The Super User

The Super User

Earlier, we discussed how every file on the server has three sets of permissions: user, group, and everyone else. This restricts people from accessing files that ar critical to the system, or other people's private files. It would be awfully difficult to administer a system if you could only access and change your own files—or if everyone could access and change everyone else's files! There is one person who has access to every file on the server though: the System Administrator. The administrator has access to the super user account with the login name of *root*. Along with the power granted to the super user comes great responsibility. Having access to every file on the system means that you could destroy any file there. A couple of wrong keystrokes and you may have to re-install the whole system from scratch!

If that scares you a little bit, well...it should. It's important to remember to be extremely careful when you don't have any restrictions.

Sudo

So how do you get super user privileges? In modern Linux distributions, that task is accomplished with the **sudo** command (it's a contraction of "super do," but some pronounce it so it rhymes with "judo"). Prefixing an operation with the sudo command allows you to perform that operation as the super user. The first time you invoke sudo, it will ask for your password before it performs the operation you've requested. Subsequent calls to sudo will not ask for your password for up to five minutes as a convenience, but after that period of time you'll need to enter your password again. Here's an example that illustrates how sudo works:

INTERACTIVE SESSION:

```
[username@username-m0 ~]$ sudo whoami
[sudo] password for username:
root
```

The **whoami** command would normally return your username, but because it was run with sudo, it returned "root," which is the super user's username. But wait, if any user can just run the sudo command and supply their own password, doesn't that make the system really insecure? Good question!

The Sudoers File

The short answer is "no." Not every user can employ sudo to elevate their privileges. Access to the sudo command is controlled by the **sudoers** file, which is located at **/etc/sudoers**. This file contains a list of users and groups, and the commands upon which they can use sudo. The sudoers file is editable only by someone with super user privileges, so regular users cannot add themselves to the file. There is also a special

program used to edit the sudoers file, called **visudo**. You can edit the sudoers file by hand, but I strongly discourage it. If a mistake is made in the sudoers file it could mess up you system; it can be pretty difficult to recover from such mistakes. The visudo command edits the file and then does syntax checking on the file to ensure that all entries in the file are compliant. Keep in mind that even syntactically correct entries can have unintended and unwanted consequences, so be really careful when you edit the file.

A More Convenient Way to Use Sudo

Using **sudo** to run a command here or there is fine, but what happens when you have several tasks to perform that require you to have elevated privileges? Sudo provides a mechanism that allows you to keep your elevated privileges for an extended period of time while running sudo only once: the **-s** option. It causes sudo to launch a new shell with super user privileges:

INTERACTIVE SESSION:

```
[username@username-m0 ~]$ sudo -s
[sudo] password for username:
[root@username-m0 username]# cd
[root@username-m0 ~]# whoami
root
[root@username-m0 ~]# pwd
/root
[root@username-m0 ~]# exit
exit
[username@username-m0 username]#
```

When you start a shell using **sudo**, you have essentially become root within that shell. You can go anywhere on the system and edit any of the files. Be *very* careful with this power. An inadvertant "rm *" in the wrong place can render your machine unusable.

Source: http://courses.oreillyschool.com/sysadmin2/sudo.html