

WORKSHOP



CONFIGURATION MANAGEMENT

RANCID Tool Keeps Config Files Clean

By Bruce Boardman

There's nothing rotten about RANCID, or Really Awesome New Cisco config Differ, a Linux and Unix freeware tool for configuration management. Able to handle devices from a variety of router and switch vendors, it's easy to use once you get the hang of it.

RANCID automatically downloads configurations from your network devices and compares them with previously gathered data so you can monitor changes, problems and software updates. Once the tool compares the latest configuration data from your switches and routers with the data stored in your config database, it shoots you an e-mail message noting any changes. It also can send config files to a router or switch.

True, RANCID doesn't do Windows. But while you can find plenty of vendor-supported, Windows-based commercial configuration products that are more user-friendly, such as those from Opsware and AlterPoint, they don't come cheap. And besides its can't-beat-it price, RANCID can scale with large networks—several ISPs and network service providers, including America Online and Global Crossing, use it.

Here's how to get started with RANCID: First, if you're not already running a version of Linux or Unix, go to www.linux.org or download Red Hat Linux at www.redhat.com, as we did in our Syracuse Real-World Labs®. We monitored a switch and router with RANCID 2.3.1 on Red Hat 9.0. The RANCID package uses Perl 5.0 and later versions as well as Tcl Expect scripts. If you're unfamiliar with Perl and Expect, don't worry—RANCID will work anyway.

Perl and Tcl Expect are usually part of Linux distributions, but there are plenty of free packages available if you don't already have them. ActiveState (www.activestate.com) is a good site to get free versions of both Perl and Tcl, as well as more advanced paid versions. You need these base Perl and Tcl Expect environments to run the first RANCID install task, a configuration

script. John Ousterhouse, who created Tcl, runs an Expect site at expect.nist.gov.

RANCID, which is a compressed tar file, can be downloaded from www.shrubbery.net/rancid/. If you're living on the character interface, `tar -vxf filename` should decompress the distribution into a file directory under your current user name. If you run Red Hat or another Linux distribution, the GUI file manager will offer an "extract" option. By default, the directory name will show the RANCID version you've just extracted—in our case, *rancid-2.3.1*.

We extracted the RANCID file into a *rancid* subdirectory in our home directory so we'd have permission to run and write files (\$HOME is the variable defined for your home directory if you're using the default Red Hat BASH shell). You also could allow RANCID files to be extracted to a shared area with access permissions, but we kept it simple.

Under the \$HOME/*rancid* directory is a *rancid-2.3.1* directory containing all the extracted files. The README in \$HOME/*rancid/rancid-2.3.1* provides step-by-step installation and configuration directions. At the command prompt, type:

less README

This will display the steps to install RANCID.

Configure It

Run a configuration that sets up the files needed for your OS environment. The executable script for this is "configure," which you can find in the \$HOME/*rancid/rancid-2.3.1* directory.

Before executing the configure-script notice, the

```
Index: configs/switch.nwc.syr.edu
-----
retrieving revision 1.5
diff -u -4 -r1.5 switch.nwc.syr.edu
@@ -35,16 +35,16 @@
 (Flash: 2 -rwx 1645810 Jul 18 2000 01:26:29
c2900XL-c3h2s-mz-120.5.2-XU.bin
 (Flash: 3 -rwx 105970 Jul 18 2000 01:26:29
c2900XL-diag-mz-120.5.2-XU
 (Flash: 4 drwx 6784 Jul 18 2000 01:26:30 html
 (Flash: 112 -rwx 287 Jan 01 1970 00:00:26 env vars
- (Flash: 113 -rwx 4424 Mar 30 1993 05:14:12 config.text
 (Flash: 114 -rwx 600 Mar 02 1993 23:28:38 nwc
+ (Flash: 115 -rwx 4423 Mar 30 1993 06:01:46 config.text
 (Flash: 3612672 bytes total (828928 bytes free)
)
 (Flash: : Directory of nvram:/
- (Flash: : 1 -rw- 4424 <no date> startup-config
+ (Flash: : 1 -rw- 4423 <no date> startup-config
 (Flash: : 2 ---- 25 <no date> private-config
 (Flash: : 32768 bytes total (28344 bytes free)
```

When there's a configuration change in your network devices, RANCID automatically e-mails you with the "diffed" config files.

WORKSHOP

CONFIGURATION MANAGEMENT

README points to a couple of directory parameters that can be modified. By default, the “configure” script will set up files to be installed in the directory `/usr/local/rancid` for executable, configuration and help files. It also by default sets up log and configuration files for the managed routers and switches in `/home/rancid/var`. For this installation, we placed the router and switch log and configuration files in our `$HOME/rancid` directory, where you can install the app without it being “root.”

We ran “configure” by specifying two directory parameters—PREFIX and localstatedir. PREFIX directs where “configure” should place the application files,

By default, the ‘configure’ script will set up files to be installed in `/usr/local/rancid`.

and localstatedir sets up the locations for switch and router configuration files as well as RANCID execution logs. We specified the following command in the `$HOME/rancid/rancid-2.3.1` directory:

```
configure -prefix=$HOME/rancid —  
localstatedir=$HOME/rancid/var
```

We didn’t include the “.” in front of the “configure” command above. The “.” tells the shell, BASH, to look in the current directory for the configure script. This may seem odd if you’re a Windows user, because the current directory is always searched on the Windows command line. But Linux and Unix shells by default do not include the current directory. We added it to our path so we wouldn’t have to specify “.” for every command.

You’ll get a list of checked system requirements and environmental data from “configure.” If there’s a problem, “configure” will stop processing and display an error message, and no files would be moved to the directories.

Next, move files by executing the “make” command. The “make install” command requires no parameters, as it will use those set up in the previous “configure” step. I ran the following command in the `$HOME/rancid/rancid-2.3.1` directory:

```
make install
```

This compiles run-time binaries, creates the directories for RANCID and moves the RANCID files to the specified directories. You’ll notice that the `$HOME/rancid/var` directory does not yet have any files, but it will get populated when RANCID runs in real time.

Now the basic files that log on to routers and switches can be executed. Here’s how they work manually: There are two files for running a RANCID connection to a router or switch—Cisco, Extreme, Juniper ERX

/ E, Procket Networks and Redback devices are supported using “clogin” and “.cloginrc.” A similar pair of Expect script and configuration files exists for Bay Networks, ADC Kentrox EZ-T3 mux, Foundry Networks and HP ProCurve switches; Hitachi routers; and Juniper, NetScreen, NetScaler, Riverston and Lucent TNT network devices. Juniper, for example, uses the files “jlogin” and “jloginrc” for its routers.

You can pass parameters on the command line to “clogin,” so you don’t need to configure parameters in “.cloginrc.” To attach to a Cisco switch in the lab, for instance, we ran the command:

```
clogin -v vtypassword -e enablepassword switchinlab.nwc.com
```

After “clogin” was completed, we were left at the enable prompt of the target switch:

```
clogin -v vtypassword -e enablepassword -c 'sh ver'  
switch.nwc.com
```

This made “clogin” log on to the switch in the lab and run the “show version” command, which then displayed the results in our telnet session. Any native switch or router configuration command can be specified. “Clogin” supports multiple serialized commands, each separated with a semicolon:

```
clogin -v vtypassword -e enablepassword -c 'sh ver;sh vlan'  
switch.nwc.com
```

Home Sweet Home Directory

The real power of RANCID is how it automatically tracks and notifies you of any configuration changes. To take advantage of that, you must set up e-mail notification recipients.

RANCID is designed to support multiple groups of network engineers working on different groups of switches and routers. In our labs, for instance, we have a set of devices in Syracuse, N.Y., Green Bay, Wis., and Gainesville, Fla. It makes sense for the technical editors in each location to oversee the devices located in their respective labs.

But to show how to get automation running, we defined a single group. RANCID’s install README directs the modification of the configuration file “rancid.conf.” This file will be in your `$HOME/rancid/etc` directory, where you can set variables such as work directories, job time-outs and groups for switches and routers. We modified the group setting and added only a single group—“nwc”:

```
List_OF_GROUPS=$LIST_OF_GROUPS nwc
```

The configuration file “.cloginrc” passes various parameters to “clogin.” Userid, password and unique prompts, for example, can be stored in the “.cloginrc” file. The README shows how to configure “.cloginrc.”

This file in our download came as a file named “cloginrc.sample.” But we had to rename it to “.cloginrc” in order to use it. We then placed in the user’s home directory (*\$HOME* if using BASH on Red Hat). This renaming and moving process sets up the operating system so that RANCID will execute. Because “.cloginrc” can include the passwords for your devices, make sure you protect it from unauthorized use. Having it in your home directory helps.

RANCID won’t execute “clogin” if the permissions on “.cloginrc” are open and readable by anyone. Make the permission in the GUI file manager read/write only by the owner by typing “ls -la .cloginrc” at the command line. To change permission on the command line, type “chmod 600 .cloginrc.” Check “man chmod,” which is the short description of the chmod (file permissions) command for more information on setting file permissions.

We added our test switch into “.cloginrc” by including:
add password switch.nwc.com vtypassword enablepassword

Then we could log in to our switch using:
clogin switch.nwc.syr.edu

Be careful not to add tabs or null characters when you add lines like the one above to the “.cloginrc” file. Null characters are interpreted by “clogin” as part of the password, which will cause it to fail. If you can’t get the correct login and enable password to work, first test it manually by feeding the parameters directly into “clogin” as in the example above. If that’s successful, the passwords are correct and working with “clogin.” Then delete “add password” in “.cloginrc” and retype it, being careful not to inadvertently introduce additional characters. If you’re unfamiliar with “vi,” the default Unix editor, transfer the file to a Windows machine and edit with Notepad or another text editor.

The next step defines the user’s e-mail address to the operating system’s e-mail service. This adds two aliases for our “nwc” group into the mail */etc/aliases* file with the following two lines:

```
rancid-nwc: youremail@yourorg.com
rancid-admin-nwc: youradminemail@yourorg.com
```

These two lines resolve to the e-mail addresses listed. For the e-mail shown on the first line, RANCID sends compared switch and router files, and in the second line, any RANCID error messages. You can test whether this is working from the command line by entering the following commands:

```
“mail rancid-nwc” enter
for subject type, “test” enter
for the body type, “test” enter
Hold the “Ctrl” key and hit the “C”
```

You’ll get a “cc” line
Hold the “Ctrl” key and hit the “C” again to send

If the mail transport is working, you’ll receive the test e-mail. Then set up the files for each group, which in our case is only one “nwc”:

```
rancid-cvs
```

You now have three new directories in the *\$HOME/rancid/var* directory— CVS (Concurrent Versions System), logs and our group “nwc.” Then you must configure an empty file (router.db) for the targeted switches and routers. Using a text editor, you can add a switch or router:

```
switch.nwc.syr.edu:switch_vendor_name:up
```

The line specifies the fully qualified domain name or IP address of the targeted switch or router, the router manufacturer and the device status—whether it’s up or down.

Drum Roll, Please

It’s time to see if everything is working properly. At a command prompt, type:

```
rancid-run
```

This executable will read your “router.db” and “.cloginrc” files. It also downloads a copy of the switch configuration to *\$HOME/rancid/var/nwc*. The file will have the domain name or IP address defined in your “router.db.”

Now when you log on to a switch or router to make a change and rerun “rancid-run,” you will have another log file. You also will receive an e-mail with the configuration files “dified.” The diff e-mail is sent when a change has occurred. It looks something like the screenshot on page 107. **NWC**

Bruce Boardman, executive editor of NETWORK COMPUTING, tests and writes about network and systems management. He has 12 years’ experience managing networks and distributed computing for a financial service provider. Write to him at bboardman@nwc.com. Post a comment or question on this story at www.nwc.com/go/ask.html.

SITES TO SEE

Where to get RANCID, www.shrubbery.net/rancid/

All things Linux, www.linux.org

Red Hat, www.redhat.com

Perl and Expect source (free), www.activestate.com

All things Expect, expect.nist.gov

How-To for Linux, tldp.org