"Kyiv Professional College of Communication"

Cycle Commission of Computer Engineering

REPORT ON THE IMPLEMENTATION

LABORATORY WORK №8

in the discipline: "Operating systems"

Topic: "Saving system service data and network configuration"

Performed by

student

of BICS-13 group

Zlata Volodymyrivna Los

Checked by the teacher

Sushanova V.S.

Kyiv 2024

**Objectives:**

1. Gaining practical skills in working with the Bash shell.

2. Familiarity with the basic structures for storing system data - processes, memory, log files and kernel status messages.

3. Familiarity with the FHS standard.

4. Familiarization with the actions when setting up a network.

**Material support of classes:**

1. Computer such as IBM PC.

2. Windows operating system and Virtual Box (Oracle) virtual machine.

3. GNU/Linux OS (any distribution).

4. The Cisco Network Academy website netacad.com and its online Linux courses.

**Tasks for preliminary preparation:**

1. \*Read the brief theoretical information for the laboratory work and make a small dictionary of basic English terms on the purpose of commands and their parameters.

A network is a system that consists of interconnected elements or nodes that can exchange data or resources. In the context of computer technology, a network is defined as an intercommunication structure that allows computers and other devices to connect to each other to exchange information, resources, or services.

Internet is a worldwide system of interconnected computer networks based on a set of Internet protocols.

Client - a computer device or software that accesses another device or service (server) to obtain a particular service or resource.

Router - A network device that determines the shortest path for sending data between different networks.

Packet: A small block of data that is sent over a network. Packets contain information about their destination, source, and contents.

IP Address: A unique identifier assigned to a device on a network. An IP address is used to localize and identify devices on a network.

Mask: A bit mask used to determine which bits in an IP address are network-related and which are host-related.

Hostname: A unique name that is assigned to a device on the network. A hostname is usually used to replace or supplement an IP address.

URL - The address of a resource on the Internet. A URL includes the protocol, domain name, path to the resource, and other information needed to access a web page or other resource.

DHCP (Dynamic Host Configuration Protocol) is a standard application layer protocol that allows computers to automatically obtain an IP address and other settings necessary for networking.

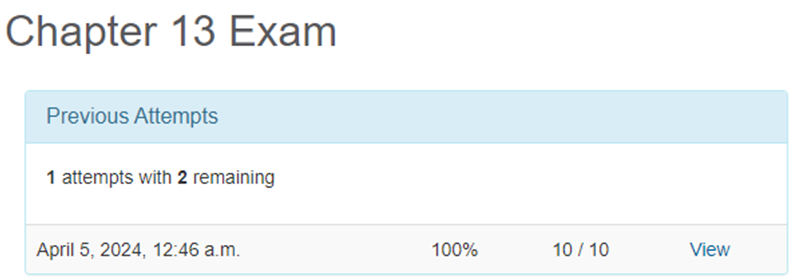
2. Take the NDG Linux Essentials online course from Cisco Academy:

- Chapter 13 - Where Data is Stored

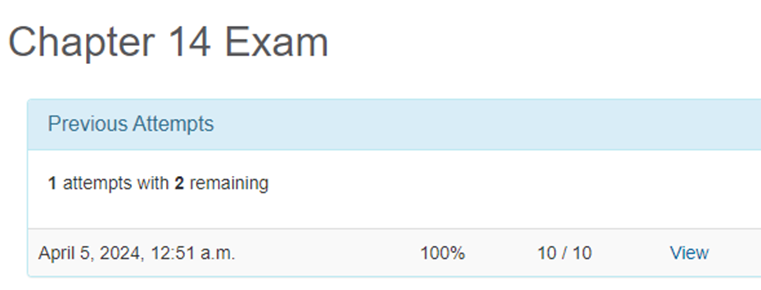
- Chapter 14 - Network Configuration

3. Take the NDG Linux Essentials quizzes on the following topics:

- Chapter 13 Exam



- Chapter 14 Exam



4. Based on the material reviewed, answer the following questions:

4.1 Explain the concept of "pseudo file system", why does the system need it?

**A pseudo-file system** is an abstract layer of software that provides an interface for accessing various system resources and information in the form of files or directories. It is not a real file system, but represents system resources, such as processes, devices, network connections, regulators, system information, and more, as a file hierarchy.

Pseudo-file systems are used in operating systems to provide a single point of access to various system resources through a standard file system interface. This makes working with system resources more convenient and consistent for software developers.

4.2 Why do users rarely access the /proc directory directly, and how can information be obtained from it?

Users rarely access the /proc directory directly on Unix and Linux systems for several reasons:

- **Interface complexity:** Files and directories in /proc do not have conventional names and extensions that are easily understood by the user. They are presented as numeric identifiers, sometimes without corresponding explanatory names. This can make working with /proc less convenient for the average user.

- **System-specific information:** The information contained in /proc is often specific to a particular operating system or even a particular kernel. This means that it can be difficult for users to interpret this information or use it without the appropriate knowledge of the system.

4.3 \*What is the purpose of /proc/cmdline, /proc/meminfo, and /proc/modules?

**/proc/cmdline:**

- Purpose: This file contains the command line arguments that were passed to the Linux kernel at boot time.

- Usage: useful for obtaining information about the parameters used to boot the kernel, such as boot parameters for the bootloader, kernel and its modules. This can be useful for analyzing the parameters used to solve boot problems or to customize the operating system.

**/proc/meminfo:**

- Purpose: This file contains information about the system's memory usage.

- Usage: Provides information about the total amount of memory available, free, and in use, as well as information about the use of swap space, buffers, cache, and other memory parameters. This information is useful for monitoring and analyzing the system's memory usage.

**/proc/modules:**

- Purpose: This file contains a list of loaded Linux kernel modules.

- Usage: Provides information about the current kernel modules that have been loaded on the system. This is useful for checking the presence and versions of loaded kernel modules, which can be important for diagnosing hardware problems or for analyzing system configuration.

4.4 \*What is the purpose of the free command?

The free command is used to display information about memory usage in the system. It provides statistics about the amount of physical and virtual memory, its utilization and availability.

The free command displays the following information:

**- Total:** The total amount of physical memory in the system.

**- Used:** The amount of physical memory used.

**- Free:** The amount of free physical memory that is available for use.

**- Shared:** The amount of memory that is shared with other processes.

**- Buffers:** The amount of memory used by the kernel to buffer data before it is written to disk.

**- Cached:** The amount of memory used to cache data from the file system for quick access.

**- Swap:** The amount of swap space used by the system.

4.5 \*What are log files for, and what are some examples of their use?

Log files are used to record the history of events, actions, or errors that occur in a system, program, or service. They are an important tool for analyzing and monitoring system performance, identifying problems, and debugging software. Here are some examples of how log files are used:

**- Diagnosing errors:** Log files allow you to store information about errors that occur during program execution or system operation. This helps administrators and developers identify problems and fix them quickly.

**- Monitoring system performance:** Log files can contain information about a variety of events, such as processes starting or ending, system connections, configuration changes, and more. This allows administrators to monitor system performance and respond to possible problems or security threats in a timely manner.

**- Performance analysis:** Log files can contain information about the use of system resources, such as central processing unit (CPU), RAM, disk space, etc.

**- History storage:** Log files allow you to store a history of events and activities on the system, which can be useful for analyzing and understanding previous events or for recovering the system after problems occur.

4.6 \*\*What is the purpose of the /var/log/dmesg file?

The /var/log/dmesg file is a text log file that stores dmesg output, which includes information about devices detected by the kernel, their initialization status, error messages that occur during system boot, and other important events related to the kernel.

This file can be useful for analyzing system boot problems, detecting hardware problems, monitoring the boot process, and identifying potential problems that occur during kernel operation. This file can be used by system administrators to diagnose problems, debug, or analyze system performance.

4.7 \*\*What is FHS designed for?

- Standardization of the file system structure: The FHS defines a standard directory structure that should be common to all Unix-like operating systems. This is important to ensure software compatibility and data exchange between different systems.

- Simplify system management: With a standardized directory structure, it is easier for system administrators to manage files and directories, which makes it easier to administer and maintain systems.

- Ensuring software portability: Software developers can use FHS to create applications that will run on a variety of Unix-like operating systems without the need to adapt to each system separately.

- Enhancing stability and security: Another important function of FHS is to ensure system stability and security. Standardized file system structure rules allow administrators and developers to ensure system security and resilience by considering specific file locations and access rights restrictions.

4.8 \*\*What are the basic commands in Linux for viewing and configuring a network?

- ifconfig: The ifconfig (interface configuration) command is used to display information about the network interfaces on the system, such as IP address, subnet mask, MAC address, etc.

- ip: The ip command provides advanced network configuration options, such as configuring IP addresses, subnet masks, routing, VLANs, etc.

- route: The route command is used to view and modify the routing table on the system.

- netstat: The netstat command is used to display a variety of information about network connections, routes, ports, statistics, and more.

- nmcli: The Network Manager Command-Line Interface (nmcli) is a command-line interface for managing the network using NetworkManager. It allows you to configure connections, network devices, VPNs, and more.

- iw: The iw command is used to configure and display information about wireless interfaces (Wi-Fi).

- ss: The ss command displays detailed information about network connections, sockets, and other network statistics.

5. Prepare an initial version of the report in electronic form:

- Title page, topic and purpose of the work.

- Glossary of terms.

- Answers to p.4.1 and p.4.5 from the preliminary preparation tasks.

**Procedure:**

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

1.1. Start the VirtualBox virtual machine, select CentOS, and start it. Log in to the system as a user: CentOS, login password: reverse (if you are performing the LP in 401) and launch the terminal.

1.2. Start the Ubuntu\_PC virtual machine (if you are performing the PL task through the netacad academy)

1.3. Start your Linux operating system (if you are working on your own PC and have installed it) and launch the terminal.

2. Work through all the sample commands presented in the labs of the NDG Linux Essentials course - Lab 13: Where Data is Stored and Lab 14: Network Configuration. Create a table to describe these commands.

|  |  |
| --- | --- |
| **Team name** | **Its purpose and functionality** |
| su | Change the current user to root |
| ls /proc | View the contents of the /proc system directory (root access is required) |
| cat /proc/cmdline | Displays the command line arguments that were passed to the Linux kernel during boot |
| ping localhost > /dev/null | Used to perform a ping to the localhost and redirect the ping output to /dev/null, which is a special file on Linux that ignores input and always returns empty output. |
| jobs | Used to display a list of active (running) tasks in the current shell or session |
| fg %1 | Used to switch the background process to the foreground |
| bg %1 | Used to control processes that have been suspended or to start a new process in the background |
| kill %3 | Used to send a signal to the process that is in the list of active (running) processes under the specified number, which is indicated in the percentage sign |
| killall ping | Used to send a signal to all processes named "ping" on your system |
| top | Used to display information about the system and processes running on it |
| sleep 888888 & | Used to delay the execution of a script or command for a specified time in seconds |
| ls /var/log | Used to display a list of files and directories in the /var/log directory on your file system |
| route | Used in Unix-like operating systems to display and manage the routing table that defines which networks are reachable through which network interfaces and how |
| dig cserver.example.com | Used to perform a DNS query for the domain name cserver.example.com |
| netstat --help | Used to display help information about using the netstat command |

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

- \*In this lab, the cat command was used, explore its capabilities and describe what tasks it is intended for;

- \*demonstrate examples where the cat command is used to create a file, view the contents of a file, redirect information to another file, and merge several files into one;

- What parameters of the cat command should be used to number lines in a file, display unprintable characters, and delete blank lines?

- describe the capabilities of the dig command and provide examples;

The dig command is a tool for diagnosing network problems and performing DNS queries. It allows you to get information about DNS records for specific domains, check server availability, resolve hostnames, and much more. The main features of the dig command include:

- Check DNS records: Allows you to check DNS records such as A, AAAA, CNAME, MX, TXT, etc. for the specified domain.

For example: 'dig example.com'

- Display full detailed result: Shows the full information including request and response headers.

For example: 'dig +trace example.com'

- Query a specific DNS server: Allows you to specify a specific DNS server to perform the query.

For example: 'dig @8.8.8.8 example.com'

- Queries using a specific record type: Allows you to select a specific type of DNS record.

For example: 'dig -t MX example.com'

- Reverse DNS check: Allows you to get the hostnames associated with the specified IP address.

For example: 'dig -x 8.8.8.8'

- Describe the capabilities of the netstat command and provide examples;

The netstat command is a tool for displaying various information about network connections, routes, interfaces, and more. It allows you to analyze network status, check active connections, listening ports, and other information about network resources. The main features of the netstat command include:

- Displaying all active connections: Displays all active connections that are established or active on the current system.

For example: 'netstat -a'

- Displaying information about network interfaces: Shows information about all network interfaces, including IP addresses, status, and other useful information.

For example: 'netstat -i'

- Show listening ports: Shows the ports on which the system is listening for connections.

For example: 'netstat -l'

- Display routing information: Shows the routing table that defines how the system determines where to route network traffic.

For example: 'netstat -r'

- Show other user's current network connections: Shows the current network connections for a specific user.

For example: 'netstat -tu -p | grep username'

**Control questions:**

1. What is the relationship between the 'cat' and 'tac' commands?

The 'cat' command is designed to print the contents of a file to the standard output. The 'tac' command, on the other hand, displays the contents of a file in reverse order, i.e. from the last line to the first. Thus, 'tac' is an analog of 'cat', but with the contents of the file printed in reverse order.

2. What does the 'ss' command do?

The 'ss' command is designed to display information about the sockets that are open on the system, including information about active network connections, listening ports, network routes, etc.

3. What is the difference between the 'ps -forest' and 'pstree' commands?

The 'ps -forest' command displays a list of processes in a tree view, where parent processes are displayed along with their children. 'pstree' also displays a list of processes in a tree, but it does so in a more graphical and understandable format.

4. \*Where are the system settings stored?

System settings are usually stored in the '/etc' and '/usr/local/etc' directories.

5. In which directories can I find the user programs installed on the system?

The installed programs available to the user are usually located in the directories '/bin', '/usr/bin', '/sbin', '/usr/sbin'.

6. \*In which directories can I find the installed system programs and programs intended for execution by the superuser?

System programs and programs executed by the superuser are usually located in the directories '/sbin', '/usr/sbin', '/usr/local/sbin'.

7. \*\*Explain the purpose of the ping, ifconfig, traceroute commands.

* **ping** - the command is used to check the availability of network devices by sending ICMP requests (usually "ping requests") and waiting for responses from them.
* **ifconfig** - the command is used to display information about network interfaces on the system, such as their IP addresses, status, subnet mask, etc.
* **traceroute** - the command is used to determine the path that data packets take from one network node to another, displaying the sequence of routes that the packets take and the delay time at each step.

8. \*\*What are network interfaces called in Linux?

Network interfaces in Linux are usually referred to simply as "network interfaces". Each network interface has its own unique name, which usually consists of a prefix indicating the type of interface (for example, "eth" for Ethernet), followed by the interface number. For example:

- Ethernet interfaces can have names like 'eth0', 'eth1', and so on.

- Wireless interfaces can have names like 'wlan0', 'wlan1', and so on.

- The loopback interface is named 'lo'.

9. \*\*How can I use the ifconfig command to display the parameters of only one network interface (for example, eth1), and not all of them?

To display the parameters of only one network interface using the 'ifconfig' command, you need to specify the name of this interface. For example, to display the parameters of the eth1 interface, use the following command: 'ifconfig eth1'.

**Conclusion:** you have gained practical skills in working with the Bash shell; familiarized yourself with the basic structures for storing system data - processes, memory, log files, and kernel status messages; familiarized yourself with the FHS standard; and learned how to configure the network.