Sudoers

Introduction

Conteúdo

The `/etc/sudoers` file controls who can run what commands as what users on what machines and can also control special things such as whether you need a password for particular commands. The file is composed of aliases (basically variables) and user specifications (which control who can run what).

Editing the sudoers file

For Ubuntu 8.04: The default editor for visudo has changed to **vi**, which may cause confusion to those who are not familiar to its command bindings. To change this behavior, open your `~/.bashrc` file, and append the line to the bottom of the file:

```
export EDITOR="nano"
```

If you wish to use another text editor, replace "nano" with any text editor of your choice. Run `source ~/.bashrc` to ensure the changes you made have taken effect.

Now launch visudo with `sudo -E visudo` in the terminal, and put in `Defaults editor=/usr/bin/nano` in the file.

So the top of the file should now look like this:

```
Defaults    editor=/usr/bin/nano
Defaults    env_reset
```

Again, if you wish to use another text editor, replace "nano" with the full path to the text editor of your choice. Save, and from now on launching visudo with `gksu visudo` for Ubuntu; `kdesu visudo` for Kubuntu or `sudo visudo` in the terminal will now open with your chosen editor.

For Ubuntu 8.10: The default editor for visudo has changed to **sensible-editor**. sensible-editor defaults to nano now, and **select-editor** (which will run by default the first time) allows you to select another one.

To setup the default editor for visudo.

```
sudo select-editor
```

You will get a prompt to choose which editor you want.

```
Select an editor.  To change later, run 'select-editor'.
  1. /usr/bin/vim.tiny
  2. /bin/ed
  3. /bin/nano        <---- easiest
  4. /usr/bin/vim.basic

Choose 1-4 [3]:
```

If you don't consider yourself competent in any of the enlisted terminal-based editors, it is recommended that you choose option 3, nano.

Because sudo is such a powerful program you must take care not to put anything formatted incorrectly in the file. To prevent any incorrect formatting getting into the file you should edit it using the command `visudo` run as root or by using sudo.

```
sudo visudo
```

The sudoers file is read in one pass so when multiple entries match for a user, they are applied in order. Where there are conflicting values, the last match is used (which is not necessarily the most specific match). Also you must set an alias before you can use it. Normally you will set the aliases at the beginning of the file and then set the user specifications after all the aliases are laid out.

You can insert comments by prefixing them with a # but this is also used to specify a uid in certain parts of the file when it is followed by a number.

Aliases

There are four kinds of aliases: User_Alias, Runas_Alias, Host_Alias and Cmnd_Alias. Each alias definition is of the form:

> Alias_Type NAME = item1, item2, ...

Where Alias_Type is one of User_Alias, Runas_Alias, Host_Alias or Cmnd_Alias. A name is a string of uppercase letters, numbers and underscores starting with an uppercase letter. You can put several aliases of the same type on one line by separating them with colons (:) as so:

> Alias_Type NAME1 = item1, item2 : NAME2 = item3

You can include other aliases in an alias specification provided they would normally fit there. For example you can use a user alias wherever you would normally expect to see a list of users (for example in a user or runas alias).

There are also built in aliases called ALL which match everything where they are used. If you used ALL in place of a user list it matches all users for example. If you try and set an alias of ALL it will be overridden by this built in alias so don't even try.

User Aliases

User aliases are used to specify groups of users. You can specify usernames, system groups (prefixed by a %) and netgroups (prefixed by a +) as follows:

```
# Everybody in the system group "admin" is covered by the alias ADMINS
User_Alias ADMINS = %admin
# The users "tom", "dick", and "harry" are covered by the USERS alias
User_Alias USERS = tom, dick, harry
# The users "tom" and "mary" are in the WEBMASTERS alias
User_Alias WEBMASTERS = tom, mary
# You can also use ! to exclude users from an alias
# This matches anybody in the USERS alias who isn't in WEBMASTERS or ADMINS aliases
User_Alias LIMITED_USERS = USERS, !WEBMASTERS, !ADMINS
```

Runas Aliases

Runas Aliases are almost the same as user aliases but you are allowed to specify users by uid's. This is helpful as usernames and groups are matched as strings so two users with the same uid but different usernames will not be matched by entering a single username but can be matched with a uid. For example:

```
# UID 0 is normally used for root
# Note the hash (#) on the following line indicates a uid, not a comment.
Runas_Alias ROOT = #0
# This is for all the admin users similar to the User_Alias of ADMINS set earlier
# with the addition of "root"
Runas_Alias ADMINS = %admin, root
```

Host Aliases

A host alias is a list of hostname, ip addresses, networks and netgroups (prefixed with a +). If you do not specify a netmask with a network the netmask of the hosts ethernet interface(s) will be used when matching.

```
# This is all the servers
Host_Alias SERVERS = 192.168.0.1, 192.168.0.2, server1
# This is the whole network
Host_Alias NETWORK = 192.168.0.0/255.255.255.0
# And this is every machine in the network that is not a server
Host_Alias WORKSTATIONS = NETWORK, !SERVER
# This could have been done in one step with
# Host_Alias WORKSTATIONS = 192.168.0.0/255.255.255.0, !SERVERS
# but I think this method is clearer.
```

Command Aliases

Command aliases are lists of commands and directories. You can use this to specify a group of commands. If you specify a directory it will include any file within that directory but not in any subdirectories.

The special command '"sudoedit"' allows users to run sudo with the `-e` flag or as the command `sudoedit`. If you include command line arguments in a command in an alias these must exactly match what the user enters on the command line. If you include any of the following they will need to be escaped with a backslash (\): ",", "\", ":", "=".

Examples:

```
# All the shutdown commands
Cmnd_Alias SHUTDOWN_CMDS = /sbin/poweroff, /sbin/reboot, /sbin/halt
# Printing commands
Cmnd_Alias PRINTING_CMDS = /usr/sbin/lpc, /usr/sbin/lprm
# Admin commands
Cmnd_Alias ADMIN_CMDS = /usr/sbin/passwd, /usr/sbin/useradd, /usr/sbin/userdel, /usr/sbin/usermod, /usr/sbin/visudo
# Web commands
Cmnd_Alias WEB_CMDS = /etc/init.d/apache2
```

User Specifications

User Specifications are where the sudoers file sets who can run what as who. It is the key part of the file and all the aliases have just been set up for this very point. If this was a film this part is where all the key threads of the story come together in the glorious unveiling before the final climatic ending. Basically it is important and without this you ain't going anywhere.

A user specification is in the format

```
<user list> <host list> = <operator list> <tag list> <command list>
```

The user list is a list of users or a user alias that has already been set, the host list is a list of hosts or a host alias, the operator list is a list of users they must be running as or a runas alias and the command list is a list of commands or a cmnd alias.

The tag list has not been covered yet and allows you set special things for each command. You can use PASSWD and NOPASSWD to specify whether the user has to enter a password or not and you can also use NOEXEC to prevent any programs launching shells themselves (as once a program is running with sudo it has full root privileges so could launch a root shell to circumvent any restrictions in the sudoers file.

For example (using the aliases and users from earlier)

```
# This lets the webmasters run all the web commands on the machine
# "webserver" provided they give a password
WEBMASTERS webserver= WEB_CMDS
# This lets the admins run all the admin commands on the servers
ADMINS SERVERS= ADMIN_CMDS
# This lets all the USERS run admin commands on the workstations provided
# they give the root password or and admin password (using "sudo -u <username>")
USERS WORKSTATIONS=(ADMINS) ADMIN_CMDS
# This lets "harry" shutdown his own machine without a password
harry harrys-machine= NOPASSWD: SHUTDOWN_CMDS
# And this lets everybody print without requiring a password
ALL ALL=(ALL) NOPASSWD: PRINTING_CMDS
```

The Default Ubuntu Sudoers File

The sudoers file that ships with Ubuntu 8.04 by default is included here so if you break everything you can restore it if needed and also to highlight some key things.

```
# /etc/sudoers
#
# This file MUST be edited with the 'visudo' command as root.
#
# See the man page for details on how to write a sudoers file.
#

Defaults    env_reset

# Uncomment to allow members of group sudo to not need a password
# %sudo ALL=NOPASSWD: ALL

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root    ALL=(ALL) ALL
```

```
# Members of the admin group may gain root privileges
%admin ALL=(ALL) ALL
```

This is pretty much empty and only has three rules in it. The first (`Defaults env_reset`) resets the terminal environment after switching to root. So, ie: all user set variables are removed. The second (`root ALL=(ALL) ALL`) just lets root do everything on any machine as any user. And the third (`%admin ALL=(ALL) ALL`) lets anybody in the admin group run anything as any user. Note that they will still require a password (thus giving you the normal behaviour you are so used to).

If you want to add your own specifications and you are a member of the admin group then you will need to add them after this line. Otherwise all your changes will be overridden by this line saying you (as part of the admin group) can do anything on any machine as any user provided you give a password.

Common Tasks

This section includes some common tasks and how to accomplish them using the sudoers file.

Shutting Down From The Console Without A Password

Often people want to be able to shut their computers down without requiring a password to do so. This is particularly useful in media PCs where you want to be able to use the shutdown command in the media centre to shutdown the whole computer.

To do this you need to add some cmnd aliases as follows:

```
Cmnd_Alias SHUTDOWN_CMDS = /sbin/poweroff, /sbin/halt, /sbin/reboot
```

You also need to add a user specification (at the end of the file after the "`%admin ALL = (ALL) ALL`" line so it takes effect - see above for details):

```
<your username> ALL=(ALL) NOPASSWD: SHUTDOWN_CMDS
```

Obviously you need to replace "<your username>" with the username of the user who needs to be able to shutdown the pc without a password. You can use a user alias here as normal.

Multiple tags on a line

There are times where you need to have both NOPASSWD and NOEXEC or other tags on the same configuration line. The man page for sudoers is less than clear, so here is an example of how this is done:

```
myuser ALL = (root) NOPASSWD:NOEXEC: /usr/bin/vim
```

This example lets the user "myuser" run as root the "vim" binary without a password, and without letting vim shell out (the :shell command).

Enabling Visual Feedback when Typing Passwords

As of Ubuntu 10.04 (Lucid), you can enable visual feedback when you are typing a password at a sudo prompt.

Simply edit `/etc/sudoers` and change the *Defaults* line to read:

```
Defaults        env_reset,pwfeedback
```

Troubleshooting

If your changes don't seem to have had any effect, check that they are not trying to use aliases that are not defined yet and that no other user specifications later in the file are overriding what you are trying to accomplish.

CategorySecurity

Sudoers (editada pela última vez em 2014-12-11 10:54:25 por i121-115-220-98)