * * * /usr/lib/rancid/bin/rancid-run

... then save and quit from the file.

That's it. The command "rancid-run" will execute automatically from now on every
30 minutes all the time (every day, week and month).

19. Now add all the other routers

    Note the addresses for the routers

        rtrX.ws.nsrc.org where X goes from 1 to 9

    If you have less routers in your class, then only include the actual, available
    routers.

    Become the rancid user and update the router.db file:

        # su -s /bin/bash rancid
        $ editor /var/lib/rancid/all/router.db

    Add the other classroom routers to the file. You should end up with
    something like (COPY and PASTE):

        rtr1.ws.nsrc.org:cisco:up
        rtr2.ws.nsrc.org:cisco:up
        rtr3.ws.nsrc.org:cisco:up
        rtr4.ws.nsrc.org:cisco:up
        rtr5.ws.nsrc.org:cisco:up
        rtr6.ws.nsrc.org:cisco:up

    (Note that "cisco" means this is Cisco equipment -- it tells Rancid
    that we are expecting to talk to a Cisco device here. You can also
    talk to Juniper, HP, ...).

    Be sure the entries are aligned to the left of the file.

20. Run rancid again:

        $ /usr/lib/rancid/bin/rancid-run

    This should take a minute or more now, be patient.

21. Check out the logs:

        $ cd /var/lib/rancid/logs
        $ ls -l

    ... Pick the latest file and view it

        $ less all.YYYYMMDD.HHMMSS

    This should be the last file listed in the output from "ls -l"

    You should notice a bunch of statements indicating that routers have been
    added to the Subversion version control repository, and much more.

22. Look at the configs

        $ cd /var/lib/rancid/all/configs
        $ more *.ws.nsrc.org

     Press the SPACE bar to continue through each file. Or, you could do:

            $ less *.ws.nsrc.org

     And press the SPACE bar to scroll through each file and then press ":n" to
     view the next file. Remember, in both cases you can press "q" to quit at
     any time.

     If all went well, you can see the configs of ALL routers

23. Run RANCID again just in case someone changed some configuration on the router

        $ /usr/lib/rancid/bin/rancid-run

     This could take a few moments, so be patient....

23. Play with clogin:

        $ /usr/lib/rancid/bin/clogin -c "show clock" rtrX.ws.nsrc.org

     Where "X" is the number of your group.

     What do you notice ?

     Even better, we can show the power of using a simple script to make changes
     to multiple devices quickly:

        $ editor /tmp/newuser

     ... in this file, add the following commands (COPY and PASTE):

configure terminal
username NewUser secret 0 NewPassword
exit
write

     Save the file, exit, and run the following commands from the command line:

            $ for r in 1 2 3 4

            Your prompt will now change to be ">". Continue by typing:

            > do
            > /var/lib/rancid/bin/clogin -x /tmp/newuser rtr$r.ws.nsrc.org
            > done

     Now your prompt will go back to "$" and rancid clogin command will run and
     execute the commands you just typed above on routers rtr1, rtr2, rtr3 and
     rtr4. This is simple shell scripting in Linux, but it's very powerful.

     Q. How would you verify that this has executed correctly ? Hint: "show run | inc"

     A. Connect to rtr1, rtr2, rtr3 and rtr4. Type "enable" and then type
        "show run | inc username" to verify that the NewUser username now exists.
        Type exit to leave each router. Naturally you could automate this like we just
        did above.

24. Add the RANCID SVN (Subversion) repository in to WebSVN

    If you are still logged in as user rancid, get back to root. Remember you can
    type "id" to check what userid you are.

        $ exit
        #

    Install WebSVN:

        # apt-get install websvn
                * Select <Yes> to the question if you want to configure WebSVN now and
                  press ENTER
                * Select <Ok> for the next question about supporting various web servers
                  and press ENTER
                * When asked for the "svn parent repositories" change the path to be:

                  /var/lib/rancid/svn

                  Select <Ok> and press ENTER. Do the same when asked about "svn
                  repositories" on the next screen. That is, use the path:

                  /var/lib/rancid/svn

                  and not what is shown by default. Select <Ok> and press ENTER.
                * Select <Ok> for the next screen talking about permissions and press
                  ENTER.

25. Fix permissions. The web server must be able to read the SVN (Subversion) folder

        # chgrp -R www-data /var/lib/rancid/svn
        # chmod g+w -R /var/lib/rancid/svn

26. Browse the rancid files from your Web browser!

        http://pcX.ws.nsrc.org/websvn

    Browse the files under the 'all/configs' directory.
    You can see all your router configuration files here.

27. Review revisions

    WebSVN lets you see easily the changes between versions.

        * Browse to http://pcXXX.ws.nsrc.org/websvn again, go to all, configs.
        * Click on your router file (rtrX.ws.nsrc.org) name. You will get a new screen
        * Click "Compare with Previous" at the top of the screen.
        * You should now see the latest changes highlighted.

    Click on "REPOS 1" to back to the main WebSVN page:

        * Click on "all/" under "Path"
        * Click on "configs/"
        * Select two of the routers that are next to each other. I.E. rtr1 and rtr2,
          rtr3 and rtr4.
        * Click on Compare Paths

     This will show you the differences between two separate router configurations.
    WebSVN is a convenient way to quickly see differences via a GUI between mulitple
    configuration files. Note, this is a potential security hole so you should limit
    access to the URL http://host/websvn using passwords (and SSL) or appropriate
    access control lists.   +----   Rev. 12 Oct 2012