“Kyiv Vocation College of Communication”
Cyclical Commission of Computer Engineering

REPORT ON EXECUTION
LABORATORY WORK №8

on the discipline: "Operating Systems"

Topic: "Preservation of system's operational data and its network configuration"

Performed by student
group RPZ-13a
Kateryna Hranat
Checked by teacher
V.S. Sushanova

Performed by student group RPZ-13a Kateryna Hranat

Kyiv 2024

Work objectives:

1. Acquiring practical skills in working with the Bash command shell.
2. Introduction to basic structures for storing system data - processes, memory, log files, and kernel status messages.
3. Familiarization with the FHS standard.
4. Introduction to networking configuration actions.

Material Support for Classes:

1. IBM PC type computers.
2. Windows operating system and VirtualBox virtual machine (Oracle).
3. GNU/Linux operating system (any distribution).
4. Cisco Networking Academy website netacad.com and its online courses on Linux.

Performed by student group RPZ-13a Kateryna Hranat

Tasks for Preliminary Preparation:

1. Read the brief theoretical information for the laboratory work and create a small dictionary of basic English terms related to the purpose of commands and their parameters.

English terms	Ukrainian terms
Log Files	Журнальні файли
Kernel Messages	Повідомлення ядра
Filesystem Hierarchy Standard	Стандарт ієрархії файлової системи
Host	ЦиклиХост
Network	Мережа
Porting	Портування

2. Study the materials of the online course "NDG Linux Essentials" from Cisco:
    - Chapter 13 - Where Data is Stored
    - Chapter 14 - Network Configuration

Complete✔

3. Complete testing in the NDG Linux Essentials course on the following topics:
    - Chapter 13 Exam
    - Chapter 14 Exam

Complete✔

4. Based on the material covered, provide answers to the following questions:

    4.1. What is the concept of a "pseudo filesystem," and why is it needed by the system?

A pseudo filesystem is a filesystem-like interface provided by the operating system kernel to access kernel data structures and system information. It

doesn't store data on disk like a traditional filesystem but presents information dynamically generated by the kernel. This mechanism allows users and applications to interact with various aspects of the kernel and system without needing special privileges or access to kernel internals directly.

4.2. Why don't users often directly access the /proc directory, and how can information be obtained from it?**

Users typically don't access the /proc directory directly because its contents are not traditional files stored on disk but rather dynamically generated by the kernel. Information from /proc can be obtained using various commands such as `cat`, `grep`, `awk`, or specialized utilities like `top`, `ps`, or `vmstat`. These commands read and interpret the information exposed by the virtual files in /proc.

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

- `/proc/cmdline`: Contains the command line parameters passed to the kernel during boot, including boot options and parameters.
- `/proc/meminfo`: Provides information about memory usage, including total memory, free memory, swap usage, and more.
- `/proc/modules`: Lists the currently loaded kernel modules along with information about each module.

4.4. *What is the purpose of the `free` command?

The `free` command displays information about system memory usage, including total available memory, used memory, free memory, and memory used for buffers and caches.

4.5. *Why are log files necessary, and what are examples of their application?

Log files are essential for recording system events, errors, warnings, and other relevant information for troubleshooting, monitoring, and auditing purposes. Examples of their application include diagnosing system crashes,

tracking user activity, monitoring network traffic, and identifying security breaches.

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

The file /var/log/dmesg contains the kernel ring buffer, which logs boot messages and system hardware initialization information. It provides a historical record of events that occurred during system startup.

4.7. **What is the purpose of the Filesystem Hierarchy Standard (FHS)?

The Filesystem Hierarchy Standard (FHS) is a set of guidelines and standards for organizing the structure and contents of the Linux filesystem. It defines the directory structure and the purpose of each directory, ensuring consistency and compatibility across different Linux distributions.

4.8. **What are the main commands in Linux for viewing and configuring the network?

Main commands for viewing and configuring the network in Linux include:

- `ifconfig` (or `ip addr`): Displays and configures network interfaces.
- `route` (or `ip route`): Displays and configures routing tables.
- `netstat` (or `ss`): Displays network statistics and connections.
- `ping`: Tests network connectivity to a remote host.
- `traceroute` (or `traceroute`): Traces the route packets take to reach a destination host.
- `iptables` (or `firewalld`): Configures firewall rules.
- `nmcli` (Network Manager Command-Line Interface): Manages NetworkManager-based connections.
- `iwconfig` (or `iw`): Configures wireless network interfaces.

5. Prepare an initial version of the report in electronic form:
   - Title page, topic, and purpose of the work
   - Glossary of terms
   - Answers to points 4.1 - 4.8 from the tasks for preliminary preparation

Complete✔

Performed by student group RPZ-13a Kateryna Hranat

Progress of Work:

1. Initial work in CLI mode in a Linux operating system of the Linux family:
   1.1. Start your Linux-based operating system (if you are using your own PC and have it installed) and open the terminal.
2. Work through all the command examples provided in the lab assignments of the NDG Linux Essentials - Lab 13: Where Data is Stored та Lab 14: Network Configuration. Create a table to describe these commands.

Command	Description
`ls`	Lists files and directories in the current directory.
`cat`	Concatenates and displays the contents of files.
`mkdir`	Creates a new directory.
`cp`	Copies files and directories.
`mv`	Moves or renames files and directories.
`rm`	Removes (deletes) files and directories.
`touch`	Creates an empty file or updates the access and modification times of an existing file.
`nano`	A text editor used for creating and editing text files.

`grep`	Searches for a specified pattern in a file or files.
`head`	Displays the first few lines of a file.
`tail`	Displays the last few lines of a file.
`less`	Allows navigation through large text files.
`chmod`	Changes the permissions of files and directories.
`chown`	Changes the owner and group of files and directories.
`sudo`	Executes a command with superuser (root) privileges.
`passwd`	Changes the password of the current user or another user.
`hostname`	Displays or sets the system's hostname.
`ping`	Tests the reachability of a host on an IP network.
`ifconfig`	Displays or configures network interfaces.
`ip`	Shows/manages routing, devices, policy routing, and tunnels.
`route`	Displays or modifies the IP routing table.
`traceroute`	Traces the route taken by packets across an IP network.

`netstat`	Displays network connections, routing tables, interface statistics, masquerade connections, etc.
`ssh`	Securely connects to a remote computer.
`scp`	Securely copies files between hosts on a network.
`ftp`	Transfers files between a client and a server over a network.
`wget`	Downloads files from the web using HTTP, HTTPS, or FTP protocols.
`curl`	Transfers data from or to a server, supports multiple protocols including HTTP, HTTPS, and FTP.

3. Execute practical tasks in the terminal (demonstrate screenshots):

- In this lab, the `cat` command was used, explore its capabilities and describe its purposes for various tasks.

1. Exploring the `cat` command:

- The `cat` command is primarily used for concatenating files and displaying their contents.
- It is also useful for creating new files, redirecting information to other files, and numbering lines in a file.

- *Demonstrate examples of using the `cat` command for file creation, viewing file content, redirecting information to another file, and concatenating multiple files into one.

2. Examples of `cat` command usage:

- Creating a new file:

```
sysadmin@localhost:~$ cat > newfile.txt
```

- Viewing the contents of a file:

```
sysadmin@localhost:~$ cat file.txt
```

- Redirecting information to another file:

```
sysadmin@localhost:~$ cat file1.txt > file2.txt
```

- Concatenating multiple files into one:

```
sysadmin@localhost:~$ cat file1.txt file2.txt > combined.txt
```

- *Describe the parameters needed for the `cat` command to number lines in a file, display non-printable characters, and remove empty lines.

3. Parameters for `cat` command:
   - To number lines in a file: `cat -n file.txt`
   - To display non-printable characters: `cat -v file.txt`
   - To remove empty lines: `cat -s file.txt`

- **Describe the capabilities of the `dig` command and provide examples.

4. Description and examples of the `dig` command:
   - The `dig` command is used for querying DNS servers to retrieve DNS information.

```
sysadmin@localhost:~$ dig example.com

; <<>> DiG 9.11.3-1ubuntu1.13-Ubuntu <<>> example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 36645
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 35ed7b4ffccf97db6b2328ad660d5e65e0253cf2c7cfcce4 (good)
;; QUESTION SECTION:
;example.com.                   IN      A

;; ANSWER SECTION:
example.com.            86400   IN      A       192.168.1.2

;; AUTHORITY SECTION:
example.com.            86400   IN      NS      example.com.

;; Query time: 1 msec
;; SERVER: 127.0.0.11#53(127.0.0.11)
```

- **Describe the capabilities of the `netstat` command and provide examples.

5. Description and examples of the `netstat` command:

- The `netstat` command is used for displaying network connections, routing tables, interface statistics, and more.

- Example:

```
sysadmin@localhost:~$ netstat -tuln
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.11:42505        0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:953           0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:53            0.0.0.0:*               LISTEN
tcp        0      0 192.168.1.2:53          0.0.0.0:*               LISTEN
tcp6       0      0 :::22                   :::*                    LISTEN
tcp6       0      0 :::53                   :::*                    LISTEN
udp        0      0 192.168.1.2:53          0.0.0.0:*
udp        0      0 127.0.0.1:53            0.0.0.0:*
udp        0      0 127.0.0.11:54945        0.0.0.0:*
udp6       0      0 :::53                   :::*
```

Control questions

1. How are the `cat` and `tac` commands related to each other?

- The `cat` command is used to concatenate files and display their contents, while the `tac` command is used to concatenate and display files in reverse order, i.e., the last line first.

- Execution:

```
cat file.txt    # Display contents of file
tac file.txt    # Display contents of file in reverse order
```

2. What does the `ss` command do?

- The `ss` command is used to investigate sockets, which includes displaying information about TCP, UDP, and UNIX domain sockets.

- Execution:

```
ss -tuln        # Display TCP and UDP listening sockets
```

3. What is the difference between the `ps --forest` command and `pstree` command?

- The `ps --forest` command displays processes in a hierarchical tree-like structure, while the `pstree` command displays processes as a tree.
    - Execution:
      ps --forest # Display processes in a hierarchical tree-like structure
      pstree # Display processes as a tree

4. *In which directories are system settings stored?

- System settings are stored in directories such as `/etc`, `/usr/share`, and `/var`.

5. *In which directories can installed programs available to users be found?

- Installed programs available to users can be found in directories such as `/bin`, `/usr/bin`, and `/usr/local/bin`.

6. *In which directories can system programs and programs intended for superuser execution be found?

- System programs and programs intended for superuser execution can be found in directories such as `/sbin`, `/usr/sbin`, and `/usr/local/sbin`.

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

- `ping`: Tests the reachability of a host on an IP network by sending ICMP echo request packets.
- `ifconfig`: Displays or configures network interfaces, including their IP addresses, netmasks, and status.
- `traceroute`: Traces the route taken by packets across an IP network, displaying the path and latency of each hop.

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

- Network interfaces in Linux are called network devices or network cards.

9. **How can you use the `ifconfig` command to display parameters for only one network interface (e.g., eth1) instead of all?

- You can use the `ifconfig` command followed by the specific interface name to display parameters for only that interface.
    - Execution:
      ifconfig eth1 # Display parameters for the eth1 interface

Conclusions:

Acquiring practical skills in working with the Bash command shell is essential for efficient system administration and automation tasks. By mastering commands, operators, and scripting techniques, users can streamline their workflow and effectively manage system resources.

Familiarity with basic structures for storing system data such as processes, memory, log files, and kernel status messages is crucial for system monitoring, troubleshooting, and optimization. Knowing where and how this data is stored allows administrators to diagnose issues, track system performance, and ensure stability.

Understanding the Filesystem Hierarchy Standard (FHS) is essential for maintaining consistency and organization within the Linux filesystem. Adhering to FHS guidelines ensures that system files, directories, and configurations are placed in predictable locations, facilitating easier management and interoperability across different Linux distributions.

Introduction to networking configuration actions provides users with the foundational knowledge required to set up, manage, and troubleshoot network connections. Learning how to configure network interfaces, set IP addresses, manage routing tables, and utilize network services enables administrators to establish reliable communication channels and ensure seamless connectivity within their infrastructure.

Overall, acquiring these skills not only enhances proficiency in Linux system administration but also empowers users to effectively manage and optimize their systems, ensuring reliability, security, and performance.