7.2.4 User Management Facts

This lesson covers the following topics:

- Managing users and passwords
- User account management files
- Troubleshooting user access

Managing Users and Passwords

Although it is possible to manage user accounts by manually editing the /etc/passwd and /etc/shadow files, if not done properly, doing so can disable your system.

The best practice is to manage user account using the graphic user interface (GUI) tools or by using commands from the shell prompt as follows:

Command	Description	Example
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).	**useradd pmaxwell** creates the pmaxwell user account**.** **useradd -c "Paul Morril" pmorril** creates the pmorril account with a comment**.** **useradd -d /tmpusr/sales1 sales1** creates the sales1 user account with home directory located at /tmpusr/sales1. **useradd -u 789 dphilips** creates the dphilips account with user ID 789.

	• **-n, N** does not create a group with the same name as the user (Red Hat and Fedora, respectively). • **-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 change 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 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 before the password expires that the user is warned.
	• **-i** sets the number of days following the password expiration that the account will be disabled.
		passwd jsmith changes the password for the jsmith account.
		passwd -d removes the password from an account.
		passwd -d jsmith removes the password from the jsmith account.
		passwd -x 40 jsmith requires jsmith to change his password every 40 days.
		passwd -n 10 jsmith means that jsmith cannot change his password for 10 days following the most recent change.
		passwd -w 2 jsmith means that jsmith will be warned 2 days before his password expires.
		passwd -i 7 jsmith disables the jsmith account after 7 days if the password is not changed.
		passwd -l jsmith locks the jsmith account.
		passwd -u jsmith unlocks the jsmith account.

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, then **-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.
		usermod -c "Paul Morril" pmorril changes the comment field for user pmorril.
		usermod -l esmith -d /home/esmith -m ejones renames the ejones account to esmith, renames the home directory, and moves the old home directory contents to the new location.
		usermod -s /bin/tsch esmith points the shell for esmith to /bin/tsch.
		usermod -U esmith unlocks the esmith 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.	**userdel pmaxwell** deletes the pmaxwell account while leaving the home directory on the hard drive. **userdel -r pmorril** removes both the account and the home directory.

User Account Management Files

Be aware of the following configuration files when managing user accounts:

File	Description
/etc/default/useradd	The /etc/default/useradd file contains default values used by 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 used to define 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

Troubleshooting User Access

At times, a user may have difficulties logging into a system. When this happens the following may be helpful:

Local Access

- Verify that the username and password being entered are correct. When doing this, ensure that the proper capitalization for both the name and password are being used, since both are case sensitive.
- From a shell prompt (terminal), verify that the user's account has not been locked and that a password has been assigned by typing: **sudo grep** *username* **/etc/shadow**

  Example: **sudo grep mary /etc/shadow**
  Result: **mary: ! ! 6OLrJmRgu$4hiY8j ehfAAZ1m3v4T4/OWkJ j IJ6XHYaRErwrhGnY5/eXH2ba6Xj rL11/ : 17940 : O : : 7 : : :**

  The account is locked if there are two exclamation marks (!!) after the user name. If this is the case, and you know the account should not be locked, use this command to unlock the account: **passwd -u** *username*

  If there are two exclamation marks, but no encrypted password, a password will need to be assigned (see the passwd command above).

- In some cases, the graphical user interface (GUI) may have issues (such as a broken graphics driver) preventing a user from logging in. In this case, open a tty session and test a no GUI login:
  - To access a tty login, press **Ctrl** + **Alt** + **F**# (where # is some number, typically 2-9).

- From the tty prompt, try logging in using the user's account and password. If the login is unsuccessful, take the proper steps to troubleshoot the GUI.

Remote Access

Remote access can be accomplished using a variety of methods. For the purpose of this lesson, troubleshooting remote access will be limited to connecting using Secure Shell (ssh). When creating a remote connection, you can use the ssh command from a Linux computer or use a utility, such as PUTTY on a Windows computer. When you connection attempt fails, consider the following:

- Verify the hostname is properly spelled or if you are using an IP address, that the IP address is correct.
- Verify that you can resolve the hostname or IP address from your client machine using the ping command. If the ping command fails, your issue is probably a network issue and not ssh. Example: **ping mywks** or **ping 192.16.8.1.25**
- Verify that your network supports connectivity over the ssh port being used. For example, the default port of 22, may be blocked.
- Verify that a firewall is not blocking your connection or the desired port.
- Verify that the ssh daemon is enabled and running (and bound to the correct port).
  - To see if the daemon is running enter: **sudo systemctl status sshd.service**
    Note that the name of your daemon may be different.
    The output will show you if the daemon is running and if so, on what port it is listening.
  - To see if the daemon exist run: **sudo systemctl list-unit-files | grep ssh**
    Look for the ssh daemon, such as sshd.service. Verify that the daemon is enabled.
    If needed, enable the daemon and start the daemon as follows:
    systemctl enable *daemon_name*
    systemctl start *daemon_name*
  - If the port is incorrect, make the applicable changes in the /etc/ssh/sshd.config file.