1.111.1
Manage users and
group accounts and
related system files
Weight 4

Nick Urbanik

1.111.1
Manage users and group accounts and related system files
Weight 4

Linux Professional Institute Certification — 102

Nick Urbanik <nicku@nicku.org>

This document Licensed under GPL—see section 15

2005 October

Outline

1.111.1
Manage users and
group accounts and
related system files
Weight 4

Nick Urbanik

Topic 111 Administrative Tasks [21]
Where we are up to

1.111.1 **Manage users and group accounts and related system files [4]**

1.111.2 Tune the user environment and system environment variables [3]

1.111.3 Configure and use system log files to meet administrative and security needs [3]

1.111.4 Automate system administration tasks by scheduling jobs to run in the future [4]

1.111.5 Maintain an effective data backup strategy [3]

1.111.6 Maintain system time [4]

1.111.1
Manage users and
group accounts and
related system files
Weight 4

Nick Urbanik

Description of Objective
1.111.1 Manage users and group accounts and related system files [4]

Candidate should be able to *add*, *remove*, *suspend* and *change* user accounts. Tasks include to add and remove groups, to change user/group info in passwd/group databases. The objective also includes creating *special purpose* and *limited accounts*.

1.111.1
Manage users and
group accounts and
related system files
Weight 4

Nick Urbanik

Key files, terms, and utilities include: I
1.111.1 Manage users and group accounts and related system files [4]

1.111.1
Manage users and
group accounts and
related system files
Weight 4

Nick Urbanik

/etc/passwd — text file containing user account information

/etc/shadow — text file containing user password information

/etc/group — text file containing groups

/etc/gshadow — text file that may contain group passwords

chage — change user password expiry information

gpasswd — change group membership, group passwords

groupadd — create a new group

groupdel — delete an existing group

groupmod — modify a group

grpconv — moves all group password information to /etc/gshadow

grpunconv — creates group from group and gshadow and then removes gshadow.

Key files, terms, and utilities include: II
1.111.1 Manage users and group accounts and related system files [4]

1.111.1
Manage users and
group accounts and
related system files
Weight 4

Nick Urbanik

passwd — set or change passwords to authenticate users

pwconv — moves all user password information from /etc/passwd to /etc/shadow

pwunconv — moves all password information from /etc/shadow to /etc/passwd then deletes /etc/shadow

useradd — create a new user account

userdel — delete an existing user account

usermod — modify a user account

The account information files

1.111.1
Manage users and
group accounts and
related system files
Weight 4

Nick Urbanik

/etc/passwd — text file containing user account information

/etc/shadow — text file containing user password
information

/etc/group — text file containing groups

/etc/gshadow — text file that may contain group passwords

- ▶ Note that all are simple text files that can be edited with an editor (best to use vipw and vigr so that file is not edited while others are changing it)
- ▶ These are for *local* accounts only: network accounts may be obtained through LDAP, Samba or active directory through Winbind, NIS or NIS+ or Hessiod, to name a few.
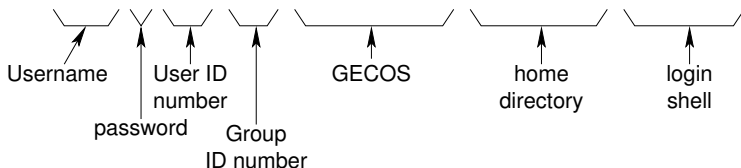
/etc/passwd

- The `passwd` file is documented with
  $ **man 5 passwd** ↵
- Each line of the file corresponds to one user; here is mine:

```
nicku:x:1000:1000:Nick Urbanik:/home/nicku:/bin/bash
```

Username

password

User ID number

Group ID number

GECOS

home directory

login shell

Fields in `/etc/passwd`

user name — account name. Tradition has it in lower case.

password — This is always 'x' unless you have run `pwunconv` to move the hashed password from the `shadow` file back here. You can change it to a star '*' to *disable* the account *if `shadow` is not used*

user ID number — an integer that uniquely identifies a particular user to the system. Call it UID

group ID number — an integer that uniquely identifies the *primary group* of this user to the system.

GECOS — holds the user's actual name, and perhaps their phone number, or any information about the user you like! Called GECOS for hysterical reasons: read
$ **man 5 passwd** ↩

home directory — after the user logs in, this directory is made their current directory

login shell — after logging in, the user has this shell. Usually the shell should be listed in `/etc/shells`

1.111.1
Manage users and
group accounts and
related system files
Weight 4

Nick Urbanik

More about the `passwd` fields

- The computer identifies files, processes, . . . by the UID and GID.
- the passwd file (or its equivalent) is the only link to the account name.
- the group ID number links the account to the one group that (by default) owns files and processes created by the user
- Although you can suspend an account to prevent logging in by replacing the login shell with something like /bin/false or /bin/nologin, a user can su to this account.

The /etc/group file

1.111.1
Manage users and
group accounts and
related system files
Weight 4

Nick Urbanik

▶ Here are two lines from my /etc/group file:

```
nicku:x:1000:
linusgames:x:516:linus,pam,nicku
```

▶ The first says group with GID 1000 has the name nicku.
It has no members, except for the user for whom this is the
primary group ID

▶ The second line maps the group name linusgames to
the GID 516. It has the members with user names linus,
pam and nicku.

Primary Group

1.111.1
Manage users and
group accounts and
related system files
Weight 4

Nick Urbanik

- Every user has a *primary group*
- This is the default group attached to any files or processes created by the user
- A member can belong to any number of *secondary groups*
- An example from earlier: `nicku` has a primary group called `nicku` and a secondary group called `linusgames`.
- You can change your group to any of your other groups with the `newgrp` command.

`newgrp`: Changing to other Groups

1.111.1
Manage users and
group accounts and
related system files
Weight 4

Nick Urbanik

- ▶ A group may have a password associated with it
  - ▶ I do not recommend shared passwords, hence do not use group passwords
  - ▶ A shared secret remains a secret only if no one else is interested
- ▶ The password is put into `/etc/gshadow`
- ▶ If group has a password associated with it, a user who is not a member can change to this group using the `newgrp` command by entering the password when prompted.
- ▶ group members can change their current group to a group they are a member of using the `newgrp` command regardless of whether there is a password with that group.
- ▶ Group passwords are created using the `gpasswd` command.

/etc/shadow must be readable only by root

- The /etc/shadow file must be *readable only by root*
- This is to avoid other people getting a copy of all the hashed passwords and running Crack or John the Ripper to recover passwords at leisure

```
$ ls -l /etc/shadow ↩
-rw-------  1 root root 2085 Aug 24 13:13 /etc/shadow
```

Fields in `/etc/shadow`

1.111.1
Manage users and
group accounts and
related system files
Weight 4

Nick Urbanik

From $ **man 5 shadow** ↩ the nine fields are:

- login name
- encrypted password
    - This is incorrect, wrong,... and makes me splutter!!
    - It is a *hash* of the password
    - Prefix with an exclamation mark '!' to *disable an account temporarily*.
- days since Jan 1, 1970 that password was last changed
- days before password may be changed
- days after which password must be changed
- days before password is to expire that user is warned
- days after password expires that account is disabled
- days since Jan 1, 1970 that account is disabled
- a reserved field

Making a user account

Any method of creating an account goes through the following steps (assuming the use of local files to hold account information)

1. Find the next available UID and GID numbers, or use the ones provided, checking they are unique
2. Add an entry to the `/etc/passwd` and `/etc/shadow` files using all the information provided, including a hash of the password into `/etc/shadow`
3. Create the home directory
4. Create a mail spool file `/var/spool/mail/`⟨*username*⟩
5. Copy the files and directories from `/etc/skel` to the home directory
6. Change the ownership of the home directory and all its contents to the user, and the group ownership to the primary group of the user
7. Change the ownership of the mail spool file to the user, and make the group owner equal to `mail`

/usr/sbin/useradd

- On Red Hat/Fedora (and some other UNIX systems), useradd does all the above, although you need to create a hash of the password beforehand
- On Debian systems, the program adduser is more capable, and useradd less so
- See $ **man useradd** ↩, $ **man adduser** ↩
- Make an account for me:

  ```
  $ sudo useradd -c "Nick Urbanik" nicku ↩
  $ sudo passwd nicku ↩
  Changing password for user nicku.
  New password:
  Retype new password:
  passwd: all authentication tokens updated successfully.
  ```

Differences between Debian and Red Hat
`useradd`

- On Debian systems, you need to specify the $-m$ option to `useradd` or the home directory will not be created.
- People use `adduser` instead on Debian systems.

/usr/sbin/usermod

- You can modify the account parameters in the /etc/passwd file for an existing account using usermod.
- See $ **man usermod** ↩

/usr/sbin/groupadd

- You can create a new group with: $ **sudo groupadd** ⟨*groupname*⟩ ↵
- Note that useradd (and adduser on Debian/Ubuntu) will automatically create the primary group for a user if it does not already exist

/usr/sbin/groupdel

1.111.1
Manage users and
group accounts and
related system files
Weight 4

Nick Urbanik

▶ You can remove an existing group with: $ **sudo groupdel** ⟨*groupname*⟩ ↩

Adding a user to a group

1.111.1
Manage users and
group accounts and
related system files
Weight 4

Nick Urbanik

- ▶ It may seem that usermod is the best tool, but it actually removes the user from any groups not specified!
- ▶ Use gpasswd instead :-)
- ▶ Syntax: # **gpasswd -a** ⟨*user*⟩ ⟨*group*⟩ ↩
- ▶ To add the user nicku to the group linusgames without removing nicku from any existing group memberships:

$ **sudo gpasswd -a nicku linusgames** ↩

/usr/sbin/userdel

1.111.1
Manage users and
group accounts and
related system files
Weight 4

Nick Urbanik

▶ To delete the `nicku` account *including the home directory*:

$ **sudo userdel -r nicku** ↩

Suspending an account

1.111.1
Manage users and
group accounts and
related system files
Weight 4

Nick Urbanik

- ▶ You can suspend ("lock") a shadow account by inserting an exclamation mark '!' in front of the password field in /etc/shadow using vipw
- ▶ ... or you can use $ **sudo passwd -l** ⟨*username*⟩ ↩ to do the same thing
- ▶ You can unlock the account by removing the '!' either manually with vipw or with
  $ **sudo passwd -u** ⟨*username*⟩ ↩

Setting the password expiry information

- ▶ The easiest program to use for this is chage
- ▶ You can also use passwd to change some password information.
- ▶ Ordinary users can use $ **chage -l** ↩ to read the account aging information for their own account.

Creating special purpose accounts I

- ▶ A number of special system accounts are needed:, e.g.,

```
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
news:x:9:13:news:/etc/news:
uucp:x:10:14:uucp:/var/spool/uucp:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
gopher:x:13:30:gopher:/var/gopher:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
vcsa:x:69:69:virtual console memory owner:/dev:/sbin/nologin
```

1.111.1
Manage users and
group accounts and
related system files
Weight 4

Nick Urbanik

Creating special purpose accounts II

```
nscd:x:28:28:NSCD Daemon:/:/sbin/nologin
rpm:x:37:37::/var/lib/rpm:/sbin/nologin
haldaemon:x:68:68:HAL daemon:/:/sbin/nologin
netdump:x:34:34:Network Crash Dump user:/var/crash:/bin/bash
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
rpc:x:32:32:Portmapper RPC user:/:/sbin/nologin
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
nfsnobody:x:65534:65534:Anonymous NFS
User:/var/lib/nfs:/sbin/nologin
mailnull:x:47:47::/var/spool/mqueue:/sbin/nologin
smmsp:x:51:51::/var/spool/mqueue:/sbin/nologin
pcap:x:77:77::/var/arpwatch:/sbin/nologin
xfs:x:43:43:X Font Server:/etc/X11/fs:/sbin/nologin
ntp:x:38:38::/etc/ntp:/sbin/nologin
gdm:x:42:42::/var/gdm:/sbin/nologin
```

- ▶ These accounts generally have a user ID that is lower than some particular value
- ▶ Use the user ID numbers and names recommended by the distribution to avoid unintentional conflicts

Creating special purpose accounts III

- See `/usr/share/doc/setup-*/uidgid` on Red Hat/Fedora systems

1.111.1
Manage users and
group accounts and
related system files
Weight 4

Nick Urbanik

Creating limited accounts

1.111.1
Manage users and
group accounts and
related system files
Weight 4

Nick Urbanik

- ▶ Network servers such as Apache, Sendmail, Postfix, Samba, Bind, ntpd,... all run under special accounts that have limited access to the system
- ▶ You may need to create accounts for users who are just there for accessing email by POP3 or IMAP, or just for Samba
- ▶ To do this: create an account with a login shell of `/bin/false` (or possibly `/sbin/nologin`), and a disabled password.

Topics Covered

1.111.1
Manage users and
group accounts and
related system files
Weight 4

Nick Urbanik

License Of This Document

1.111.1
Manage users and
group accounts and
related system files
Weight 4

Nick Urbanik