“Київський фаховий коледж зв’язку”

Циклова комісія Комп’ютерної та програмної інженерії

ЗВІТ ПО ВИКОНАННЮ

ЛАБОРАТОРНОЇ РОБОТИ №9

з дисципліни: «Операційні системи»

Тема: “Захист системи та користувачів у Linux. Створення користувачів та груп”

Виконали

студенти

групи РПЗ-03

Команда 6: Sichkar Maxim,

Brytyuk Bohdan

Kyiv 2023

**The goal of the work:**

1. Getting practical skills for working with the Bash command shell.

2. Getting to know the basic actions when creating new users and new user groups.

**Material provision of classes**

1. IBM PC type computer.

2. OS family Windows (Windows 7).

3. Virtual machine - Virtual Box (Oracle).

4. GNU/Linux operating system - CentOS.

5. Cisco network academy site netacad.com and its online Linux courses

**Tasks for preliminary preparation.**

1. Read the short theoretical information for the laboratory work and make a small dictionary

basic English terms for command assignments and their parameters.

2. Study the materials of the online course of the Cisco Academy “NDG Linux Essentials”:

- Chapter 15 - System and User Security

- Chapter 16 - Creating Users and Groups

3. Take the test in the NDG Linux Essentials course on the following topics:

- Chapter 15 Exam

- Chapter 16 Exam

4. On the basis of the considered material, answer the following questions:

**4.1. Explain the concept of UPG, when it is appropriate to use them?**

UPG stands for Unweighted Pair Group Method with Arithmetic Mean. It is a clustering algorithm that is commonly used in data analysis and machine learning to group together similar items based on their features or attributes.

In UPG, the similarity between two groups is measured by the distance between their arithmetic means. The algorithm works by merging the two closest groups into a single new group until all groups are merged into a single cluster.

UPG is appropriate to use when you have a dataset with numerical variables and want to group the items based on their similarity. UPG is particularly useful when the dataset is not too large and when the items have a relatively simple structure.

One advantage of UPG is that it is relatively easy to interpret the resulting clusters because they are based on the arithmetic means of the groups. Another advantage is that UPG is computationally efficient and can handle datasets with many variables.

However, UPG has some limitations. One limitation is that it assumes that the groups are spherical and have the same variance. This may not be true in some datasets, and it can lead to inaccurate clustering results. Additionally, UPG is sensitive to outliers, and it may produce unstable results when the dataset has a high degree of variability or noise.

**4.2. What commands can be used to create user groups? Give examples**

To create a user group on a Unix or Linux-based operating system, you can use the following commands:

groupadd: This command creates a new user group on the system.

Example:

$ sudo groupadd mygroup

This will create a new group named "mygroup" on the system.

groupmod: This command is used to modify the properties of an existing group.

Example:

$ sudo groupmod -n newgroupname oldgroupname

This will change the name of the group "oldgroupname" to "newgroupname".

groupdel: This command is used to delete an existing group from the system.

Example:

$ sudo groupdel mygroup

This will delete the group "mygroup" from the system.

usermod: This command can be used to add a user to an existing group.

Example:

$ sudo usermod -a -G mygroup username

This will add the user "username" to the group "mygroup".

**4.3. What commands can be used to change user group settings? Give examples**

To change user group settings on a Unix or Linux-based operating system, you can use the following commands

usermod: This command is used to modify the properties of an existing user account, including its group membership.

Example:

$ sudo usermod -g newgroupname username

This will change the primary group of the user username to newgroupname.

useradd: This command is used to create a new user account and add it to one or more groups.

Example:

$ sudo useradd -G group1,group2 username

This will create a new user account named username and add it to the group1 and group2 groups.

groups: This command displays the group membership of a user.

Example:

$ groups username

This will display a list of groups that the user username belongs to.

newgrp: This command allows a user to temporarily switch to a different group.

Example:

$ newgrp groupname

This will switch the user's group to groupname until the user logs out or uses the exit command.

5. Prepare the initial version of the report electronically:

- Title page, topic and purpose of the work

- Glossary of terms

- Answers to p. 4.1 and p. 4.5 from tasks for preliminary training

Progress.

1. Initial work in CLI mode in Linux OS of the Linux family:

1.1. Start the VirtualBox virtual machine, select CentOS and run it. Log in

under user: CentOS, password for login: reverse (if you run LR in 401 aud.) and run

terminal.

1.2. Start the Ubuntu\_PC virtual machine (if you are doing the LR tasks through the netacad academy)

1.3. Start your Linux family operating system (if you are working on your own PC and its

installed) and launch the terminal.

2. Work through all the command examples presented in the labs of the NDG Linux Essentials course -

Lab 15: System and User Security and Lab 16: Creating Users and Groups. Create a table to describe these

teams\*\*\*

The name of the command Its purpose and functionality

|  |  |
| --- | --- |
| The name of the command | Its purpose and functionality |
| useradd | The useradd command is a Unix/Linux utility that is used to create new user accounts on the system. When executed, it creates a new user account with a specified username and other settings such as home directory, default shell, and group membership. |
| groupmod | The groupmod command is used to modify the properties of an existing user group on a Unix or Linux-based operating system. It allows an administrator to change the name, GID (Group ID), or other properties of a group. |
| groupadd | The groupadd command is used to create a new group on a Unix or Linux-based operating system. This command is typically used by system administrators to manage user accounts and permissions.  The groupadd command takes a single argument, which is the name of the new group to be created. By default, the new group is created with no members and with a group ID (GID) that is the next available ID number in the system. However, you can also specify a specific GID using the -g option. |
| grep | grep is a powerful command-line utility in Unix, Linux, and other Unix-like operating systems that is used to search for specific text patterns in files and directories. |

\*\*\*Screenshots of the execution of commands in the terminal may not be presented, it is enough to briefly describe the commands

in the table.

3. Perform the following practical tasks in the terminal following actions (show screenshots):

- display information about the current user in different ways (use the hint

id and grep commands);

- practice the last, w and who commands in the terminal. Compare the output of each command,

What details are missing from each team compared to the others?

- create two new user groups - super\_admins, noob\_users and good\_students, define them

identifiers;

- create a new user for each member of your team using the terminal (if

if you work alone, then just three random users), do not forget after creating a new one

user to set a password immediately;

- add new users to the new groups you created so that in the super\_admins and

noob\_users had 2 users, one of which is in both groups, add everyone to the good\_students group

three users;

- view information about groups and which users belong to them, explain what you see;

- delete the first user you created, see if the information about him will remain in

groups where he was;

- delete the second user, see if information about him remains in the groups where he is

was there;

- delete the third user, see if information about him remains in the groups where he is

was there;

- view information about existing user groups;

- delete user groups created by you;

- view information about existing user groups.

**Control questions**

1. **Why is the password not stored explicitly in the configuration files?**

Storing passwords explicitly in configuration files is considered a bad practice from a security perspective because it can make sensitive information easily accessible to unauthorized parties. If an attacker gains access to the configuration file, they would be able to view the password in plaintext and potentially use it to gain unauthorized access to the system.

To mitigate this risk, passwords are typically stored in an encrypted form rather than in plaintext. This can be done using techniques such as hashing or encryption, which convert the password into an unintelligible form that cannot be easily reversed. When a user enters their password, the system will apply the same hashing or encryption algorithm to the entered password and compare the resulting value with the stored value to determine if the password is correct.

In addition to encryption, passwords are often stored in separate files or databases that are only accessible to authorized users or processes. Access to these files or databases may be restricted by permissions or access control mechanisms to limit the risk of unauthorized access.

Overall, storing passwords explicitly in configuration files is considered a security risk and is generally avoided in favor of more secure techniques such as encryption and access control.

1. **Why is it not recommended to perform daily operations using the root account?**

It is not recommended to perform daily operations using the root account for several reasons related to security and safety:

* Accidental damage: The root account has full access to the system, including the ability to modify critical system files and settings. If a mistake is made while logged in as root, it can cause severe damage to the system that may be difficult or impossible to repair. By using a non-root account for daily operations, you can limit the scope of any accidental damage that may occur.
* Increased security risks: The root account is a high-value target for attackers because it has full access to the system. If an attacker gains access to the root account, they can make changes to the system without any restrictions. By using a non-root account, you limit the potential damage that an attacker can cause if they are able to compromise your account.
* Accountability: When using a non-root account, your actions are logged and traceable. This means that in the event of a security incident, it may be possible to determine who was responsible for the actions that led to the incident. When using the root account, it can be difficult to determine who was responsible for any actions taken, which can make it harder to investigate and prevent future incidents.

Overall, it is best practice to use a non-root account for daily operations and only use the root account when necessary to perform system-level tasks that require elevated privileges. By limiting the use of the root account, you can reduce the risk of accidental damage and make it harder for attackers to compromise your system.

1. **What is the difference between the mechanisms for obtaining su and sudo special privileges?**

Both su (short for "switch user") and sudo (short for "superuser do") are commands that allow a user to execute commands with elevated privileges. However, there are some important differences in the way they work.

The su command allows a user to switch to a different user account, typically the root account, by entering the root password. Once the switch is made, all subsequent commands are executed with the privileges of the new user account. The main advantage of su is that it provides complete access to the system, but this can also be a disadvantage because it allows for unrestricted access to all files and commands on the system.

On the other hand, the sudo command allows a user to execute a specific command with elevated privileges, without having to switch to a different user account. By default, sudo requires the user to enter their own password, rather than the root password, to authenticate themselves. This allows for more fine-grained control over which commands can be executed with elevated privileges, and provides an audit trail of who executed each command. Additionally, sudo can be configured to restrict which users are allowed to execute which commands, providing an additional layer of security.

In summary, the main difference between su and sudo is that su allows a user to switch to a different user account and obtain full access to the system, while sudo allows a user to execute a specific command with elevated privileges, using their own password and within a set of defined privileges. sudo provides a more fine-grained control and better audit trail over command execution, while su provides full control over the system.

1. **Why is the home directory of the root user not located in the /home directory?**

The home directory of the root user is typically located at /root instead of the standard /home/<username> directory used for regular user accounts. This is because the root user is a special system account with elevated privileges and should not be subject to the same restrictions and conventions as regular user accounts.

Here are some reasons why the root user's home directory is located at /root:

* Security: By placing the root user's home directory in a different location than regular user accounts, it helps to prevent accidental modification or deletion of the root user's files by regular users. It also makes it more difficult for attackers to locate and tamper with the root user's files.
* Convention: The /home directory is conventionally used to store the home directories of regular users, and it is usually reserved for non-root users. Placing the root user's home directory in a separate location helps to avoid confusion and maintain consistency in the file system layout.
* Permissions: The /root directory is typically set to only be readable and writable by the root user, which helps to prevent unauthorized access to the root user's files. On the other hand, the /home directory is often accessible to other users on the system, which could potentially allow them to view or modify the root user's files.

In summary, placing the home directory of the root user in a different location than regular user accounts helps to maintain security, convention, and permissions on the system.

1. **What is the getent command used for?**

The getent command is a Unix/Linux utility used to retrieve information from various databases configured on a system, including the /etc/passwd, /etc/group, and /etc/hosts files, as well as network services like LDAP or NIS.

The getent command takes one or more database names as arguments, along with the name or ID of the object to retrieve. For example, getent passwd <username> will display the user account information for the specified username, while getent group <groupname> will display the group information for the specified group.

Here are some common uses of the getent command:

Lookup user or group information: You can use getent passwd <username> to retrieve information about a specific user, such as their home directory, shell, and user ID, or getent group <groupname> to retrieve information about a specific group, such as its members and group ID.

Check if a user or group exists: You can use getent passwd <username> or getent group <groupname> to check if a user or group exists on the system. If the command returns a result, the user or group exists.

Lookup hostname information: You can use getent hosts <hostname> to retrieve the IP address associated with a hostname, or getent networks <network> to retrieve the network address for a specific network.

Query network services: If your system is configured to use network services like LDAP or NIS, you can use getent passwd <username> or getent group <groupname> to retrieve user or group information from these services.

In summary, the getent command is a useful tool for retrieving information from various databases and services configured on a Unix/Linux system. It can be used to lookup user or group information, check if a user or group exists, lookup hostname information, or query network services.

1. **How can existing user groups be deleted? Will information about them remain somewhere in system?**

In Unix/Linux systems, you can delete existing user groups using the groupdel command. Here is the syntax:

groupdel <groupname>

Where <groupname> is the name of the group you want to delete.

When you delete a group using groupdel, the group information is removed from the /etc/group file, which is the system file that stores information about all groups on the system. The group's GID (Group ID) is also removed from the /etc/passwd file, which stores information about all users on the system.

Note that deleting a group using groupdel does not remove the files or directories that belong to that group. You may need to manually update the ownership of these files to avoid permission issues.

If a group has been used in ACLs (Access Control Lists) on files or directories, information about the group may remain in these ACLs even after the group is deleted. You can use the getfacl command to view and modify ACLs on files and directories.

In summary, when you delete a user group using the groupdel command, the group information is removed from the system files /etc/group and /etc/passwd, but files or directories that belong to that group are not automatically removed, and information about the group may remain in ACLs on files and directories.

1. **How can you change a user's password?**

In Unix/Linux systems, you can change a user's password using the passwd command. Here is the syntax:

passwd <username>

Where <username> is the name of the user whose password you want to change.

When you run the passwd command, you will be prompted to enter the new password twice for the user. The password will not be displayed on the screen as you type it for security reasons.

Note that in some cases, you may need to have root or sudo privileges to change a user's password.

If you want to force a user to change their password the next time they log in, you can use the -e option with the passwd command. Here is the syntax:

passwd -e <username>

This will set the password's expiration date to the current date, which will force the user to change their password the next time they log in.

In summary, to change a user's password, use the passwd command followed by the username. You will be prompted to enter the new password twice, and in some cases, you may need root or sudo privileges to change the password.

1. **What is the purpose of the chage command?**

The chage command in Unix/Linux systems is used to manage password aging policies for user accounts. The term "aging" refers to the concept of making passwords expire after a certain period of time, which is a security measure designed to ensure that users regularly change their passwords.

The chage command allows system administrators to view and modify password aging policies for user accounts. Here are some common uses of the chage command:

View password aging information: You can use the chage -l <username> command to view the password aging information for a specific user, including the date the password was last changed, the minimum and maximum number of days before the password can be changed again, and the number of days before the password must be changed.

Set password aging information: You can use the chage command with various options to set password aging information for a specific user. For example, chage -m 7 -M 90 -W 14 <username> will set the minimum password age to 7 days, the maximum password age to 90 days, and the warning period to 14 days before the password expires.

Disable password aging: You can use the chage -E -1 <username> command to disable password aging for a specific user, which means that the password will never expire.

In summary, the chage command is used to manage password aging policies for user accounts in Unix/Linux systems. It allows system administrators to view and modify password aging information, set minimum and maximum password ages, and disable password aging altogether.

**9. Which parameters of the usermod command do you consider to be the most used?**

The usermod command in Unix/Linux systems is used to modify user account attributes. Here are some of the most commonly used parameters of the usermod command:

* -a: This parameter is used to add a user to an existing group without removing them from their current groups. For example, usermod -a -G groupname username will add the user to the specified group.
* -d: This parameter is used to change the home directory of a user account. For example, usermod -d /new/home/directory username will change the home directory for the specified user.
* -s: This parameter is used to change the default shell for a user account. For example, usermod -s /bin/bash username will change the default shell to Bash for the specified user.
* -l: This parameter is used to change the username of an existing user account. For example, usermod -l newusername oldusername will change the username from oldusername to newusername.
* -e: This parameter is used to set an expiration date for a user account. For example, usermod -e 2024-12-31 username will set the expiration date for the specified user to December 31, 2024.

In summary, the most commonly used parameters of the usermod command are -a, -d, -s, -l, and -e. These parameters allow system administrators to add users to groups, change home directories and default shells, change usernames, and set expiration dates for user accounts.

**Conclusion:**