System Administration UNIX

Checking Security With COPS

Introduction

COPS is a collection of about a dozen programs that each attempt to tackle a different problem area of UNIX security.

COPS does not correct any errors found.

COPS provides a method of checking for common procedural errors. The author of COPS repeatedly stresses that it is an aid, a first line of defense, and not an impenetrable shield against security woes.

See Also

perl (1), passwd (1), touch (1), chmod (1), tar (1), pwunconv (8)

Crack, SATAN

p.246, p.257, p.628, Frisch

The Task

In this exercise you will install and configure COPS. You will inject a number of potential security lapses into your system and see whether COPS can find the problems and how it reports what it finds.

Notes:

- You will need to be root to perform this exercise.
- For the purposes of this exercise, disable shadow passwords (see the pwunconv (8) manual page for details)

Injecting Problems Into Your System

Perform the following sequence of commands, each of which offer a cracker a 'door' into your system (don't forget to substitute your 'normal' username in place of *your_username*):

```
# echo "buggy::0:0:Mr. Buggy User:/tmp:/bin/sh" >> /etc/passwd
# echo "buggy2::0:0:Another Buggy User:/tmp:/bin/sh" >> /etc/passwd
# passwd buggy2
(give buggy2 the password: hello)
# chmod 666 /etc/passwd /root/.bashrc
# touch ~your_username/.rhosts; chmod 666 ~your_username/.rhosts
# chmod 777 /etc/rc.d/rc.sysinit
```

You should think about why these commands introduce problems for a system.

Installing COPS

COPS is distributed in two formats: as C source code and as a set of PERL scripts.

In this session, you will use the PERL version—it is easier to use and is also slightly faster.

You will be given the COPS compressed tar archive file cops_1_04.tar.gz. You should place it into root's home directory and execute the following commands:

```
# cd
# tar -z -x -v -f cops_1_04.tar.gz
# cd ./cops_104/perl
```

COPS needs to examine the system on which it is installed to determine the location of the various utility programs it uses. Execute the following command:

```
# perl ./reconfig.pl
```

Checking Security With COPS

Once COPS has reconfigured itself, you should set various options by editing the file cops.cf:

- change the value for SECURE-USERS to be “root\@localhost”
- change the value for $chk_strings'recurse to be 1
- add the parameters –p ../pass.words (in this order) to the pass.chk command

You may like to look at the cops file itself, it gives you a broad idea of how the program operates:

```
# less ./cops
```

Running COPS

It is easy to run cops:

```
# perl ./cops -v
```

The version of COPS that you are using was written for an earlier version of PERL than the one that is installed with Redhat Linux—you will see a few warning messages regarding “Precedence problems” that you can safely ignore.

COPS will take a little while to run all its tests, so please be patient: it is doing quite a lot of processing!

Interpreting The Results

As we are using all the default settings, COPS will place its results into a file named according to the date when COPS was executed, in subdirectory whose name is the host machine’s name (replace *filename* with the name you see in the directory):

```
# cd `hostname`
# ls
# less filename
```

Look closely at the results. You will see a number of warnings. Many of these concern the format of the password file.

You should carefully review the warnings that are being issued—some of these will be important, some will not.

Of potential concern is the line that says:

```
Warning!  buggy2 password Problem: Guessed:            hello
```

This means that someone attempting a dictionary-based attack can access your system.

Also of concern are the lines that say:

```
Warning!  /root/.bashrc is _World_ writable!
```

```
Warning!  /home/your_username/.rhosts is _World_ writable!
Warning!  /home/your_username/.rhosts is _World_ readable!
```

```
Warning!  File /etc/rc.d/rc.sysinit (inside /etc/inittab) is _World_ writable!
```

You should think about why these lines represent potential problems…

Checking Security With COPS

The last line is important and deserves some explanation: it comes from the kuang "expert system" that COPS runs as the last step in its execution. Kuang is warning that it is possible for any user to become root by replacing (the contents of) /etc/passwd.

In Summary

You can see that COPS is a useful tool. You may wish to run it on your system regularly (as a cron job, perhaps…).

You might like to look at the carp tool to see how COPS can be used to consolidate the reports generated by running COPS on multiple systems.

Cleaning Up

It is **important** that you perform the following command after you have completed this exercise:

```
# chmod 644 /etc/passwd /root/.bashrc ~your_username/.rhosts
# chmod 755 /etc/rc.d/rc.sysinit
```

You **must** also remove the 'buggy' and buggy2 users from the password file.

Failure to do these things leaves open the possibility (albeit faint) of a cracker compromising the security of the host's site through your machine.