User Management & Software Management

by Mohamad H. Danesh

Managing Users

- Each system has two kinds of users:
  - Superuser (root)
  - Regular user
- Each user has his own username, password, and permissions that can only be assigned by the user.
- All users have a user ID (UID) and a group ID (GID).

The Password File

- /etc/passwd
- It is the database file for all users on the system.
- Username:password:uid:gid:comment:homedir:shell
- * in password means disable.
- x in password means it is included in /ets/shadow.

Shadow Passwords

- /etc/shadow
- It is considered to use the encrypted passwords found in /etc/passwd.
- Only x or * appears in the password field of /etc/passwd.

Shadow Password Fields

- The user's login name
- The encrypted password
- The number of days since jan 1970
- The number of days before the the password can be changed
- The number of days before the password is to expire that the user is warned it will expire.
- The number of days after the password expires the account is disabled.
- The number of days since jan 1 1970 that account has been disabled.

Shadow Password Fields Example

- Shadow file example:
  - smithj:Ep6mckrOLChF:10063:0:99999:7:::

Groups

- User groups are a convenient way to logically organize sets of user accounts and allow users to share files within their group or groups.
- Each file on the system has both a user and a group owner associated with it.
- Every user is assigned to at least one group
- User has "group access" to any files on the system with a group ID contained in his list of groups (default and additional groups)

Groups

- /etc/group
- The custom is to use GIDs of 500 or more for regular users and less for administrations.
- Groupname:password:gid:users
  - root:x:0:root
  - bin:x:1:root,bin,daemon
  - test:x:500:

User Management Commands

- useradd
  - Create a new user
- userdel
  - Delete a user
- usermod
  - Modify a user account
- passwd
  - Modify a user password
- groupadd
  - Create a new group
- groupdel
  - Delete a group
- groupmod
  - Modify a group
- grpck
  - verify integrity of group files

Users' Home Directory

- When each user is created, a home directory is created for him (/home/<username>).
- The set of files that initially are used to populated this home directory are kept in /etc/skel.

Some Useful Commands

- chmod
  - Change file access permission
- chown
  - Change file owner or group
- chroot
  - Change root directory

Some Useful Commands

- setuid and setgid
  - When an executable file's setuid permission is set, users may execute that program with a level of access that matches the user who owns the file.
  - chmod u+s myfile

GUI Administration Tool

- Creating users
  - KDE: K, System, Kuser
  - GNOME: Main Menu, KDE menus, System, Kuser
- Change Password
  - KDE: K, System, Change Password
  - GNOME: Main Menu, Programs, System, Change Password

The su Command

- It is necessary for regular users to run a command as if they were root.
- The su means substitute user.
- This command changes the UID and GID of the existing user.
- The syntax for the su command is this:

  su option username arguments
- To return to the regular users' identity
  - exit

The sudo Command

- It gives to the certain users only a few superuser permissions.
- The list of authorized users is kept in /etc/sudoers
- Sudo will prompt for a password and then check the /etc/sudoers.
- Sample:
  - Sudo fdisk /dev/hda1

Communicating With Users

- wall
  - Sends a message to the terminals of all user connected to the system.
  - Wall "message"
- talk
  - Allows two-way communication between any two users.
  - talk username
- write
  - Send a message to a user
  - write username

Assignment (1)

- Write an shell script for adding user
  - Create users home directory
  - Delete user
    - Note: don’t forget to find every files which belong to this user and remove them or change owner to root, such as: home directory,…
- Note:
  - Write this script as an function which user be able to call it from shell
    - Guide: in .bashrc file at $HOME
  - It should be interactive with user, asking username, password, home directory.

Installing Software

- Install Manually
  - Install Binary Distribution
  - Compile & Install from Source
- RPM Package
- Debian Packages

Basics to install (manually)

- Archive and Compression Utilities
  - tar and compress
  - gzip and bzip2

Using gzip and bzip2

- gzip is a fast and efficient compression program
  - .gz filename extension
  - gzip test.txt
- How efficiently a file is compressed depends upon its format and contents
  - gzip –l test.txt.gz
- Uncompress
  - gunzip tst.txt.gz
  - gzip –d test.txt.gz

Using gzip

- specify the speed and quality of the compression
  - -1 (fast) specifies the fastest method, which compresses the files less compactly
  - -9 (best) uses the slowest, but best compression method.
  - default is -6.

Using tar

- tar is a general-purpose archiving utility
  - Stands for Tape Archive
  - capable of packing many files into a single archive file,
  - while retaining information needed to restore the files fully, such as file permissions and ownership.

- The format of the tar command:
  - tar functionoptions files

Using tar(2)

- tar functionoptions files
- function can be one of the following:

c	To create a new archive
x	To extract files from an archive
t	To list the table of contents of an archive
r	To append files to the end of an archive
u	To update files that are newer than those in the archive
d	To compare files in the archive to those in the filesystem

- the more commonly used are c, x, and t.

Using tar(3)

- The most common options are:
- v To print verbose information when packing or unpacking archives.
- k To keep any existing files when extracting — that is, to not overwrite any existing files which are contained within the tar file.
- f *filename* To specify that the tar file to be read or written is *filename*.

Using tar(4)

- Examples:
- tar cvf result.tar dirName
  - Note: If you use v multiple times, additional information will be printed
  - If you don't specify f filename at all, tar assumes for historical reasons that it should use the device /dev/rmt0 (that is, the first tape drive).
- tar xvf result.tar
  - Note: The new files will be owned by the user running the tar xvf (you) unless you are running as root, in which case the original owner is preserved.

Package Installation

- Compile from source code
  - ./configure
  - make
  - make install
  - [optional] make check
- Install Binary Packages
  - create (or choose) a directory to install the package
  - extract the contents of the package.
    - tar commands
  - Set environment variable,.. If needed
- Install using RPM or Debian package system

Advance Package Managers

- Update Automatically, automatic dependency resolution and signed packages,…
- Debian: apt
  - apt-get [install/remove/update] packagename
- Red Hat: yum
  - yum [install/remove/update] packagename

Assignment (2)

- Write an shell script to install and erase this package: http://curl.haxx.se/download/curl-7.32.0.tar.gz
  - use tar commands