Assignment-2 Linux

Format:lab session

Time:90mins

Instruction:

Complete the assignment and also upload the solutions in yourGit repo
Take screenshot of the solution and paste it in word document3.Convert the word doc into pdf before uploading.

1.In Linux FHS (Filesystem Hierarchy Standard) what is the /?

In the Linux Filesystem Hierarchy Standard (FHS), the root directory, represented by a forward slash (/) root directory is the parent directory of all other directories in the system.

2.What is stored in each of the following paths?
/bin, /sbin, /usr/bin and /usr/sbin
/etc
/home
/var
/tmp
/bin: This directory contains executable files (programs) that are essential for the system to boot and operate properly. These programs can be used by both the system administrator and normal users.
/sbin: Similar to /bin, this directory contains executable files that are essential for system administration tasks, such as mounting file systems, configuring network interfaces, and managing system resources. used by the system administrator and not normal users.
/usr/bin: This directory contains non-essential command-line user programs that are installed by the system administrator or the user.
/usr/sbin: Similar to /usr/bin, this directory contains non-essential system administration programs that are not required for basic system operation but may be useful for the system administrator.
/etc: This directory contains system configuration files
/home: This directory contains user home directories, which are created automatically when a new user account is added to the system.
/var: This directory contains variable data files
/tmp: This directory contains temporary files that are created and deleted automatically by the system and by applications.

3.What is special about the /tmp directory when compared to other directories?
The /tmp directory is used to store temporary files
These files are typically short-lived and are not intended to be stored for a long time.
The /tmp directory is accessible to all users on the system, regardless of their user permissions or privileges
Files stored in the /tmp directory are typically deleted automatically when the system is rebooted

4.What kind of information one can find in /proc?
The /proc directory in a Linux file system contains a virtual file system that provides information about the current state of the system, processes, and hardware. This directory does not contain actual files, but instead contains virtual files that are created on the fly and provide real-time information about various aspects of the system

5.
What makes /proc different from other filesystems?
the /proc file system is different from other file systems in that it is a virtual file system that contains system and process information, its contents are constantly changing, and its files are not real files. It is a valuable resource for system administrators and developers who need to

monitor and manage system and process information.

6.
True or False? only root can create files in /proc--□ False

7.What can be found in /proc/cmdline?
The /proc/cmdline file contains the kernel command line parameters that were used to boot the Linux system

8.In which path can you find the system devices (e.g. block storage)?

In Linux, system devices are typically located in the /dev directory. This directory contains special files that represent various devices attached to the system, including block storage devices such as hard drives, solid-state drives, and USB drives.

Permissions

9. How to change the permissions of a file?
You can change the permissions of a file in Linux using the chmod command. The chmod command allows you to modify the read, write, and execute permissions for the owner, group, and other users. Here's the basic syntax for the chmod command
chmod [permissions] [filename]

10. What does the following permissions mean?:
777
644
750

In Linux, file permissions are represented by a three-digit number that consists of three separate digits. Each digit represents a different set of permissions: the first digit represents the permissions for the owner, the second digit represents the permissions for the group, and the third digit represents the permissions for other users.

Here's what each digit represents:

7 = read, write, and execute permissions (4+2+1)
6 = read and write permissions (4+2)
5 = read and execute permissions (4+1)
4 = read-only permissions
3 = write and execute permissions (2+1)
2 = write-only permissions
1 = execute-only permissions
So, to answer the question:

777 means that the owner, group, and other users have read, write, and execute permissions on the file or directory.
644 means that the owner has read and write permissions, while the group and other users have read-only permissions on the file.
750 means that the owner has read, write, and execute permissions, the group has read and execute permissions, and other users have no permissions on the file or directory.

11.What this command does? chmod +x some_file
The command "chmod +x some_file" sets the executable permission (+x) for the user (owner) on the file "some_file". The "+x" indicates that the executable permission is being added to the file's existing permissions

12. Explain what is setgid and setuid
Setgid is a permission that can be set on a file or directory to enable group ownership inheritance.
Setuid is a permission that can be set on a file to enable the file to be executed with the permissions of the file owner instead of the permissions of the user executing the file.

13. What is the purpose of sticky bit?
In Linux, the sticky bit is a special permission that can be set on directories to prevent users from deleting files that they do not own within that directory. When the sticky bit is set on a directory, only the owner of a file within that directory or the root user can delete or rename the file.
sticky bit does not restrict users from modifying the contents of a file within a directory, only from deleting or renaming the file itself.

14. What the following commands do?
chmod
chown
Chgrp

chmod:"chmod u+x file.txt"
The chmod command is used to change the permissions of a file or directory. It allows the owner of a file or directory to set the read, write, and execute permissions for themselves, their group, and others.

chown:"chown user1 file.txt"
The chown command is used to change the owner of a file or directory. It allows the current owner of a file or directory to transfer ownership to another user or group.

chgrp: "chgrp group1 file.txt"
The chgrp command is used to change the group ownership of a file or directory. It allows the current owner of a file or directory to transfer group ownership to another group.

15. What is sudo? How do you set it up?
sudo (short for "superuser do") is a command-line utility that allows users with the appropriate permissions to execute commands as another user, usually the root user, on a Unix-based system. It is commonly used to perform administrative tasks that require elevated privileges.

To set up sudo, follow these steps:

1)Log in as the root user or a user with sudo privileges.

2)Install the sudo package if it is not already installed on the system. This can be done using the package manager of the distribution. For example, on Ubuntu, run the command "apt-get install sudo" to install the package.

3)Add the user who will be allowed to use sudo to the sudoers file. This can be done using the "visudo" command, which opens the sudoers file in a text editor. The file is typically located at /etc/sudoers. Add a line to the file in the following format:

4)username ALL=(ALL) ALL

5)Replace "username" with the username of the user who will be allowed to use sudo. This line allows the user to run any command with elevated privileges.

6)Save and close the sudoers file.

Once sudo is set up, the user can use it by running the "sudo"

16. True or False? In order to install packages on the system one must be the root user or use the sudo command

True. By default, regular users on a Linux system do not have permission to install or modify system-wide packages or files. These tasks require elevated privileges and can only be performed by the root user or a user with sudo privileges

17. Explain what are ACLs. For what use cases would you recommend to use them?
ACLs (Access Control Lists) are a set of permissions that are used to grant or deny access to files or directories in a Linux or Unix-based system. Unlike traditional Unix permissions which only allow for three levels of access (owner, group, and others)

ACLs can be used in a variety of use cases, some of which include:

1)Granting access to specific users or groups: With ACLs, it is possible to grant access to specific users or groups, rather than just to the owner, group, or others.

2)Granting different levels of access to different users or groups: With ACLs, it is possible to grant different levels of access to different users or groups, such as read-only access or full access.

3)Providing additional security for sensitive files or directories: ACLs can be used to provide additional security for sensitive files or directories, by restricting access to only authorized users or groups.

4)Enforcing compliance with regulatory or legal requirements: ACLs can be used to enforce compliance with regulatory or legal requirements, by restricting access to files or directories based on specific criteria such as job function or clearance level.

18. You try to create a file but it fails. Name at least three different reason as to why it could happen

Insufficient permissions
File or directory already exists
Disk space or quota limits: If the file system is full or the user has exceeded their disk quota, creating a new file may fail.

19. A user accidentally executed the following chmod -x $(which chmod).
How to fix it?

The command chmod -x $(which chmod) removes the execute permission for the chmod command, which means that the user will no longer be able to use chmod to modify file permissions.
In order to fix this, the following steps can be taken:

1.Log in as the root user or a user with sudo privileges.
2.Use the which command to determine the path to the chmod command: which chmod
3.Use the chmod command to restore the execute permission for the chmod command: chmod +x /path/to/chmod
4.Verify that the execute permission has been restored by running ls -l /path/to/chmod. The output should show that the execute permission has been restored (e.g. -rwxr-xr-x).

Once the execute permission has been restored for the chmod command, the user will be able to use it to modify file permissions again

Scenarios

20. You would like to copy a file to a remote Linux host. How would you do?

SCP: SCP (secure copy) is a command-line utility that allows you to securely transfer files between hosts using SSH.
"scp /path/to/local/file username@remotehost:/path/to/remote/directory/"

/path/to/local/file with the path to the file you want to copy, username with the username for the remote host, remotehost with the hostname or IP address of the remote host, and /path/to/remote/directory/ with the destination path on the remote host where you want to copy the file.

SFTP: SFTP (secure file transfer protocol) is a file transfer protocol that uses SSH for secure file transfer
"sftp username@remotehost"

rsync: Rsync is a command-line utility that allows you to synchronize files and directories between hosts.
"rsync /path/to/local/file username@remotehost:/path/to/remote/directory/"

/path/to/local/file with the path to the file you want to copy, username with the username for the remote host, remotehost with the hostname or IP address of the remote host, and /path/to/remote/directory/ with the destination path on the remote host where you want to copy the file. Rsync will only transfer the parts of the file that have changed

21. How to generate a random string?
In Linux, you can generate a random string using the openssl command with the rand function
"openssl rand -hex 16"
This command will generate a random string of 16 hexadecimal characters.

22. How to generate a random string of 7 characters?
"pwgen -1 -A 7"---->To generate a random string of 7 uppercase letters using pwgen, you can use the -A option like this:

"pwgen -1 -a 7"---->This will generate a random string of 7 characters using both uppercase and lowercase letters.

Systemd

23. What is systemd?
Systemd is a system and service manager for Linux operating system. It is responsible for managing the system's startup and shutdown processes, as well as managing system services and daemons.

24. How to start or stop a service?
start service--->"sudo systemctl start service-name"
stop service---->"sudo systemctl stop service-name"

25. How to check the status of a service?
restart---->sudo systemctl status service-name

26. On a system which uses systemd, how would you display the logs?
To display all system logs:
sudo journalctl
To display logs for a specific unit, such as a service or a target:
sudo journalctl -u unit-name
sudo journalctl -o verbose

27. Describe how to make a certain process/app a service

To make a certain process or application a service in Linux, you can create a systemd unit file

1.Create a new file with a ".service" extension in the "/etc/systemd/system/" directory.

```
sudo nano /etc/systemd/system/myservice.service
```

2.Add the following lines to the file, replacing "myuser" with the username of the user who will run the service, and "/path/to/myscript.py" with the path to the script you want to run:

```
[Unit]
Description=My Service
After=multi-user.target

[Service]
User=myuser
ExecStart=/usr/bin/python3 /path/to/myscript.py
Restart=always

[Install]
WantedBy=multi-user.target
```

3.Save the file and exit the editor.

4.Reload the systemd configuration to read the new unit file:

```
sudo systemctl daemon-reload
```

5.Start the service using its name:

```
sudo systemctl start myservice
```

28. Troubleshooting and Debugging

Troubleshooting and debugging are important skills for any system administrator or developer working with Linux systems.

Check system logs,Use debugging tools,Check resource usage,Check configuration files,Use command-line options,Research online resources

29. Where system logs are located?

System logs in Linux are typically located in the "/var/log/" directory

To display logs for a specific time range:

sudo journalctl --since "yyyy-mm-dd hh:mm:ss" --until "yyyy-mm-dd hh:mm:ss"

To display logs with additional information, such as the process ID, message IDs, and source file name:

30. How to follow file's content as it being appended without opening the file every time?

To follow a file's content as it is being appended without opening the file every time, you can use the "tail" command with the "-f" option. The "tail" command displays the last few lines of a file, and the "-f" option tells "tail" to continuously monitor the file for changes and display any new lines that are appended to the file.

```
tail -f /var/log/syslog
```

31. What are you using for troubleshooting and debugging network issues?

1.Ping: The ping command can be used to test connectivity between two network devices by sending ICMP packets and measuring the response time.

2.Traceroute: The traceroute command can be used to identify the path that network packets take between two devices by sending packets with increasing TTL values and observing the route.

3.Netstat: The netstat command can be used to display information about active network connections, open ports, and network statistics.

4.tcpdump: The tcpdump command can be used to capture and analyze network traffic in real-time, which can help identify issues with network protocols and applications.

5.Wireshark: Wireshark is a network protocol analyzer that can be used to capture, analyze, and troubleshoot network traffic.

6.Log files: Network-related logs can provide valuable information about network events and issues. Logs can be found in various locations, such as "/var/log/messages" and "/var/log/syslog".

7.Network monitoring tools: Network monitoring tools, such as Nagios, Zabbix, and Cacti, can be used to monitor network devices and services for issues and generate alerts when problems are detected.

These tools and techniques are just a few examples of what can be used to troubleshoot and debug network issues. Successful network troubleshooting and debugging often requires a systematic approach and a combination of tools and techniques to identify and resolve problems.

32. What are you using for troubleshooting and debugging disk & file system issues?

1.fsck: The fsck command can be used to check and repair file system errors, including bad blocks, directory issues, and file system corruption.

2.df: The df command can be used to display information about disk space usage, including available space, used space, and file system types.

3.du: The du command can be used to display disk space usage for specific directories and files.

4.mount: The mount command can be used to mount and unmount file systems, including network file systems.

5.lsblk: The lsblk command can be used to display information about block devices, including disk partitions and their mount points.

6.dd: The dd command can be used to copy and convert data between files and devices, including creating disk images and testing disk performance.

7.Log files: Disk and file system-related logs can provide valuable information about disk errors and file system issues. Logs can be found in various locations, such as "/var/log/messages" and "/var/log/syslog".

8.SMART: Self-Monitoring, Analysis, and Reporting Technology (SMART) is a disk monitoring system that can be used to detect and predict disk failures.

These tools and techniques are just a few examples of what can be used to troubleshoot and debug disk and file system issues.

33. What are you using for troubleshooting and debugging process issues?

1.ps: The ps command can be used to display information about running processes, including process IDs, CPU and memory usage, and the command used to launch the process.

2.top: The top command can be used to display real-time information about system processes and resource usage, including CPU and memory usage.

3.kill: The kill command can be used to terminate a process by sending a signal to the process.

4.strace: The strace command can be used to trace system calls and signals made by a process, which can help identify issues with the process.

5.lsof: The lsof command can be used to display information about files and network sockets opened by a process.

6.Log files: Process-related logs can provide valuable information about process events and issues. Logs can be found in various locations, such as "/var/log/messages" and "/var/log/syslog".

7.Debuggers: Debuggers, such as GDB and LLDB, can be used to trace and analyze program execution and identify issues with a process.

These tools and techniques are just a few examples of what can be used to troubleshoot and debug process issues.

34. What are you using for debugging CPU related issues?

1.top: The top command can be used to display real-time information about system processes and resource usage, including CPU usage.

2.sar: The sar command can be used to collect and report system resource usage, including CPU usage, over time.

3.vmstat: The vmstat command can be used to display information about virtual memory and system performance, including CPU usage.

4.perf: The perf command can be used to analyze performance statistics, including CPU usage, for individual processes and the system as a whole.

5.strace: The strace command can be used to trace system calls and signals made by a process, which can help identify issues with the process and its CPU usage.

6.Log files: System logs, such as the kernel log and system message log, can provide valuable information about CPU-related issues, including hardware errors and kernel panics.

7.Monitoring tools: CPU monitoring tools, such as Zabbix, Nagios, and Munin, can be used to monitor system resource usage, including CPU usage, and alert administrators to potential issues.

These tools and techniques are just a few examples of what can be used to debug CPU-related issues.

35. You get a call from someone claiming "my system is SLOW". What do you do?

If someone contacts me claiming that their system is slow, here are the steps I would take to diagnose and address the issue:

Gather more information: I would ask the user for more specific details about the issue, such as when the system started to slow down, which programs or tasks are running when the system slows down, and whether any error messages or warnings appear.

Check system resource usage: I would check the system resource usage, including CPU, memory, and disk usage, to identify if any resources are being maxed out or if there are any abnormal spikes.

Check system logs: I would check the system logs for any error messages, warnings, or system events that could be related to the slow performance.

Check for malware or viruses: I would check the system for any signs of malware or viruses that could be affecting performance.

Check for system updates: I would check for any available system updates, including security updates and performance improvements, that could be installed to address the issue.

Check for hardware issues: I would check the system hardware for any issues that could be affecting performance, such as a failing hard drive or insufficient RAM.

Identify and address software issues: I would check the software running on the system for any issues, such as a program or service that is consuming too many resources, and address them accordingly.

Provide guidance and advice: Depending on the cause of the slow performance, I would provide guidance and advice to the user on how to optimize their system, such as disabling unnecessary startup programs, removing unnecessary files, and upgrading hardware if necessary.

36. Explain iostat output

The iostat command is a system monitoring tool that can be used to display I/O (input/output) statistics for block devices such as hard drives and SSDs.

The iostat command is a system monitoring tool that can be used to display I/O (input/output) statistics for block devices such as hard drives and SSDs. Here is an example output of the iostat command:

Device:	tps	kB_read/s	kB_wrtn/s	kB_dscd/s	kB_read	kB_wrtn
sda	9.82	299.31	106.16	0.00	802687784	284790204
sdb	0.00	0.00	0.00	0.00	210616	952
sdc	0.02	0.32	0.56	0.00	855404	

The output is divided into several columns that represent different aspects of I/O performance:

Device: The name of the block device being monitored.

tps: The number of I/O transactions per second that are being performed on the device.

kB_read/s: The amount of data that is being read from the device per second, measured in kilobytes per second.

kB_wrtn/s: The amount of data that is being written to the device per second, measured in kilobytes per second.

kB_dscd/s: The amount of data that is being discarded (or thrown away) per second, measured in kilobytes per second.

kB_read: The total amount of data that has been read from the device since the system was started, measured in kilobytes.

kB_wrtn: The total amount of data that has been written to the device since the system was started, measured in kilobytes.

37. How to debug binaries?

Debugging binaries can be done using a number of tools and techniques, depending on the system and programming language being used

Debugging symbols
Core dumps
Dynamic tracing
Reverse engineering
Fuzz testing

38. What is the difference between CPU load and utilization?

CPU load refers to the amount of processing work that the CPU is currently handling, relative to its capacity. It is typically measured as an average over a period of time, such as the past minute or 5 minutes. A high CPU load indicates that the CPU is working hard and potentially may not be able to keep up with the amount of work that is being requested of it. The CPU load is often represented as a percentage, with values greater than 100% indicating that the CPU is overloaded.

CPU utilization, on the other hand, refers to the percentage of time that the CPU is actively working, as opposed to being idle. It is typically measured as an instantaneous value, representing the current state of the CPU. High CPU utilization indicates that the CPU is being fully utilized and is actively processing work, while low utilization indicates that the CPU is idle and not doing any work.

CPU load represents the amount of work being performed relative to the CPU's capacity, while CPU utilization represents the percentage of time that the CPU is actively working. Both measures can be useful for monitoring system performance and identifying potential performance issues.

39. How you measure time execution of a program?

Time command,Clock function,Profiling tools,Manual instrumentation

the best method for measuring the execution time of a program will depend on the specific system and programming language being used, as well as the level of detail and accuracy required.

Scenarios

40.You have a process writing to a file. You don't know which process exactly, you just know the path of the file. You would like to kill the process as it's no longer needed. How would you achieve it?

One way to achieve this would be to use the fuser command to identify the process ID (PID) of the process that is currently accessing the file, and then use the kill command to terminate that process.

```
fuser -v /path/to/file.txt
kill <PID>
```

Kernel

40.You have a process writing to a file. You don't know which process exactly, you just know the path of the file. You would like to kill the process as it's no longer needed. How would you achieve it?

One way to achieve this would be to use the fuser command to identify the process ID (PID) of the process that is currently accessing the file, and then use the kill command to terminate that process.

```
fuser -v /path/to/file.txt
kill <PID>
```

41. What is a kernel, and what does it do?

The kernel is a central component of an operating system (OS) that manages system resources and provides an interface between software applications and hardware components. It is responsible for controlling low-level system functions, such as device drivers, memory management, process management, and input/output operations.

The kernel acts as a bridge between the user-level applications and the hardware resources, and it provides a layer of abstraction that allows the applications to interact with the system hardware in a consistent and predictable way. It manages the resources of the system, including the processor, memory, and input/output devices, and ensures that each application has access to the resources it needs to function properly.

In addition, the kernel also provides security features such as access control and privilege separation to

protect the system from malicious attacks and prevent unauthorized access to sensitive data.

Overall, the kernel is a critical component of any operating system, and it plays a crucial role in ensuring that the system runs smoothly, efficiently, and securely.

42. How do you find out which Kernel version your system is using?

uname -r

43. What is a Linux kernel module and how do you load a new module?

A Linux kernel module is a piece of code that can be dynamically loaded and unloaded into the running kernel to extend its functionality or add device drivers. Kernel modules are used to add new features or drivers to the kernel without having to recompile the entire kernel or reboot the system.

To load a new kernel module, you can use the modprobe command followed by the name of the module-->sudo modprobe nvidia

To unload a kernel module, you can use the rmmod command followed by the name of the module.--->sudo rmmod nvidia

44. Explain user space vs. kernel space

User space refers to the memory and resources that are used by user-level processes, such as applications and services. User space processes run in a restricted environment and can only access a limited set of system resources, including the CPU, memory, and I/O devices, through system calls that are managed by the kernel.

kernel space is the part of the operating system where the kernel code and data reside. The kernel has unrestricted access to all system resources, including the CPU, memory, and I/O devices. The kernel provides system services and manages system resources, such as device drivers, process management, memory management, and input/output operations. The kernel also provides system calls that allow user-level processes to request services and access system resources.

In general, user space and kernel space are kept separate to ensure the security, stability, and performance of the system. By restricting user-level processes to a limited set of resources and managing access to system resources through system calls, the kernel can ensure that user-level processes do not interfere with each other or with the system as a whole.

45. In what phases of kernel lifecycle, can you change its configuration?

During the initial kernel configuration and compilation

After the kernel has been installed

During the boot process

46. Where can you find kernel's configuration?

The kernel configuration is stored in a file called .config, which is located in the root directory of the kernel source code.

If you are using a Linux distribution that provides precompiled kernels, you can usually find the kernel configuration file in the /boot directory. The file is named after the kernel version, with the suffix .config.

47. Where can you find the file that contains the command passed to the boot loader to run the kernel?

The file that contains the command passed to the boot loader to run the kernel is usually the grub.cfg file.

The grub.cfg file is located in the /boot/grub directory on most Linux distributions.

To view the contents of the grub.cfg file, you can use a text editor or the cat command.

48. How to list kernel's runtime parameters?

You can list the kernel's runtime parameters by viewing the contents of the /proc/cmdline file. This file contains the command line parameters that were passed to the kernel at boot time.

cat /proc/cmdline

49. Will running sysctl -a as a regular user vs. root, produce different result?

Yes, running sysctl -a as a regular user vs. root will produce different results.

When run as a regular user, the sysctl command will only display the values of the kernel parameters that the user has permission to view.

When run as the root user, the sysctl -a command will display the values of all kernel parameters, including those that are restricted to regular users.

50. You would like to enable IPv4 forwarding in the kernel, how would you do it?

To enable IPv4 forwarding in the kernel, you can use the sysctl command to modify the net.ipv4.ip_forward parameter:

1.Open a terminal window and log in as the root user or a user with sudo privileges.

2.Edit the /etc/sysctl.conf file using a text editor such as nano or vi:

sudo nano /etc/sysctl.conf

3.Add the following line to the end of the file:

net.ipv4.ip_forward=1

This sets the net.ipv4.ip_forward parameter to 1, which enables IPv4 forwarding in the kernel.

4.Save the changes and exit the text editor.

5.Apply the changes to the kernel by running the following command:

sudo sysctl -p

This command reloads the kernel parameters from the sysctl.conf file and applies the changes.

51. How sysctl applies the changes to kernel's runtime parameters the moment you run sysctl command?

When you run the sysctl command to modify a kernel parameter, the changes take effect immediately. This is because the sysctl command communicates directly with the kernel's proc filesystem, which is a virtual filesystem that provides a way for the kernel to expose system information and configuration parameters to user space.

The sysctl command writes the new value of the kernel parameter to the appropriate file in the proc filesystem. For example, if you run the command sudo sysctl net.ipv4.ip_forward=1, the command writes the value 1 to the file /proc/sys/net/ipv4/ip_forward.

The kernel watches the appropriate file in the proc filesystem and applies the new value of the parameter as soon as it detects a change to the file. This means that the change takes effect immediately without the need to reboot the system or restart any processes.

the sysctl command are not permanent and will be lost when the system is rebooted. To make changes permanent, you should add the appropriate sysctl command to a configuration file such as /etc/sysctl.conf so that it is applied automatically during boot.

52. How changes to kernel runtime parameters persist? (applied even after reboot to the system for example)

Changes to kernel runtime parameters made using the sysctl command are not persistent and will be lost when the system is rebooted. In order to make changes to kernel parameters persistent, you need to add them to a configuration file that is read during system startup.

53. Are the changes you make to kernel parameters in a container, affects also the kernel parameters of the host on which the container runs?

No, the changes you make to kernel parameters inside a container are isolated and do not affect the kernel parameters of the host system. Containers use kernel namespaces to create an isolated environment that has its own view of the system resources, including the kernel parameters.

When you modify a kernel parameter inside a container, you are actually modifying the kernel parameters for the container's own namespace, not the host's. Therefore, the changes will only affect processes running inside that container, and not any processes running outside the container or on the host system.

SSH

54. What is SSH? How to check if a Linux server is running SSH?

SSH (Secure Shell) is a network protocol that allows secure remote access to a server or device over an unsecured network, such as the internet. It provides a secure way to log in, transfer files, and execute remote commands on the server.

Open a terminal on your local machine and log in to the server using a user account that has administrative privileges.
command to check if the SSH service is running:

```
systemctl status ssh
```

55. Why SSH is considered better than telnet?

SSH (Secure Shell) is considered better than Telnet because of the following reasons:

Security: Telnet sends all data including passwords and sensitive information in plaintext over the network, which makes it easy for anyone to intercept and read the data. In contrast, SSH encrypts all data, including passwords and sensitive information, making it more secure and less vulnerable to eavesdropping and hacking.

Authentication: Telnet does not provide any authentication mechanisms, making it easy for anyone to connect to the server and access its resources. In contrast, SSH provides several authentication mechanisms, such as password authentication, public key authentication, and two-factor authentication, making it more secure and harder to breach.

Remote Command Execution: Telnet only provides remote access to a shell prompt, while SSH allows users to execute remote commands securely, making it a more versatile tool for system administration and remote access.

Portability: SSH is available on most modern operating systems, including Linux, macOS, and Windows, making it a more versatile and widely used tool than Telnet.

56. What is stored in ~/.ssh/known_hosts?

The ~/.ssh/known_hosts file contains a list of public host keys for servers that the user has previously connected to using SSH. Each line of the file represents a single server and includes the server's hostname or IP address, the type of encryption key used by the server, and the server's public key fingerprint.

57. You try to ssh to a server and you get "Host key verification failed".
What does it mean?

When you see the message "Host key verification failed" while trying to SSH to a server, it means that the SSH client could not verify the authenticity of the server you are trying to connect to. This can happen when the server's public key is not present in your ~/.ssh/known_hosts file or when the key in the known_hosts file does not match the one presented by the server.

This can happen for several reasons, such as the server's IP address has changed, or the server's SSH keys have been updated. In some cases, it could also be a sign of a man-in-the-middle attack, where an attacker is intercepting the connection and posing as the server.

58. What is the difference between SSH and SSL?

SSH (Secure Shell) is primarily used for secure remote access and management of systems. It provides a secure, encrypted communication channel between a client and a server, allowing users to remotely access and manage a system over an unsecured network.

SSL (Secure Sockets Layer), now known as TLS (Transport Layer Security), is used primarily to secure web communications. It provides a secure, encrypted communication channel between a web server and a client, allowing secure transmission of sensitive information such as login credentials, credit card

information, and other personal information.

SSH is used for secure remote access and management of systems, while SSL/TLS is used for securing web communications.

59. What ssh-keygen is used for?

"ssh-keygen" is a tool used for generating public and private key pairs for SSH authentication. It is used to create a pair of keys for a user, one of which is the private key that is kept on the user's local machine, and the other is the public key that is copied to the remote server. The public key is added to the authorized keys list on the remote server, allowing the user to log in securely without needing to enter a password.

"ssh-keygen" can also be used to generate host keys, which are used by SSH to identify remote hosts and ensure that the communication is secure and not intercepted by a third party.

60. What is SSH port forwarding?

SSH port forwarding, also known as SSH tunneling, is a technique used to create a secure encrypted connection between two computers over an insecure network. It allows a user to securely connect to a remote server and forward traffic from a specific port on the local machine to a port on the remote server, or vice versa. This can be useful for accessing remote resources that are not directly accessible from the local machine, such as a database or a web server.