Sudo

"Standard UNIX takes and all-or-nothing approach to granting root access, but many times what you actually want is something in between."

— Æleen Frisch

Introduction

Sudo is a program designed to allow a system administrator to give limited root privileges to users and log root activity. This is extremely important in situations such as student laboratories…without sudo, system administrators tend to be overrun with requests for trivial (but restricted) activities such as restarting printer queues and formatting floppy disks, etc.

The basic philosophy is to give away as few extra privileges as possible but still allow people to get a task done.

See Also

su (1), mount (8), lpc (8), tail (1), shutdown (8), syslogd (8)

After installation: sudo (1), sudoers (5), visudo (8)

http://www.courtesan.com/sudo/

p.9, Frisch

The Task

In this exercise you will configure sudo. You may also need to install it beforehand.

You will need to be root for this exercise.

Testing to see if Sudo is Present

Because it has proved so useful, Sudo is nowadays often distributed with Linux (it was not always the case). To test whether Sudo is already included in your system, execute the following:

```
# rpm -q sudo
```

If the above command reports "package sudo is not installed", continue with the following section, otherwise go straight to "Configuring Sudo."

Building Sudo

As a security measure, sudo's developer does not distribute sudo as a Linux package (although these are available from other sources). It has to be compiled and installed from the source code.

You can obtain the latest version of sudo from the URL given above. This will be distributed as a compressed 'tar' format archive and will be named something like *sudo-1.6.6.tar.gz*.

You should place this file in root's home directory and extract it:

```
# cd
copy the file
# tar zxvf ./sudo-1.6.7p5.tar.gz
# cd ./sudo-1.6.7p5/
```

You should now review the INSTALL document and read the linux-specific notes near the end. You will see that these notes indicate the presence of a dependency on the particular version of the glibc library installed on the system. You can determine the version of glibc that has been installed on your system with the command:

```
rpm -qa | grep glibc
```

You should determine whether you need to edit the files indicated in the INSTALL document to get sudo to work correctly.

```
make any necessary changes
```

Once you have done this, you should allow the configure command to set up the compilation options relevant to your system (this program is a script that closely examines various commands and libraries to determine compiler options and available facilities. It also generates some source code and writes a Makefile):

```
./configure
```

After configuration, you will need to compile the source:

```
make
```

Compilation should not give any errors or warnings.

After a successful compilation (with no errors or warnings) you should install the various files that comprise the sudo system:

```
make install
```

Sudo should now be installed on your system.

Configuring Sudo

Before sudo is useful, it must be configured. When it is invoked, sudo looks at the /etc/sudoers file to determine whether a user is allowed to run a given command.

A sample /etc/sudoers file will have been installed when you issued the “make install” command, earlier.

Take a backup copy of this file and then use the visudo command to edit the original…

```
# cp /etc/sudoers /etc/sudoers.ORIG
# /usr/local/sbin/visudo
```

Note (from sudoers (5)): *“The sudoers file should always be edited by the visudo command which locks the file and does grammatical checking. It is imperative that the sudoers [file] be free of syntax errors since sudo will not run with a syntactically incorrect sudoers file.”*

Remove all the existing content from the file and insert the following (substitute your normal user name for *your_username*):

```
# Cmnd alias specification
Cmnd_Alias      MOUNT=/bin/mount,/bin/umount
Cmnd_Alias      DIE=/sbin/shutdown -[hr] now
Cmnd_Alias      LPRESTLP=/usr/sbin/lpc restart lp

# User privilege specification
root            ALL=(ALL) ALL
your_username   ALL=NOPASSWD: MOUNT
your_username   ALL=DIE, LPRESTLP
```

These commands allow user *your_username* to:

- mount and unmount any disk (sudo will not ask for a password in this case)
- halt or reboot the system
- restart the line printer

In addition, root is allowed to execute anything via sudo (think about why you would want this…).

Running Sudo

Put a formatted floppy disk in your computer's floppy disk drive and *as your normal user 'persona'* try this command:

```
% mount /dev/fd0 /mnt/floppy
```

You will see that you do not have privileges to do this. Now try:

```
% sudo mount /dev/fd0 /mnt/floppy
% sudo umount /mnt/floppy
```

This time the mount and unmount commands will proceed. You can see that sudo is allowing your normal persona to perform tasks that it would not normally be able to do.

Use /usr/local/bin/visudo to edit the /etc/sudoers file so that the last line reads:

```
your_username    ALL=MOUNT
```

issue this command:

```
% sudo -k
```

Now re-try the mount command. This time, sudo will prompt for a password before executing the command. You should give *your_username*'s password, *not* root's.

Reviewing Sudo Activity through the System Log Files

Sudo uses the standard syslog facility to leave an audit trail of everything it does.

Root can review the information that sudo records by using the command:

```
# tail /var/log/messages
```

Sudo is much more configurable than this exercise has shown but even with this brief introduction I am sure that you can see how useful sudo can be.

Clean Up

You should tidy up your directory after you have finished this exercise. Type:

```
# cd; rm -rf ./sudo-1.6.7p5
```