7.1 Users and Groups

Type	Description
Standard user	Standard user accounts can log in to the system. Standard user accounts: • Generally have easy-to-remember usernames (such as Mary or bkaun). An administrator must create the usernames. • Have an ID of 500 or more (on some distributions) or 1,000 or more (on other distributions). The ID is automatically assigned by the system when the account is created.
System or **Service accounts**	System user accounts (also called service user accounts) are created by default during the Linux installation and are used by the system for specific roles. System user accounts: • Have names that correspond with their roles, such as ftp and mail. • Can't be used to log in to the system. • Have an ID of 500 or less (on some distributions) or 1,000 or less (on other distributions). The ID is automatically assigned by the system when the account is created. The root user account is created by default and has a UID of 0. However, the root user account can be used to log in to the system and perform tasks.
Primary group	Primary groups (also called the private group) are created by default on most Linux distributions when a standard user is created. These accounts help manage access to files and directories. Primary groups: • Have the corresponding user as the only member. • Are automatically assigned as the owner of files and directories when they are created in the file system. • Are similar to any other group (the only difference is that the group is identified as the primary group in the user account's configuration).
Secondary group	Secondary groups are also used to manage access to files and directories. Secondary groups: • Are not automatically assigned user accounts as members. • Receive their membership as assigned by the system administrator.

7.1 User and Group Databases

File	Description
/etc/passwd	The /etc/passwd file holds user account information. Be aware of the following details: • Each entry identifies a user account. • Each entry contains multiple fields, with each field separated by a colon. The following line is a sample entry in the /etc/passwd file: **pclark:x:501:501:Petunia Clark:/home/pclark:/bin/bash** The fields within this line are as follows: • User account name • Password (an "x" in the field indicates that passwords are stored in the /etc/shadow file) • User ID number • Primary group ID number (also known as the GID) • Description field (this field is typically used for the user's full name) • Path to the home directory • Path to the default shell
/etc/shadow	The /etc/shadow file holds password hashes and password expiration information for user accounts. Be aware of the following details: • Using the /etc/shadow file to separate usernames from password hashes increases the security of the user passwords. • Like the /etc/passwd file, each entry corresponds to a user account and each entry contains multiple fields, with each field separated by a colon. The following line is a sample entry in the /etc/shadow file: **pclark:$ab7Y56gu9bs:12567:0:99999:7:::** The fields within this line are as follows: • User account name • Password hash ◦ A $ preceding the password identifies the password as an encrypted entry ◦ A ! or !! indicates that the account is locked and can't be used to log in ◦ A * indicates a system account entry and can't be used to log in • Last change (the date of the most recent password change measured in the number of days since January 1, 1970) • Minimum password age (the minimum number of days the user must wait before changing the password) • Maximum password age (the maximum number of days between password changes) • Password change warning (the number of days a user is warned before the password must be changed) • Grace logins (the number of days the user can log in without changing the password) • Disable time (the number of days since January 1, 1970, after which the account will be disabled)

/etc/group

The /etc/group file holds group information including the group name, GID, and group membership information. Be aware of the following details:

- Each entry identifies a group.
- Each entry contains multiple fields, with each field separated by a colon.

The following line is a sample entry in the /etc/group file:
sales:x:510:pclark,mmckay,hsamson

The fields within this line are as follows:

- Group name
- Group password (an x indicates the group passwords are contained in the /etc/gshadow file)
- Group ID
- Group members (contains a comma-separated list of user accounts that are members of the group)

/etc/gshadow

The /etc/gshadow file holds password hashes for groups. Be aware of the following details:

- Like the /etc/group file, each line corresponds to a group.
- Each line consists of fields separated by colons.

The following line is a sample entry in the /etc/gshadow file:
sales:!:pclark:pclark,mmckay,hsamson

The fields within this line are as follows:

- Group name
- Group password (the group password allows users to add themselves as members of the account)
  - If the field contains a single exclamation point (!), the group account can't be accessed with the password
  - If the field contains a double exclamation point (!!), no password has been assigned to the group account (and it can't be accessed with the password)
  - If there is no value, only group members can log in to the group account
- Administrators (contains a comma-separated list of users who have authorization to administer the account)
- Group members (contains a comma-separated list of user accounts that are members of the group)

7.1 Managing Password File Entries

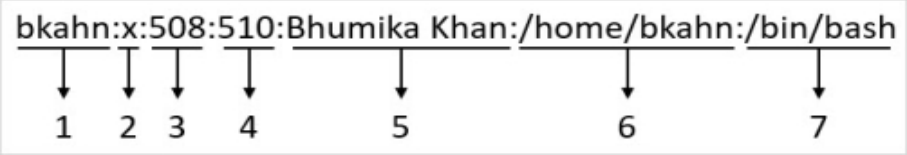
Command	Description
pwck	Verifies the entries in the /etc/passwd and /etc/shadow files to ensure that they have the proper format and contain valid data. Errors are displayed on the screen, and entries may be deleted to solve the errors. For example, checks are made to verify that each entry has: • The correct number of fields • A unique and valid user name • A valid user and group identifier • A valid primary group • A valid home directory • A valid login shell
pwconv	The pwconv command is used to move passwords from the less-secure /etc/passwd file to the more secure /etc/shadow file. You can do the opposite action with the pwunconv command. Doing this also removes the shadow file. Today, however, virtually all Linux distributions ship with shadow files enabled by default. The synchronization process is as follows: • The entries in the shadowed file that do not exist in the passwd file are removed. • The shadowed entries that don't have "x" as the password in the passwd file are updated. • Any missing shadowed entries are added. • Passwords found in the passwd file are replaced with "x".

7.2 Manage Users and Passwords

Command	Description
useradd	Creates a user account. The following options override the settings found in /etc/default/useradd: **-c** adds text for the account in the Description field of /etc/passwd. This option is commonly used to specify the user's full name. **-d** assigns an absolute pathname to a custom home directory location. **-D** displays the default values specified in the /etc/default/useradd file. **-e** specifies the date on which the user account will be disabled. **-f** specifies the number of days after a password expires until the account is permanently disabled. **-g** defines the primary group membership. **-G** defines the secondary group membership. **-M** does not create the user's home directory. **-m** creates the user's home directory (if it does not exist). **-n** do not create a group with the same name as the user (Red Hat and Fedora, respectively). **-N** " " **-p** defines the encrypted password. **-r** specifies that the user account is a system user. **-s** defines the default shell. **-u** assigns the user a custom UID. This is useful when assigning ownership of files and directories to a different user.
passwd	Assigns or changes a password for a user. • **passwd** (without a username or options) changes the current user's password. • Users can change their own passwords. The root user can execute all other passwd commands. Be aware of the following options: **-S** username displays the status of the user account. ◦ LK indicates that the user account is locked. ◦ PS indicates that the user account has a password. **-l** disables (locks) an account. This command inserts a !! before the password in the /etc/shadow file, effectively disabling the account. **-u** enables (unlocks) an account. **-d** removes the password from an account. **-n** sets the minimum number of days that a password exists before it can be changed. **-x** sets the number of days before a user must change the password (password expiration time). **-w** sets the number of days the user is warned before the password expires. **-i** sets the number of days following the password expiration that the account will be disabled.

usermod	Modifies an existing user account. **usermod** uses several of the same switches as **useradd**. Be aware of the following switches: **-a** appends the user to the supplementary groups specified with the -G option. **-c** changes the description for the account. This is usually used to modify the user's full name. **-d home_dir** assigns the user a new home directory. If -d is used with the -m option, the contents of the user's current home directory will be moved to the new home directory. **-e date** specifies the date when the account will be disabled. **-f** specifies the number of days after a password expires until the account is permanently disabled. **-g** specifies the primary group membership. **-G** specifies the secondary group membership. This option is usually used in conjunction with the -a option. If you don't use the **-a** option, **-G** will overwrite all existing supplementary group memberships. **-l** renames a user account. When renaming the account: ◦ Use **-d** to rename the home directory. ◦ Use **-m** to copy all files from the existing home directory to the new home directory. **-L** locks the user account. This command inserts a ! before the password in the /etc/shadow file, effectively disabling the account. **-m** moves the contents of the user's home directory to the new location specified by the **-d** option. **-p password** assigns the specified encrypted password to the account. **-s shell** sets the user's default login shell. **-u UID** assigns a new user ID number. **-U** unlocks the user account.
userdel	Removes the user from the system. Be aware of the following options: • **userdel username** (without options) removes the user account. **-r** removes the user's home directory. **-f** forces the removal of the user account even when the user is logged into the system.

7.2 User Account Management Files

File	Description
/etc/default/useradd	The /etc/default/useradd file contains default values for the useradd utility when creating a user account, including: • Group ID • Home directory • Account expiration • Default shell • Secondary group membership • Skeleton directory
/etc/login.defs	The /etc/login.defs file defines: • Values for defining allowed group and user ID numbers. • Protocols to be used for password encryption in the shadow file. • Password aging values for user accounts. • The path to the default mailbox directory. • Whether a home directory should be created by default.
/etc/skel	The /etc/skel directory contains a set of configuration file templates that are copied into a new user's home directory when it is created, including the following files: • .bashrc • .bash_logout • .bash_profile • .kshrc
/etc/passwd	The /etc/passwd file contains user account information. The file includes one line for each user on the system. Viewing this file lets you verify a user account and specific information about that account. You can view this file using the following command: **cat /etc/passwd**  1. Username – the name used to log on to a system. 2. Password – when an X is shown, the password is encrypted. 3. User ID (UID) – A unique number identifying the user. UID 0 is for the root user. 1-99 are reserved for predefined accounts. 100-999 are for system accounts and groups. 4. Group ID (GID) – the primary group ID. 5. User ID Info (GECOS) – used to hold extra account information, such as the full name and phone numbers. 6. Home directory – the path to the user's home directory. When a user logs on, they're taken to this directory by default. 7. Command/shell – the path to the command or shell being used.

7.3 Group Management

Command	Function
groupadd	Creates a new group. The options below override the settings found in /etc/login.defs: **-g** defines the group ID (GID). **-p** defines the group password. **-r** creates a system group.
groupmod	Modifies a group definition. Be aware of the following options: **-n** changes the name of a group. **-A** adds specified users from the group (not available on all distributions). **-R** removes specified users from the group (not available on all distributions).
groupdel	Deletes a group.
gpasswd	Changes a group password. Be aware of the following options: • **groupname** prompts for a new password. **-r** removes a group password.
newgrp	Changes your current or real group ID to the group ID specified in the command. As long as the user knows the group password, this lets them switch to a different group without being added to it. Typing **exit** removes the user from the group.
usermod	Modifies group membership for the user account. Be aware of the following options: **-g** assigns a user to a primary group. **-G** assigns a user to a secondary group (or groups). Follow the command with a comma-separated list of groups. If the user already belongs to any secondary groups, the user will be removed from those groups if the groups are not in the list. **-aG** assigns a user to a secondary group (or groups) by appending them to any groups the user already belongs to. Follow the command with a comma-separated list of groups. **-G ""** removes the user from all secondary group memberships. Do not include a space between the quotes.
groups	Displays the primary and secondary group membership for the specified user account.