Tune Environment Variables

Information

These notes were originally written in the year 2000 as part of a set of LPI Exam 101 training materials. The LPI training course at Bromley College was subsequently discontinued and some of the sections of the notes modified and incorporated into our one-day System Administration Courses. The remainder of the notes have now been made publicly available on the linuxtraining.org.uk website.

If you are a beginner please do not be put off of training courses by these notes, as they are rather technical. On the other hand if you are a more experienced Linux user we hope you find the coverage of this topic refreshingly clear.

For full details of our current Linux training please visit the site:
http://ce.bromley.ac.uk/linux

If you have reached this page from a search engine and wish to see the full contents list for the published notes please visit the site:
http://www.linuxtraining.org.uk

We hope you find these notes useful, but please remember that they apply to the 2.2 kernel. I will update them when I have the time.

Clive Gould - 21st December 2004

Tune Environment Variables

Objective 2

Tune the user environment and system environment variables: Modify global and user profiles to set environment variables, maintain skel directories for new user accounts, place proper commands in path. Involves editing / etc/profile and /etc/skel.

System Environment Variables

The default system wide environment variables, which apply to all users, are defined in the file **/etc/profile**.

Any changes made to environment variables in this file will affect all users of the system.

An example of an /etc/profile file for a typical distribution is illustrated below:

```
¤ [clive@redhat clive]$ cat /etc/profile
# /etc/profile

# System wide environment and startup programs
# Functions and aliases go in /etc/bashrc

PATH="$PATH:/usr/X11R6/bin"
PS1="[\u@\h \W]\\$ "

ulimit -c 1000000
if [ `id -gn` = `id -un` -a `id -u` -gt 14 ]; then
          umask 002
else
          umask 022
fi

USER=`id -un`
LOGNAME=$USER
MAIL="/var/spool/mail/$USER"
HOSTNAME=`/bin/hostname`
HISTSIZE=1000
HISTFILESIZE=1000
```

Tune Environment Variables

```
INPUTRC=/etc/inputrc
export PATH PS1 HOSTNAME HISTSIZE HISTFILESIZE USER LOGNAME
MAIL INPUTRC

for i in /etc/profile.d/*.sh ; do
          if [ -x $i ]; then
                    . $i
          fi
done

unset i
```

By editing this file, as root, you can change system wide environment variables such as PATH, PS1, HISTSIZE and HISTFILESIZE. When any user next logs on the new environment variables will be read from /etc/profile and take affect.

Profiles and User Environment Variables

When a new user account is created the files are copied over from the **/etc/skel** directory into the users home directory. The contents of /etc/skel are illustrated below for a typical distribution:

```
[clive@redhat skel]$ ls -la
total 17
drwxr-xr-x   4 root     root         1024 Sep 16  1999 .
drwxr-xr-x  42 root     root         4096 Aug 31 13:53 ..
-rw-r--r--   1 root     root         1422 Mar 29  1999 .Xdefaults
-rw-r--r--   1 root     root           24 Jul 14  1994 .bash_logout
-rw-r--r--   1 root     root          230 Aug 22  1998 .bash_profile
-rw-r--r--   1 root     root          124 Aug 23  1995 .bashrc
drwxr-xr-x   3 root     root         1024 Sep 16  1999 .kde
-rw-r--r--   1 root     root          966 Apr 16  1999 .kderc
-rw-rw-r--   1 root     root         3505 Apr  7  1999 .screenrc
drwxr-xr-x   5 root     root         1024 Sep 16  1999 Desktop
```

If you use the useradd command, the copying of these files takes place automatically If you set up user accounts by hand, you will need to copy these directories manually and change the permissions appropriately.

By editing the contents of the /etc/skel directory, as root, you can change the default settings which will be applied to all new user accounts. Feel free to place other files here if your new user accounts should have them.

Tune Environment Variables

Local environment variables, which apply to a particular user are defined in a variety of alternative files in the users home directory. Whenever an interactive login shell is started bash first reads the global file /etc/profile. It then looks for the file **~/.bash_profile** and tries to read it. If this file doesn't exist bash looks in turn for **~/.bash_login** and if that doesn't exist either, it looks for **~/.profile**.

An example of typical .bash_profile file is illustrated below:

```
[clive@redhat clive]$ cat .bash_profile
# .bash_profile

# Get the aliases and functions
if [ -f ~/.bashrc ]; then
          . ~/.bashrc
fi

# User specific environment and startup programs
PATH=$PATH:$HOME/bin
BASH_ENV=$HOME/.bashrc
USERNAME=""

export USERNAME BASH_ENV PATH
```

You can see that this file calls the **~/.bashrc** file. (Remember that you can put aliases in the .bashrc file). As a user you can add your own user specific (local) environment variable definitions in .bash_profile, or one of the alternative profile files. For your logon these definitions will take precedence over any environment variable definitions in /etc/profile.

If the file **.bash_logout** exists it is run when exiting an interactive login shell. An example of a typical .bash_logout file is illustrated below:

```
[clive@redhat clive]$ cat .bash_logout
# ~/.bash_logout
clear
```

In the above example all that happens is that the clear command is run to clear the screen.

Note: If an interactive, non-login shell is started (by typing bash at the command prompt), just the file ~/.bashrc will be read, if it exists.