Module 18 Linux server - Manage user and Groups and working with file systems

**Assignment Level Basic**

1. ***What is default uid for root user?***

*Ans. Root user’s default uid is ‘0’.*

1. ***What is default uid for system user?***

*Ans. The user UID identifies the username to any system on which the user attempts to log in.*

1. ***What is the uid for normal users?***

*Ans. Normal user’s uid is 10000+.*

1. ***How to add comment in user file?***

*Ans. To add comments in a user file in Linux, you can use the '#' symbol. Comments are lines in a file that are ignored by the system and are intended for human readers to provide explanatory notes or context. Here's how you can add comments to a user file:*

1. *Open a terminal.*
2. *Use a text editor like nano, vi, or vim to open the user file. For example, to open the file "user.txt" with nano, you can run the following command:*  
   ***nano user.txt***
3. *Move the cursor to the line where you want to add the comment.*
4. *Add a '#' symbol at the beginning of the line. For example:*

***# This is a comment line***

1. *Save the file and exit the text editor. In nano, you can do this by pressing Ctrl + O to save and Ctrl + X to exit.*
2. *The line starting with '#' will be treated as a comment and will not be processed by the system.*

Remember that comments are meant for human readers and are not executed or interpreted by the system. They are useful for providing additional information, explanations, or instructions within a user file.

1. ***From “/etc/passwd “which information will we gather?***

*Ans. The /etc/passwd file contains the username, real name, identification information, and basic account information for each user. Each line in the file contains a database record; the record fields are separated by a colon (:).*

1. ***From “/etc/shadow “which information will we gather?***

*Ans. The /etc/shadow file provides an enhanced authentication mechanism for Linux systems by tightening access at the account level. This text file stores actual passwords in hashed format, along with additional information related to these passwords.*

1. ***From “/etc/group “which information will we gather?***

*Ans. The /etc/group file stores group information and is the default configuration file.*

1. ***From “ /etc/shadow “which information will we gather?***

*Ans. This text file stores actual passwords in hashed format, along with additional information related to these passwords.*

1. ***What is the meaning of + and – in file permission?***

*Ans. In Linux file permissions, the symbols '+' and '-' are used to represent the addition and removal of permission attributes, respectively.*

*Here's a breakdown of the meanings of '+' and '-' in file permissions:*

1. *'+': The '+' symbol is used to add a permission attribute to a file or directory. It indicates that a specific permission is being granted or added to the existing set of permissions. The permission attributes that can be added include read ('r'), write ('w'), and execute ('x') permissions.*

*For example, if you have a file with the following permissions: -rw-r--r--, and you add the execute permission for the owner, it will change to -rwxr--r--.*

1. *'-': The '-' symbol is used to remove a permission attribute from a file or directory. It indicates that a specific permission is being revoked or removed from the existing set of permissions. The permission attributes that can be removed include read ('r'), write ('w'), and execute ('x') permissions.*

*For example, if you have a file with the following permissions: -rwxr--r--, and you remove the write permission for the group, it will change to -r-xr--r--.*

*It's important to note that these symbols are used primarily in the context of command-line tools such as chmod (change mode) for modifying file permissions. The symbols '+' and '-' provide a convenient way to modify permissions without having to specify the entire permission set each time.*

*Additionally, it's worth mentioning that there are other symbols used in file permissions, such as '=' (equal), which sets the permission to a specific value without consideration of the existing permissions.*

1. ***What is “r ““w” “x “in file permission***

*Ans. r stands for read, w for write, and x for execute.*

1. ***What is “4 ““2 ““1” in files permission***

*Ans. Each octal digit is the sum of 4 if read permission is granted, 2 if write permission is granted and 1 if executive permission is granted.*

1. ***What is the use of umask?***

*Ans. The umask command specifies the permissions that the user does not want to be given out to the newly created file or directory.*

1. ***What is default root permission for directory?***

*Ans. As you might remember, the default file permission value is 0644, and the default directory's is 0755.*

# **Assignment Level Intermediate**

1. ***How to assign another new home directory for new users?***

*Ans. There are two methods you can use to change a user's default home directory on a Linux system: using the “usermod” command or editing the “/etc/passwd” file.*

1. ***Command to check group membership of any user***

*Ans. To display who is a member of a group, use the getent command.*

1. ***What happens if I use “ su – “command?***

*Ans. The su command changes user credentials to those of the root user or to the user specified by the Name parameter and initiates a new session.*

1. ***Which command is used to delete any user with its home directory?***

*Ans. The userdel command removes the user account identified by the login parameter.*

1. ***How to add a new user without a home directory?***

*Ans. useradd -M <username> with this command we can add a new user without a home directory.*

1. ***Command to assign account expiry to the user?***

*Ans. Use the Chage command to set the expiry date for an account.*

1. ***Command to add a new group …***

*Ans. To add a new group in Linux, you can use the groupadd command. Here's the syntax:*

***groupadd [options] group name***

*Here, the group name refers to the name you want to assign to the new group. The command may require administrative privileges, so you may need to run it with Sudo or as the root user.*

*Here's an example of how to use the groupadd command:*

**sudo groupadd my group**

*In this example, the command creates a new group named "my group" with the groupadd command. The sudo command is used to execute the command with administrative privileges.*

*After executing the command, the new group will be created on your system, and you can use it for various purposes, such as assigning group permissions to files or adding users to the group.*

1. ***What is default root permission for file?***

*Ans. The default root permission for files in Linux is typically set to read, write, and execute permissions for the root user (root) and no permissions for other users or groups.*

*The permission representation for the root user (root) is as follows:*

*-rwx------ root root filename*

*This means that the owner (root) has read (r), write (w), and execute (x) permissions, while the group and other users have no permissions (---).*

*To break down the permission representation:*

* *The first character - indicates that it is a regular file.*
* *The next three characters rwx represent the permissions for the owner (root). In this case, the owner has read, write, and execute permissions.*
* *The next three characters --- represent the permissions for the group. Since there is no group specified, it indicates that the group has no permissions.*
* *The final three characters --- represent the permissions for other users. In this case, other users also have no permissions.*

*It's important to note that file permissions can be modified using the chmod command to allow or restrict access for different users and groups. The default root permission is set for security reasons to ensure that only the root user has full control over important system files and directories.*

1. ***What is the default umask for root?***

*Ans. The default umask for the root user in Linux typically depends on the distribution and configuration settings. However, a common default umask for the root user is 022.*

*Umask is a permission mask that determines the default permissions assigned to newly created files and directories. It works by subtracting the umask value from the default permissions.*

*In the case of the default umask 022:*

* *For files: The default permissions are 666 (rw-rw-rw-), and when the umask value of 022 is subtracted, the resulting permissions for new files become 644 (rw-r--r--). This means that newly created files have read and write permissions for the owner and read-only permissions for the group and others.*
* *For directories: The default permissions are 777 (rwxrwxrwx), and when the umask value of 022 is subtracted, the resulting permissions for new directories become 755 (rwxr-xr-x). This means that newly created directories have read, write, and execute permissions for the owner, and read and execute permissions for the group and others.*

*Keep in mind that the default umask can be modified in the shell configuration files like ~/.bashrc or /etc/profile to change the default permissions for newly created files and directories, including those created by the root user.*

1. ***What is the default umask for student?***

*Ans. The default umask for a student user in Linux typically depends on the distribution and system configuration. However, a common default umask for regular users, including students, is 002.*

1. ***Which command is used to set user ownership?***

*Ans. The* ***chown*** *command is used to set user ownership (also known as the user owner) of a file or directory in Linux. The syntax for the* ***chown*** *command is as follows:*

***chown [options] user [: group] file(s)***

*Here's a breakdown of the components:*

* *user: Specifies the user to whom you want to assign ownership. It can be the username or user ID.*
* *group (optional): Specifies the group to whom you want to assign ownership. It can be the group name or group ID. If not specified, group ownership remains unchanged.*
* *file(s): Specifies the file(s) or directory to which you want to assign ownership. You can provide multiple file or directory names separated by spaces.*

*Here are a few examples of using the chown command:*

1. *Set ownership of a file to a specific user:*

**chown user1 file.txt**

1. *Set ownership of a directory to a specific user and group:*

***chown user1: group1 directory/***

1. *Set ownership of multiple files and directories to a specific user:*

***chown user1 file1.txt file2.txt directory/***

*Note that the chown command typically requires administrative privileges, so you may need to run it with sudo or as the root user to change ownership of files or directories that you don't own.*

1. ***Which command is used to set group ownership?***

*Ans. The chown command is also used to set group ownership of a file or directory in Linux. To specify the group ownership, you use the colon ':' notation followed by the group name or group ID.*

*Here's an example of using the chown command to set the group ownership:*

***chown :groupname file(s)***

*In the above command, groupname represents the name or group ID of the group to which you want to assign ownership. file(s) refers to the file(s) or directory for which you want to change the group ownership.*

*For instance, to set the group ownership of a file named "file.txt" to a group called "mygroup", you would use the following command:*

***chown :mygroup file.txt***

*This command assigns the group ownership of "file.txt" to "mygroup" while keeping the user ownership unchanged.*

*Similarly, you can use the chown command with the colon ':' notation to change the group ownership of multiple files or directories simultaneously.*

*Remember that, like the user ownership change, the chown command typically requires administrative privileges, so you may need to run it with sudo or as the root user if you don't have ownership over the files or directories you want to modify.*

# **Assignment Level Advance**

1. ***I have one user with the name of KAMAL, Now, I want to add this user in the group name N which command will used?***

*Ans.* To add a user to a specific group in Linux, you can use the **usermod** command. Here's the command you can use to add the user "KAMAL" to the group "N":

sudo usermod -aG N KAMAL

*In this command:*

* ***sudo*** *is used to run the command with administrative privileges.*
* ***usermod*** *is the command for modifying user account properties.*
* ***-aG*** *specifies that we want to add the user to the group(s).*
* ***N*** *is the group name to which you want to add the user.*
* ***KAMAL*** *is the username of the user you want to add to the group.*

*By running this command, the user "KAMAL" will be added to the group "N". The -a option ensures that the user is added to the group without removing them from any other groups they may already be a part of.*

*After executing this command, you may need to log out and log back in for the group membership changes to take effect. You can use the* ***groups*** *command to verify that the user "KAMAL" has been successfully added to the group "N".*

1. ***What is the difference between “ usermod -G “and “ usermod -aG “***

*Ans. The difference between the usermod -G and usermod -aG commands lies in how they handle the modification of group membership for a user.*

1. *usermod -G:*

*The usermod -G command is used to set the primary group membership of a user. When you use this command, it replaces the user's existing group membership with the specified group(s). It removes the user from all other groups and assigns only the group(s) specified.*

*Example:*

***sudo usermod -G newgroup username***

*In this example, the user "username" will have their primary group membership changed to "newgroup". This means they will be removed from all other groups and assigned only to "newgroup". If multiple groups need to be set, you can separate them with commas (e.g., newgroup1, newgroup2).*

1. *usermod -aG:*

*The usermod -aG command is used to add a user to additional group(s) while retaining their existing group membership. It appends the user to the specified group(s) without removing them from any other groups they are currently a member of.*

*Example:*  
***sudo usermod -aG additionalgroup username***

*In this example, the user "username" will be added to the "additionalgroup" without affecting their existing group memberships.*

*To summarize, usermod -G replaces the user's current group membership with the specified group(s), while usermod -aG adds the user to additional group(s) without modifying their existing group memberships.*

1. ***What is the meaning of “ -1 “in password state information?***

*Ans. In Linux, the password state information field (also known as the password status field) in the /etc/shadow file is represented by a number or character that indicates the status of the user's password. The value "-1" in the password state information field represents that the password has been locked.*

*Here's a breakdown of the meanings of password state information values:*

* *"0" or "!" - The user can log in with no password (password is disabled).*
* *"\*" - The user cannot log in with a password (password is locked).*
* *"-1" - The password has been locked (typically used by certain utilities or system administrators to indicate a locked password).*
* *"x" or "!!" - The password is expired and must be changed upon the next login.*
* *A number (greater than 0) - The number of days since the password was last changed, indicating the password's age or time of validity.*

*When the password state information field contains "-1", it signifies that the password for that user has been locked. This means that the user will not be able to log in using a password until the password is unlocked or changed by a system administrator.*

*It's worth noting that the /etc/shadow file stores encrypted password hashes and other account-related information for user accounts on a Linux system. Access to this file is typically restricted to the root user to maintain security.*

1. ***Which command is used to remove the password of any user?***

*Ans. In Linux, the passwd command is used to change a user's password, but it requires the current password to make any modifications. If you want to remove the password of a user without knowing the current password, you will need to have root or administrative privileges. Here's how you can achieve it:*

1. *Open a terminal or command prompt.*
2. *Log in as the root user or use the sudo command to gain administrative privileges.*
3. *Run the following command, replacing <username> with the username of the user whose password you want to remove:*

***sudo passwd -d <username>***

*The -d option is used to delete or remove the password.*

1. *If the command executes successfully, the password for the specified user will be removed.*

*It's important to note that removing the password of a user effectively means anyone can log in as that user without a password. This is generally not recommended for security reasons unless you have a specific need for it.*

1. ***What is the use of “ gpasswd “?***

*Ans. The gpasswd command in Linux is used to manage group-level permissions and settings. It allows an administrator to add or remove users from a group, assign group administrators, and set group passwords. Here are some common uses of the gpasswd command:*

1. *Add a user to a group: To add a user to a group, you can use the following syntax:*

***sudo gpasswd -a <username> <groupname>***

*This command adds the specified <username> to the specified <groupname>.*

1. *Remove a user from a group: To remove a user from a group, you can use the following syntax:*

***sudo gpasswd -d <username> <groupname>***

*This command removes the specified <username> from the specified <groupname>.*

1. *Set or change the group password: Group passwords are used to restrict access to a group. Only users who know the group password can add or remove members from the group. To set or change the group password, you can use the following syntax:*

***sudo gpasswd <groupname>***

*This command prompts you to enter and confirm the new group password.*

1. *Assign a group administrator: A group administrator has the authority to manage the group membership and password. To assign a group administrator, you can use the following syntax:*

***sudo gpasswd -A <username> <groupname>***

*This command designates the specified <username> as the administrator of the specified <groupname>.*

*These are some of the common uses of the gpasswd command. It provides a way to manage group-level permissions and control access to resources within Linux systems.*

1. ***Command to change password policy***

*Ans. The command to change the password policy in Linux depends on the distribution you are using. Here are a few examples of commonly used commands for different distributions:*

1. ***Ubuntu and Debian-based distributions:*** *Ubuntu and Debian-based distributions typically use the pam-auth-update command to configure password policies. You can use the following command to change the password policy:*

***sudo pam-auth-update***

*This command launches an interactive menu that allows you to enable or disable various password-related modules and configure the password policy.*

1. *Red Hat-based distributions (RHEL, CentOS, Fedora): Red Hat-based distributions commonly use the autoconfig command to manage password policies. You can use the following command to change the password policy:*

***sudo authconfig --passalgo=<algorithm> --enablereqlower --enablerequpper --enablereqdigit --enablereqother --minlen=<length> --mindiff=<diffcount> --maxrepeat=<count> --minclass=<classcount> --maxclassrepeat=<count> --maxtries=<count> --faillocktime=<seconds>***

*In this command, you can modify various parameters such as <algorithm> for password hashing algorithm, <length> for minimum password length, <diffcount> for minimum number of character differences, <classcount> for minimum character classes, <count> for maximum character repeats, <seconds> for the duration of account lockout, and more.*

1. *SUSE-based distributions (openSUSE, SLES): SUSE-based distributions typically use the yast command for system configuration. You can use the following command to change the password policy:*

***sudo yast auth***

*This command launches the YaST (Yet another Setup Tool) interface, where you can navigate to the "Authentication" section to configure the password policy.*

*These are just a few examples, and the specific command and method for changing the password policy may vary depending on the Linux distribution and version you are using. It's recommended to refer to the documentation or man pages specific to your distribution for detailed instructions on changing the password policy.*

1. ***What is use of “ sudo “***

*Ans. Sudo is a command-line utility for Unix and Unix-based operating systems such as Linux and macOS. The utility provides an efficient way to temporarily grant users or user groups privileged access to system resources so that they can run commands that they cannot run under their regular accounts.*

1. ***Command to reset virtual machine***

*Ans. VBoxManage controlvm <vm\_name> reset*

1. ***How to change user and group ownership on same time***

*Ans. You can change both the user and group ownership of a file or directory at the same time using the chown command in Linux. The chown command allows you to modify the owner and group of a file or directory with a single command. Here's the syntax:*

***sudo chown <user>:<group> <file\_or\_directory>***

1. ***Command to change user permission on directory***

*Ans. To change user permissions on a directory in Linux, you can use the chmod command. The chmod command is used to modify the permissions of files and directories, allowing you to control which users can read, write, or execute the files or access the directories. Here's the syntax to change user permissions on a directory:*

***sudo chmod <permissions> <directory>***

*Each permission can be assigned a value:*

* *4: Read permission (r)*
* *2: Write permission (w)*
* *1: Execute permission (x)*

1. ***List of special permission in Linux 7.0 is……***

*Ans. As of my last update in September 2021, Linux version 7.0 does not exist. The latest stable Linux kernel at that time was version 5.14. However, I can provide you with a list of special permissions that are available in Linux. These permissions are additional settings that can be applied to files and directories along with the standard read, write, and execute permissions. These special permissions include:*

1. ***SetUID (SUID):*** *When set on an executable file, it allows users who execute the file to temporarily gain the effective UID (user ID) of the file owner, instead of their own UID, while running the program. This is often used for programs that need elevated privileges temporarily to perform specific tasks.*
2. ***SetGID (SGID):*** *Similar to SetUID, but it applies to directories and allows users who create files or directories within that directory to have the group ownership of the parent directory, instead of their default group.*
3. ***Sticky Bit:*** *When set on a directory, it restricts deletion or renaming of files within that directory to only the file owner and the directory owner. This is commonly used on directories like /tmp to prevent users from deleting each other's files.*

*The special permissions are represented by the following characters in the permission string:*

* *SetUID (SUID): Displayed as an 's' in place of the owner's executive permission: rws-r-xr-x or rwS-r--r--.*
* *SetGID (SGID): Displayed as an 's' in place of the group's execute permission: rwxr-sr-x or rw-r-Sr--.*
* *Sticky Bit: Displayed as a 't' in place of the others' execute permission: rwxr-xr-t or rw-r--r-T.*

*Keep in mind that new versions of Linux may have additional features, and the availability of specific features can vary based on the distribution and configuration.*

1. ***What happened if i used this command…? [ #chmod u+s /user/bin/vim]***

*Ans. The command chmod u+s /usr/bin/vim sets the SetUID (SUID) permission on the vim executable binary located at /usr/bin/vim. The SetUID permission allows the user who executes the vim command to temporarily run the program with the effective user ID (UID) of the file's owner (in this case, the owner of the vim binary).*

*What happens if you set the SUID permission on /usr/bin/vim:*

* ***Normal Execution:*** *Without SUID, when a regular user runs vim, the process runs with the user's own privileges, and any file modifications performed by vim will be done as that user. This means vim can only read or modify files that the user has access to.*
* ***With SUID Set:*** *When you set the SUID permission on /usr/bin/vim, the vim process will temporarily run with the privileges of the file owner (typically, this would be the root user since /usr/bin/vim is usually owned by root). This means that vim will have elevated privileges while running, which could potentially allow the user to edit system files or perform other actions that a regular user doesn't have permission to do.*

*Using SUID with a potentially powerful or system-level program like /usr/bin/vim can be risky, as it might open security vulnerabilities if not used carefully. It's essential to understand the implications and carefully assess the need for setting the SUID permission on a binary.*

*In general, it's not recommended to set the SUID permission on arbitrary binaries or scripts unless you have a specific use case and understand the security risks involved. If you are considering setting SUID on a specific program, make sure to thoroughly review the program's security implications and restrict access to it appropriately.*

1. ***What happened if i used this command…. [ #chmod g+s /data]***

*Ans. The command chmod g+s /data sets the SetGID (SGID) permission on the directory named /data. When the SGID permission is set on a directory, any new files or subdirectories created within that directory will inherit the group ownership of the parent directory, rather than the group ownership of the user who created the file or directory.*

*Here's what happens when you set the SGID permission on /data:*

*Normal File Creation: Without SGID, when a user creates a new file or directory within /data, the new file or directory will have the user's default group ownership.*

*With SGID Set: When you set the SGID permission on /data, any new file or directory created within /data will inherit the group ownership of the parent directory /data. This ensures that all files and directories within /data have the same group ownership, even if they are created by different users.*

# **Task: 1**

1. ***Find details about current logged-in user.***

*Ans. Done in lab.*

1. ***Show all processes on terminal***

*Ans. Done in lab.*

1. ***Create primary group***

*Ans. Done in lab.*

1. ***Create supplementary group***

*Ans. Done in lab.*

1. ***Find groups details and list on terminal P6. Find user details and list on terminal.***

*Ans. Done in lab.*

1. ***Use “sudo”***

*Ans. Done in lab.*

1. ***View the last 5 lines of the “/var/log/messages “***

*Ans. Done in lab.*

1. ***Add a new user with name “ NuPuR “***

*Ans. Done in lab.*

## ***Remove this user and user’s home directory***

*Ans. Done in lab.*

1. ***Create new supplementary group name is “ whEEL “***

*Ans. Done in lab.*

1. ***Create a new user with name “ ELviS “***

*Ans. Done in lab.*

1. ***Add / Append a user to a supplementary group***

*Ans. Done in lab.*

1. ***Restrict / Lock login access for “ ELviS “user***

*Ans. Done in lab.*

1. ***Create a new username “ LiNuX without home directory***

*Ans. Done in lab.*

1. ***Create a new username “RedHat “with new home directory “***

*Ans. Done in lab.*

1. ***/etc/HatRed***

*Ans. Done in lab.*

1. ***Create a new user with two (2) days expiry***

*Ans. Done in lab.*

1. ***Remove password for “ ELviS “user***

*Ans. Done in lab.*

1. ***Check user password policy for “ LiNuX “user***

*Ans. Done in lab.*

# Task :2

1. ***Login from “LiNuX” user***

*Ans. Done in lab.*

1. ***Create new directory on desktop name is "FoLdEr”***

*Ans. Done in lab.*

1. ***Change group ownership from LiNuX to root on “FoLdEr” directory***

*Ans. Done in lab.*

1. ***Create new file on /etc/ with name “File”***

*Ans. Done in lab.*

1. ***Check permissions of above file***

*Ans. Done in lab.*

1. ***Login from “student” user***

*Ans. Done in lab.*

1. ***Create new directory on students home with name “file1”***

*Ans. Done in lab.*

1. ***Remove read and write permission for group and other on above file “file1”***

*Ans. Done in lab.*

1. ***Add execute permission for everyone on “file2”***

*Ans. Don in lab.*

1. ***Set Read,write,execute for USER***

*Ans. Done in lab.*

1. ***Set Read and execute for GROUP***

*Ans. Done In lab.*

1. ***Set No permission for other on “Directory1”***

*Ans. Done in lab.*

1. ***Create new group name “ateam” , And add two new users in this group “andy” and “alice”, set password is “password”***

*Ans. Done in lab.*

1. ***Login from root and root home directory Create a new directory in “/home” name is “ateam-text”***

*Ans. Done in lab.*

1. ***Change the group ownership of the ateam-text directory to “ateam”.***

*Ans. Done in lab.*

1. ***Ensure the permission of ateam-text allows group members to create***

*Ans. Done in lab.*