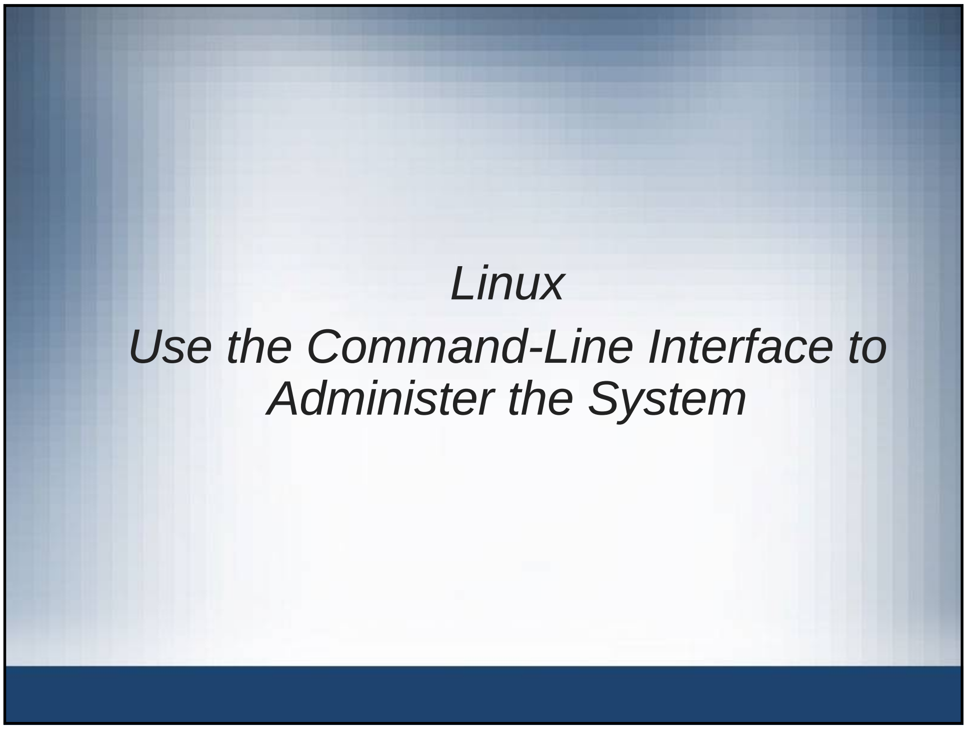
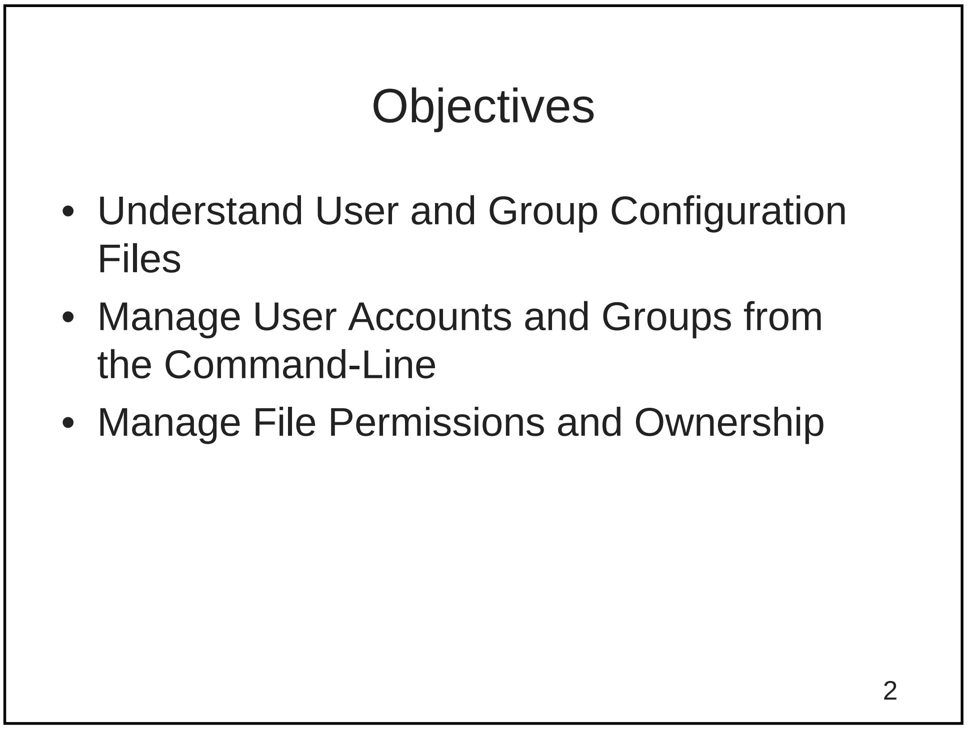


Linux

Use the Command-Line Interface to Administer the System



Objectives

- Understand User and Group Configuration Files
- Manage User Accounts and Groups from the Command-Line
- Manage File Permissions and Ownership

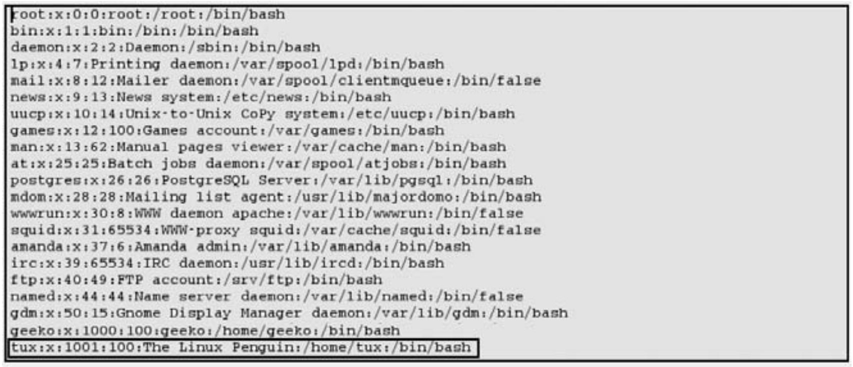
2

Understand User and Group Configuration Files

- Information on users and groups is kept in:
  - /etc/passwd
  - /etc/shadow
  - /etc/group
- You should not modify these files with an editor
  - Use YaST or the appropriate command-line tools
  - Modifying these files with an editor can lead to errors
  - To ensure consistency, you should be able to:
    - Check /etc/passwd and /etc/shadow
    - Convert Passwords to and from Shadow

3

/etc/passwd

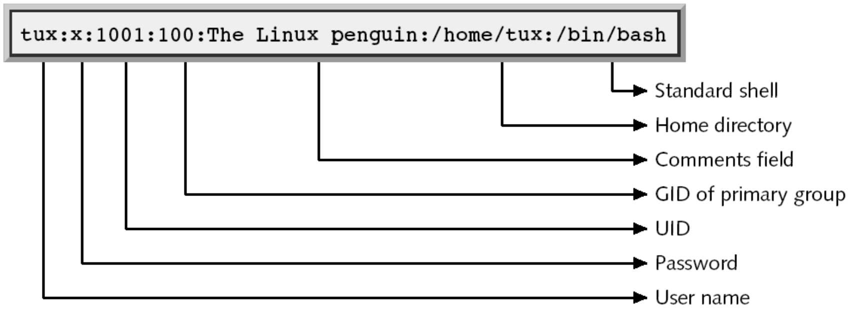


```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/bin/bash
daemon:x:2:2:Daemon:/sbin:/bin/bash
lp:x:4:7:Printing daemon:/var/spool/lpd:/bin/bash
mail:x:8:12:Mailer daemon:/var/spool/clientmqueue:/bin/false
news:x:9:13:News system:/etc/news:/bin/bash
uucp:x:10:14:Unix-to-Unix CoPy system:/etc/uucp:/bin/bash
games:x:12:100:Games account:/var/games:/bin/bash
man:x:13:62:Manual pages viewer:/var/cache/man:/bin/bash
at:x:25:25:Batch jobs daemon:/var/spool/atjobs:/bin/bash
postgres:x:26:26:PostgreSQL Server:/var/lib/pgsql:/bin/bash
mdom:x:28:28:Mailing list agent:/usr/lib/majordomo:/bin/bash
wwwrun:x:30:8:WWW daemon apache:/var/lib/wwwrun:/bin/false
squid:x:31:65534:WWW-proxy squid:/var/cache/squid:/bin/false
amanda:x:37:6:Amanda admin:/var/lib/amanda:/bin/bash
irc:x:39:65534:IRC daemon:/usr/lib/ircd:/bin/bash
ftp:x:40:49:FTP account:/srv/ftp:/bin/bash
named:x:44:44:Name server daemon:/var/lib/named:/bin/false
gdm:x:50:15:Gnome Display Manager daemon:/var/lib/gdm:/bin/bash
geeko:x:1000:100:geeko:/home/geeko:/bin/bash
tux:x:1001:100:The Linux Penguin:/home/tux:/bin/bash
```

Figure 7-1

4

/etc/passwd (continued)



```
tux:x:1001:100:The Linux penguin:/home/tux:/bin/bash
```

Standard shell

Home directory

Comments field

GID of primary group

UID

Password

User name

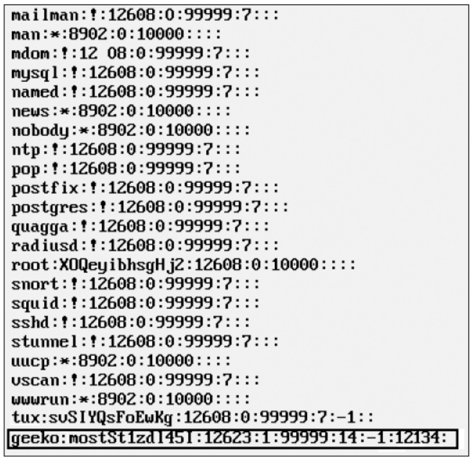
Figure 7-2

UID:
•0–99 for the system itself
•100–499 for special system users (such as services and programs)
•On SLES 9, normal users start from UID 1000

/etc/shadow

- Only root can modify it
- Root and members of the group shadow can read it
- Contains encrypted (hashed) passwords
  - Coded with *crypt*; 13 characters long
- If an invalid character occurs in password field then user cannot log in

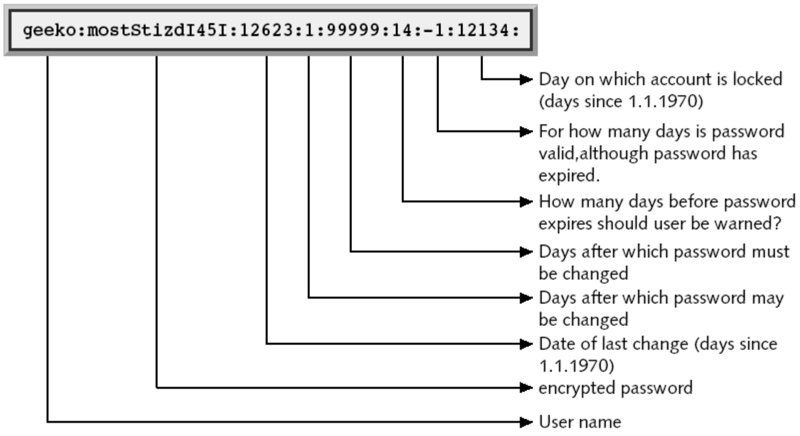
/etc/shadow (continued)



```
mailman:!:12608:0:99999:7:::
man:*:8902:0:10000::::
mdom:!:12 08:0:99999:7:::
mysql:!:12608:0:99999:7:::
named:!:12608:0:99999:7:::
news:*:8902:0:10000::::
nobody:*:8902:0:10000::::
ntp:!:12608:0:99999:7:::
pop:!:12608:0:99999:7:::
postfix:!:12608:0:99999:7:::
postgres:!:12608:0:99999:7:::
quagga:!:12608:0:99999:7:::
radiusd:!:12608:0:99999:7:::
root:XOQeyibhsgHj2:12608:0:10000::::
snort:!:12608:0:99999:7:::
squid:!:12608:0:99999:7:::
sshd:!:12608:0:99999:7:::
stunnel:!:12608:0:99999:7:::
uucp:*:8902:0:10000::::
vscan:!:12608:0:99999:7:::
wwwrun:*:8902:0:10000::::
tux:svSIYQsFoEwKg:12608:0:99999:7:-1::
geeko:mostSt1zdI45I:12623:1:99999:14:-1:12134:
```

Figure 7-3

/etc/shadow (continued)



geeko:mostStizdI45I:12623:1:99999:14:-1:12134:

Day on which account is locked (days since 1.1.1970)

For how many days is password valid,although password has expired.

How many days before password expires should user be warned?

Days after which password must be changed

Days after which password may be changed

Date of last change (days since 1.1.1970)

encrypted password

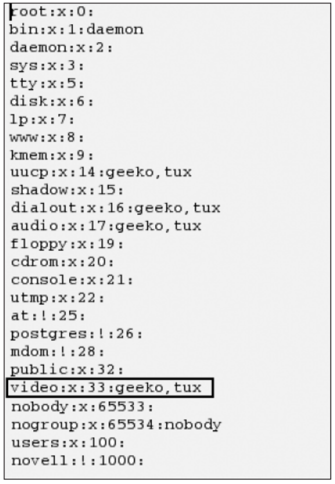
User name

Figure 7-4

/etc/group

- Each line represents a single group record
  - Group name, password hash, GID, and the members of the group
- Shows secondary group memberships only
- In older versions of SUSE LINUX, group passwords are stored in /etc/gshadow

/etc/group (continued)

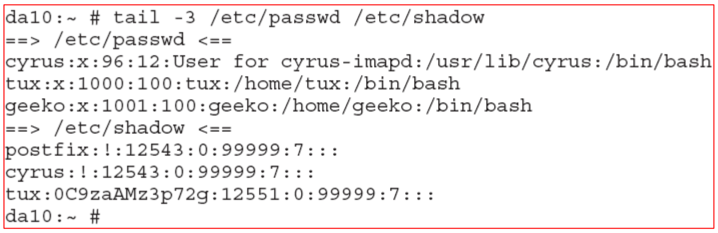


```
root:x:0:
bin:x:1:daemon
daemon:x:2:
sys:x:3:
tty:x:5:
disk:x:6:
lp:x:7:
www:x:8:
kmem:x:9:
uucp:x:14:geeko,tux
shadow:x:15:
dialout:x:16:geeko,tux
audio:x:17:geeko,tux
floppy:x:19:
cdrom:x:20:
console:x:21:
utmp:x:22:
at:!:25:
postgres:!:26:
mdom:!:28:
public:x:32:
video:x:33:geeko,tux
nobody:x:65533:
nogroup:x:65534:nobody
users:x:100:
novell:!:1000:
```

Figure 7-5

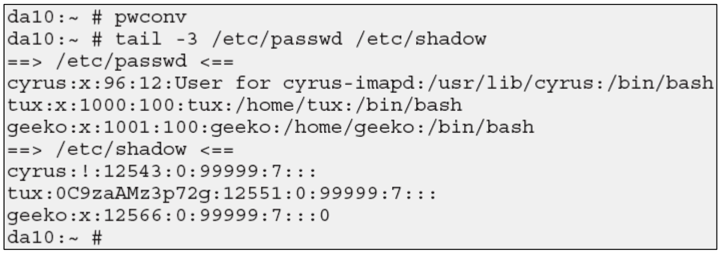
Check /etc/passwd and /etc/shadow

- Because user configuration is handled by two files (/etc/passwd and /etc/shadow), these files have to match each other
- However, discrepancies can occur



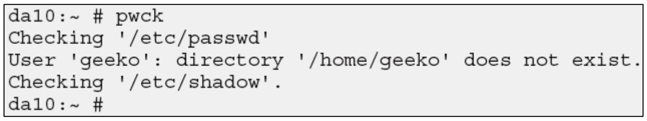
```
da10:~ # tail -3 /etc/passwd /etc/shadow
==> /etc/passwd <==
cyrus:x:96:12:User for cyrus-imapd:/usr/lib/cyrus:/bin/bash
tux:x:1000:100:tux:/home/tux:/bin/bash
geeko:x:1001:100:geeko:/home/geeko:/bin/bash
==> /etc/shadow <==
postfix:!:12543:0:99999:7:::
cyrus:!:12543:0:99999:7:::
tux:0C9zaAMz3p72g:12551:0:99999:7:::
da10:~ #
```

Check /etc/passwd and /etc/shadow (continued)



```
da10:~ # pwconv
da10:~ # tail -3 /etc/passwd /etc/shadow
==> /etc/passwd <==
cyrus:x:96:12:User for cyrus-imapd:/usr/lib/cyrus:/bin/bash
tux:x:1000:100:tux:/home/tux:/bin/bash
geeko:x:1001:100:geeko:/home/geeko:/bin/bash
==> /etc/shadow <==
cyrus:!:12543:0:99999:7:::
tux:0C9zaAMz3p72g:12551:0:99999:7:::
geeko:x:12566:0:99999:7:::0
da10:~ #
```

pwck - checks integrity of all entries in /etc/passwd and etc/shadow files



```
da10:~ # pwck
Checking '/etc/passwd'
User 'geeko': directory '/home/geeko' does not exist.
Checking '/etc/shadow'.
da10:~ #
```

Convert Passwords to and from Shadow

- Convert Password to Shadow
  - – pwconv command converts the passwd file to the shadow file
    - Replaces the password in /etc/passwd with ***x***
    - Password aging information is pulled from login.defs
  - – pwconv can also be used to add missing entries to the shadow file
- Convert Shadow to Password
  - – pwuconv moves passwords from /etc/shadow to /etc/passwd and password aging information is lost

Manage User Accounts and Groups from the Command-Line

- In addition to the YaST modules **users** and **groups**, you can use the following commands to add, change, and delete users and groups:
  - – useradd
  - – passwd
  - – usermod
  - – userdel
  - – groupadd, groupmod, and groupdel
- To prevent individual users from using system resources excessively, use the following command:
  - – ulimit

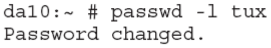
useradd

- useradd options
  - -m. /etc/skel/ used as a template for home directory
  - -c. “comment”
  - -g. GID or -g *groupname*
  - -G. defines any supplementary groups
  - -p. “encrypted password” (use mkpasswd first)
  - -e. YYYY-MM-DD: expiration date
- /etc/default/useradd

```
GROUP=1001
HOME=/home
INACTIVE=-1
EXPIRE=
SHELL=/bin/bash
SKEL=/etc/skel
GROUPS=audio,dialout,uucp,video
```

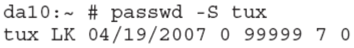
passwd

- Use without arguments to change own password
- Other options
  - -l: to lock a user account (use -u to unlock)



```
da10:~ # passwd -l tux
Password changed.
```

  - -S: to list the status of a user account



```
da10:~ # passwd -S tux
tux LK 04/19/2007 0 99999 7 0
```

passwd (continued)

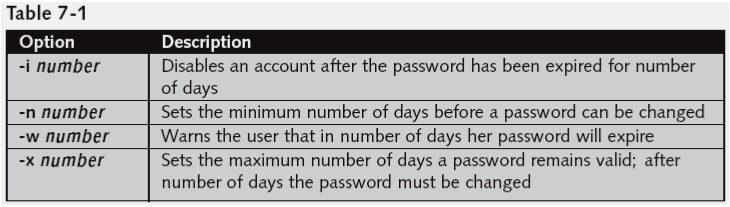
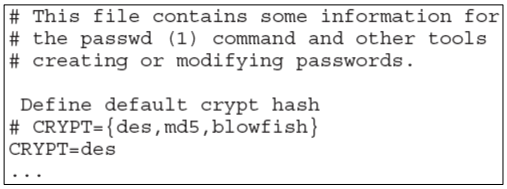


Table 7-1

Option	Description
-i *number*	Disables an account after the password has been expired for number of days
-n *number*	Sets the minimum number of days before a password can be changed
-w *number*	Warns the user that in number of days her password will expire
-x *number*	Sets the maximum number of days a password remains valid; after number of days the password must be changed

passwd (continued)

- /etc/default/passwd:



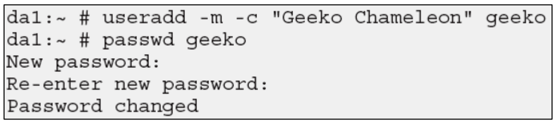
```
# This file contains some information for
# the passwd (1) command and other tools
# creating or modifying passwords.

 Define default crypt hash
# CRYPT={des,md5,blowfish}
CRYPT=des
...
```

- In SLES 9, a different algorithm (like blowfish) configured in /etc/security/pam_unix2.conf takes precedence over the one in /etc/default/passwd
  - DES supports passwords up to eight characters long
  - MD5 and Blowfish support longer passwords

passwd (continued)

- The quickest way to create a new user from a command-line is to use useradd and passwd



```
da1:~ # useradd -m -c "Geeko Chameleon" geeko
da1:~ # passwd geeko
New password:
Re-enter new password:
Password changed
```

usermod

- Used to modify information such as the UID, the standard shell, the home directory, and the primary group in an existing user account
- Its options are nearly the same as the options of the command useradd
- Examples:
  - Change the home directory:
    - usermod -d /newhome/tux -m tux
  - Change the UID:
    - usermod -u 1504 tux

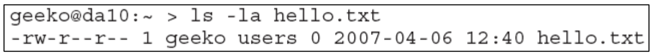
userdel

- Used to delete user accounts
  - userdel tux
- Without options, it removes the user from:
  - /etc/passwd
  - /etc/shadow
  - /etc/group
- If /var/spool/cron/tabs/*username* exists, it is deleted
- Home directory is not deleted
- To delete the user’s home directory and the data it contains:
  - userdel -r tux

groupadd, groupmod, and groupdel

- groupadd *group_name* (next free GID is used)
  - -g *GID*
  - -p *encrypted_password*
- groupmod
  - -g *newGID*
  - -n *new_group_name*
  - -A *user* (to add to group)
- groupdel *group_name*
  - You can delete a group only if no user has this group assigned as a primary group

Manage File Permissions and Ownership

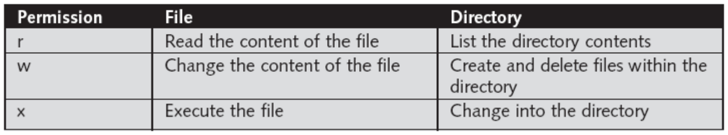


```
geeko@da10:~ > ls -la hello.txt
-rw-r--r-- 1 geeko users 0 2007-04-06 12:40 hello.txt
```

- The first 10 columns represent the following:
  - 1: File type
  - 2-4: File permissions of the user who owns the file
  - 5-7: File permissions of the owning group of the file
  - 8-10: File permissions of others

Manage File Permissions and Ownership (continued)

Table 7-2



Permission	File	Directory
r	Read the content of the file	List the directory contents
w	Change the content of the file	Create and delete files within the directory
x	Execute the file	Change into the directory

Change the File Permissions with chmod

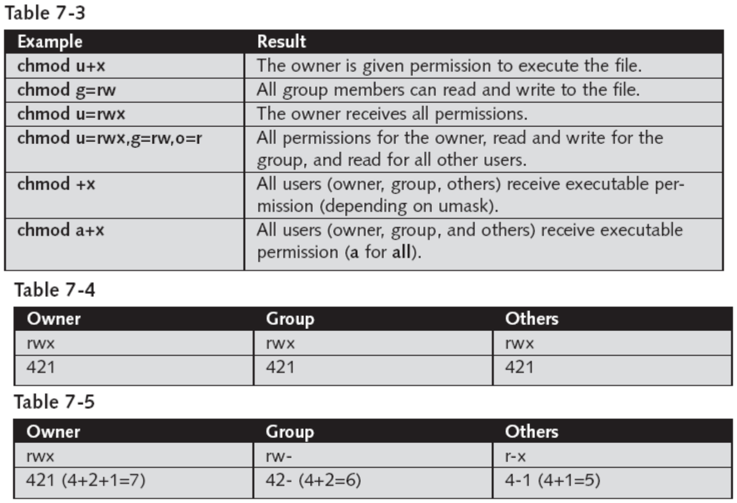


Table 7-3

Example	Result
chmod u+x	The owner is given permission to execute the file.
chmod g=rw	All group members can read and write to the file.
chmod u=rwx	The owner receives all permissions.
chmod u=rwx,g=rw,o=r	All permissions for the owner, read and write for the group, and read for all other users.
chmod +x	All users (owner, group, others) receive executable permission (depending on umask).
chmod a+x	All users (owner, group, and others) receive executable permission (**a** for **all**).

Table 7-4

Owner	Group	Others
rwx	rwx	rwx
421	421	421

Table 7-5

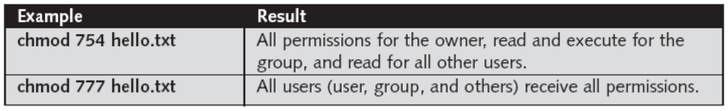
Owner	Group	Others
rwx	rw-	r-x
421 (4+2+1=7)	42- (4+2=6)	4-1 (4+1=5)

Change the File Permissions with chmod (continued)

- With the option -R and a specified directory, you can change the access permissions of all files and subdirectories under the specified directory
- If you have a certain set of permissions in mind that the file should have, the octal syntax is usually the most efficient

Change the File Permissions with chmod (continued)

Table 7-6



Example	Result
chmod 754 hello.txt	All permissions for the owner, read and execute for the group, and read for all other users.
chmod 777 hello.txt	All users (user, group, and others) receive all permissions.

Change the File Ownership with chown and chgrp

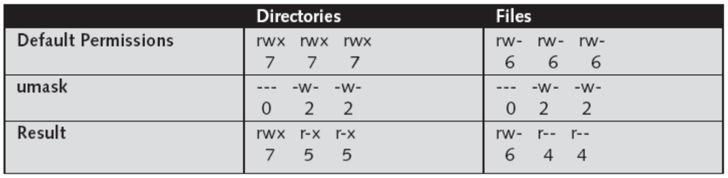
- User root can use chown and chgrp as follows:
  - chown *new_user*.*new_group file*
  - chown *new_user file*
  - chown .*new_group file*
  - chgrp *new_group file*
- A normal user can change the group affiliation of a file that he owns to a new group
  - chown .*new_group file*
  - chgrp *new_group file*
  - The user can only change the group affiliation of the file that he owns if he is a member of the new group

Modify Default Access Permissions

- By default, files are created with the access mode **666** and directories with **777**
- To modify these default access mode settings, use ***umask***
  - umask command allows you to specify the permissions that will be given to all files and folders created after issuing the command
  - The permissions set in the umask are removed from the default permissions

Modify Default Access Permissions (continued)

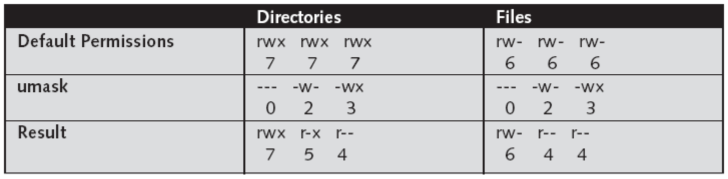
Table 7-7



	Directories	Files
Default Permissions	rwx rwx rwx 7 7 7	rw- rw- rw- 6 6 6
umask	--- -w- -w- 0 2 2	--- -w- -w- 0 2 2
Result	rwx r-x r-x 7 5 5	rw- r-- r-- 6 4 4

Modify Default Access Permissions (continued)

Table 7-8



	Directories	Files
Default Permissions	rwx rwx rwx 7 7 7	rw- rw- rw- 6 6 6
umask	--- -w- -wx 0 2 3	--- -w- -wx 0 2 3
Result	rwx r-x r-- 7 5 4	rw- r-- r-- 6 4 4

Modify Default Access Permissions (continued)

- **umask 077** restricts access to the owner and root
- To make umask setting permanent, change its value in /etc/profile
- To make the setting user-specific, enter the value of umask in the file .bashrc in the home directory of the respective user

Configure Special File Permissions

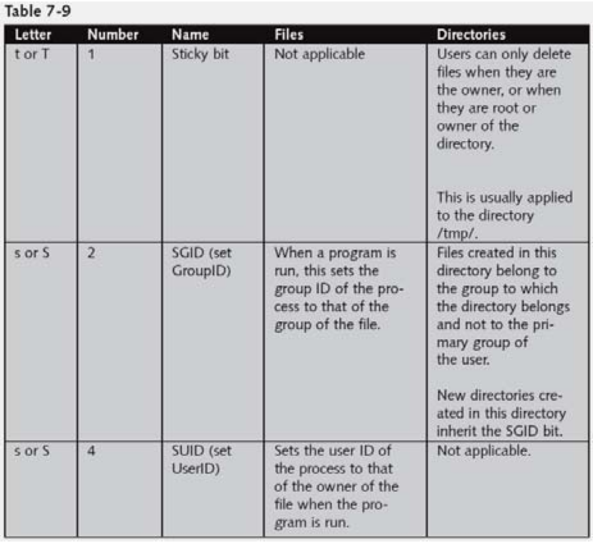
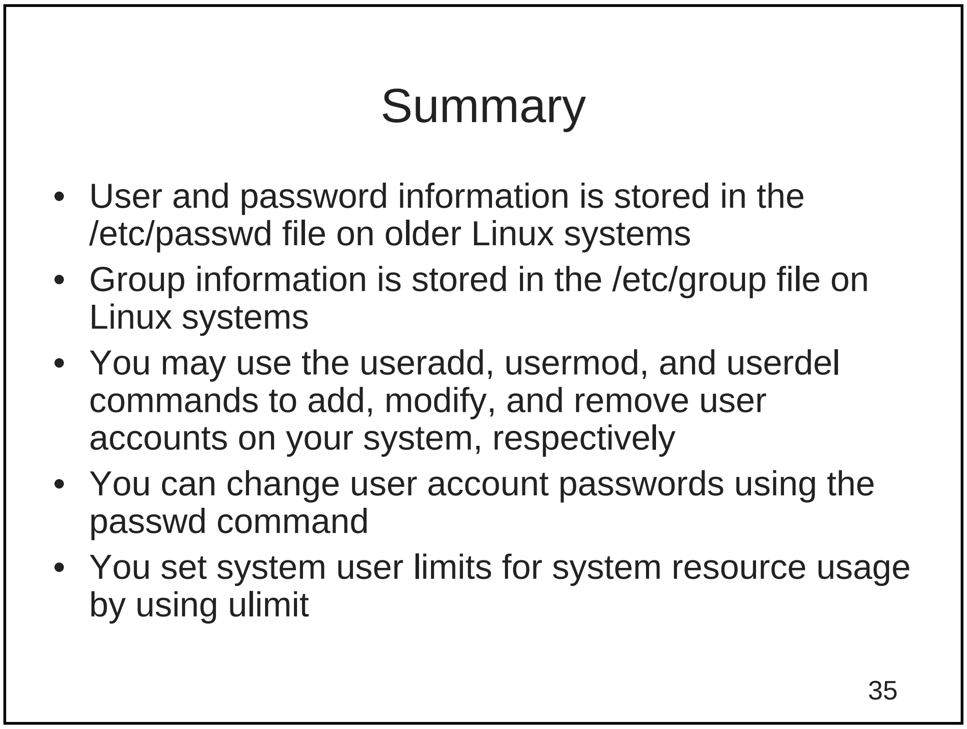


Table 7-9

Letter	Number	Name	Files	Directories
t or T	1	Sticky bit	Not applicable	Users can only delete files when they are the owner, or when they are root or owner of the directory. This is usually applied to the directory /tmp/.
s or S	2	SGID (set GroupID)	When a program is run, this sets the group ID of the process to that of the group of the file.	Files created in this directory belong to the group to which the directory belongs and not to the primary group of the user. New directories created in this directory inherit the SGID bit.
s or S	4	SUID (set UserID)	Sets the user ID of the process to that of the owner of the file when the program is run.	Not applicable.

Configure Special File Permissions (continued)

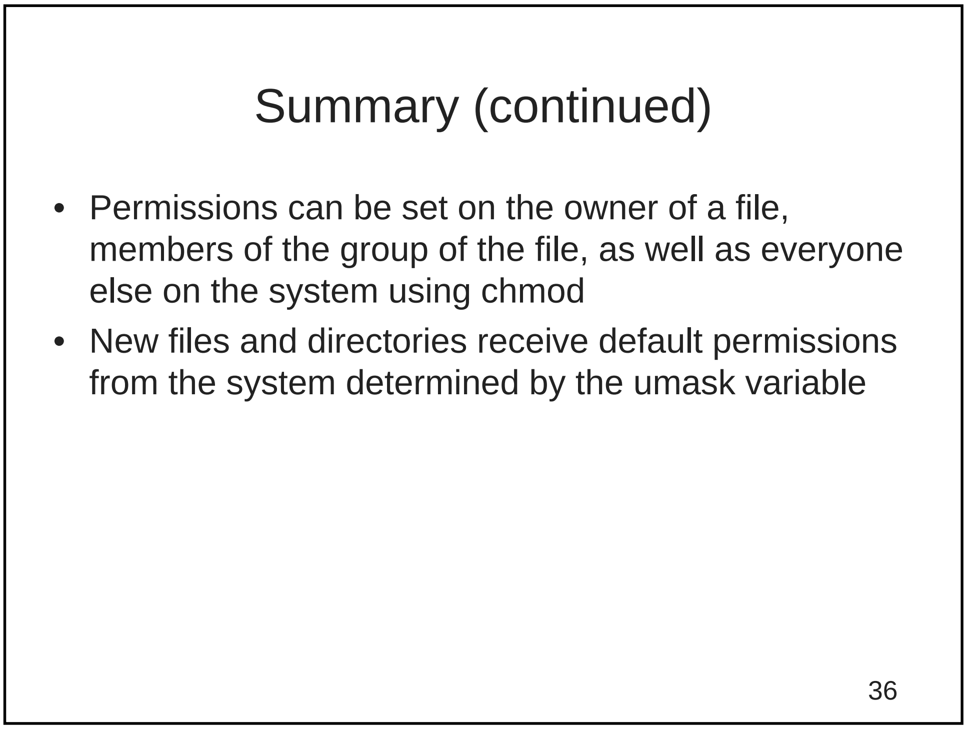
- To set the sticky bit
  - chmod o+t /tmp
  - chmod 1777 /tmp
  - The sticky bit is listed in the permissions for others
- To set the SUID
  - chmod u+s /usr/bin/passwd
  - chmod 4755 /usr/bin/passwd
- To set the SGID
  - chmod g+s /usr/bin/wall
  - chmod 2755 /usr/bin/wall



Summary

- User and password information is stored in the /etc/passwd file on older Linux systems
- Group information is stored in the /etc/group file on Linux systems
- You may use the useradd, usermod, and userdel commands to add, modify, and remove user accounts on your system, respectively
- You can change user account passwords using the passwd command
- You set system user limits for system resource usage by using ulimit

35



Summary (continued)

- Permissions can be set on the owner of a file, members of the group of the file, as well as everyone else on the system using chmod
- New files and directories receive default permissions from the system determined by the umask variable

36