“Kyiv specialized College of Communications”

Commission of computer engineering

**REPORT ON THE IMPLEMENTATION**

**LABORATORY WORK №8**

From the discipline: "Operating systems"

**Topic: "Saving system service data and its network configuration."**

The students

performed Groups RPZ-03

Team 3:

Kanavets K.S.,

Kryvenko A.I.,

Kulikovska M.V.

Checked by the teacher

Sushanova V.S.

Кyiv 2023

Tasks for preliminary preparation.

**The material was prepared by student Kulikovska Maria (@Smith5004)**

Read the brief theoretical information before the lab and make a small glossary of basic English terms on the purpose of commands and their parameters.

guideline - інструкція

governed - керований

malicious programs - вредоносні програми

unauthorized access - несанкціонований доступ

accessible network - доступна мережа

resilience - надійність

troubleshoot problems - вирішення проблем

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

A pseudo file system means a file system that has no real files, but rather virtual records that the file system itself creates in place. Pseudofile systems appear to be real files on disk, but exist only in memory. Most pseudo-file systems, /proc created to appear to be a hierarchical tree from the root system of directories, files, and subdirectories, but actually exist only in memory of the system and appear to be resident only on the storage device that the root file system is included.

*4.2 Why do users not often access the /proc directory directly, and how can they get information from it?*

The proc file system provides detailed information about the kernel, processes and configuration options in a structured way in the /proc directory. Some files can be viewed by the shell user. However, many files cannot be accessed unless you have root privileges***.***

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

/proc/cmdline. Information that was passed to the kernel when it was first started, such as command line parameters and special instructions.

/proc/meminfo. Information about the kernel's memory usage

/proc/modules. A list of modules currently loaded into the kernel to add additional functionality.

Overall, these files in /proc provide important information about the system's configuration and performance, which can be used for monitoring, debugging, and troubleshooting purposes.

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

The free command allows you to call the utility of the same name, which displays information about the use of RAM. This command allows you to view information about the use of both physical RAM memory, represented by modules with a set of chips in special slots on the computer's motherboard, and memory from a swap partition or swap file (SWAP) located on a hard disk or solid-state drive.

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

Log files are the primary data source for network [observability](https://www.sumologic.com/glossary/observability/). A log file is a computer-generated data file that contains information about usage patterns, activities, and operations within an operating system, application, server or another device. Log files show whether resources are performing properly and optimally.

For example, you can use security logs to check for successful or failed user logins, system logs to investigate server outages, application logs to detect application failures, and more.

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

/var/log/dmesg - contains information about the kernel ring buffer. When the system boots, it displays a series of messages showing information about the hardware devices detected by the kernel during the boot process. These messages are available in the kernel ring buffer, and whenever a new message arrives, the old message is overwritten. You can also view the contents of this file with the command dmesg.

*4.7. What is the FHS designed for?*

The File System Hierarchy Standard (FHS) contains a set of requirements and guidelines for placing files and directories in Linux and any Unix-like operating system. The idea behind the FHS is to organize directories so that shared files can be placed in one place to make them easily accessible to programs, while separating them from personal or exclusive files, or, in another aspect, we can separate static and variable files and put all static files on read-only media, while frequently changing files (such as logs) can be placed on a fast drive.

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

ifconfig - display and manipulate route and network interfaces.

ip - it is a replacement of ifconfig command.

traceroute - network troubleshooting utility.

tracepath - similar to traceroute but doesn't require root privileges.

ping - to check connectivity between two nodes.

netstat - display connection information.

ss - it is a replacement of netstat.

dig - query dns related information.

nslookup - find dns related query.

route - shows and manipulate ip routing table.

host - performs dns lookups.

arp - view or add contents of the kernel's arp table.

iwconfig - used to configure wireless network interface.

hostname - to identify a network name.

curl or wget - to download a file from internet.

mtr - combines ping and tracepath into a single command.

whois - will tell you about the website's whois.

ifplugstatus - tells whether a cable is plugged in or not.

***The material was prepared by student Kanavets Kateryna (kanavetsk)***

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

|  |  |
| --- | --- |
| Command name | Its purpose and functionality |
| su | Change the current user to root. |
| ls /proc | View the contents of the /proc system directory (this requires root access). |
| cat /proc/1/cmdline; echo | The command cat /proc/1/cmdline will display the command-line arguments passed to the init process on a Linux system. |
| ps -p 1 | The command ps -p 1 will display information about the process with process ID 1, which is the init process on a Linux system. The output will include information such as the process ID (PID), the parent process ID (PPID), the CPU and memory usage, and the command-line arguments used to start the process. |
| cat /proc/cmdline | The command cat /proc/cmdline displays the command line arguments passed to the Linux kernel at boot time. These arguments can include various kernel parameters and options that affect the behavior of the kernel and the system as a whole. |
| ping localhost > /dev/null | The command ping localhost > /dev/null sends ICMP echo requests to the loopback address (127.0.0.1), which represents the local machine, and redirects the output of the ping command to /dev/null. |
| jobs | The jobs command is used to list the jobs running in the current shell session. A job is a process that has been started from the current shell session and is associated with a job number. |
| fg %1 | The command fg %1 brings the job with job number 1 to the foreground, allowing you to interact with it as if it were running in the foreground. |
| bg %1 | The command bg %1 resumes the execution of the job with job number 1 in the background, allowing it to continue running without blocking the terminal. |
| kill %3 | The command kill %3 sends the default TERM signal to the job with job number 3, which will attempt to gracefully terminate the corresponding process. |
| killall ping | The command killall ping sends the default TERM signal to all processes whose name is "ping", causing them to gracefully terminate. |
| top | The top command is a Unix/Linux utility that provides real-time information about the system's performance and the processes running on the system. It displays a dynamic view of the system's processes, showing their resource usage, process ID, CPU and memory usage, and other information. |
| sleep 888888 & | The command sleep 888888 & starts a new process that sleeps for 888888 seconds (about 10.3 days) in the background. |
| ps | The ps command is used to display information about the processes running on a Unix/Linux system. It displays a list of processes, along with information such as their process ID (PID), parent process ID (PPID), CPU and memory usage, and command-line arguments. |
| kill PID | The command kill PID sends the default TERM signal to the process with the specified process ID (PID), causing it to gracefully terminate. |
| pkill -15 sleep | The command pkill -15 sleep sends the SIGTERM signal to all processes whose name is "sleep", causing them to gracefully terminate. |
| ps -e | The command ps -e displays information about all processes running on the system, regardless of which user started them. |
| ps -o pid,tty,time,%cpu,cmd | The command ps -o pid,tty,time,%cpu,cmd displays a customized output of the processes running on the system. It displays the process ID (PID), terminal (TTY), CPU time used, CPU usage percentage, and command-line arguments. |
| ps -o pid,tty,time,%mem,cmd --sort %mem | The command ps -o pid,tty,time,%mem,cmd --sort %mem displays a customized output of the processes running on the system sorted by memory usage in descending order. It displays the process ID (PID), terminal (TTY), CPU time used, memory usage percentage, and command-line arguments. |
| free | The free command is used to display information about the amount of free and used memory on a Unix/Linux system. |
| ls /var/log | The command ls /var/log displays a list of files and directories located in the /var/log directory. This directory contains log files for various system services and applications running on the system. |
| ssh localhost | The ssh localhost command initiates a secure shell (SSH) connection to the local machine (localhost). SSH is a network protocol used to securely connect to remote machines over an unsecured network. |
| tail -5 /var/log/auth.log | The command tail -5 /var/log/auth.log displays the last 5 lines of the /var/log/auth.log file, which contains information about user authentication and authorization on the system. |
| route | The route command is used to display and manipulate the network routing table in a Unix/Linux system. The routing table is a set of rules that determine how network traffic is directed between different networks. |
| grep 127.0.0.1 /etc/hosts | The grep 127.0.0.1 /etc/hosts command searches the /etc/hosts file for any lines that contain the IP address 127.0.0.1. The /etc/hosts file is a local file used to map IP addresses to hostnames on a Unix/Linux system. |
| ping -c4 localhost | The ping -c4 localhost command sends four ICMP echo request packets to the local machine (localhost) and waits for a response from the machine. It is commonly used to test network connectivity and latency between the local machine and itself. |
| cat /etc/resolv.conf | The cat /etc/resolv.conf command displays the contents of the /etc/resolv.conf file, which contains information about the domain name system (DNS) servers used by the system. |
| dig localhost.localdomain | The dig localhost.localdomain command is used to query the domain name system (DNS) for information about the localhost.localdomain domain name. localhost.localdomain is a default domain name that is commonly used for testing and development purposes. |
| dig cserver.example.com | The dig cserver.example.com command is used to query the domain name system (DNS) for information about the cserver.example.com domain name. This command will attempt to resolve the IP address associated with the cserver.example.com hostname. |
| dig -x 192.168.1.2 | The dig -x 192.168.1.2 command is used to perform a reverse DNS lookup, also known as a reverse resolution, to find the domain name associated with the IP address 192.168.1.2. This is accomplished by performing a DNS query on the special domain in-addr.arpa, which is used for reverse DNS lookups. |
| netstat --help | The dig -x 192.168.1.2 command is used to perform a reverse DNS lookup, also known as a reverse resolution, to find the domain name associated with the IP address 192.168.1.2. This is accomplished by performing a DNS query on the special domain in-addr.arpa, which is used for reverse DNS lookups. |
| netstat -tl | The netstat -tl command displays a list of all TCP sockets that are currently listening on the local machine. The output includes information about the local address and port number of each listening socket, as well as the state of the socket. |
| netstat -tln | The netstat -tln command displays a list of all TCP sockets that are currently listening on the local machine in a numerical format. The output includes information about the local address and port number of each listening socket, as well as the state of the socket. |
| ss | The ss command is a utility used to display detailed information about network connections, sockets, and routing tables on a Linux system. The ss command is similar to the netstat command, but is more powerful and provides more detailed information. |

***The material was prepared by student Kryvenko Andrew (AndrewKryvenko)***

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

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

Display the contents of a file on the screen: cat file.txt

Create a new file: cat > file.txt. After executing this command, you can enter text that will be saved in the file "file.txt". To finish typing and save the file, press Ctrl+D.

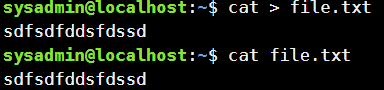
View the contents of several files at once: cat file1.txt file2.txt

Redirect the contents of a file to another file: cat file1.txt > file2.txt. This will redirect the contents of file "file1.txt" to file "file2.txt", overwriting the contents of the latter.

Append the contents of one file to the end of another file: cat file1.txt >> file2.txt. This will append the contents of file "file1.txt" to the end of file "file2.txt".

Combine the contents of several files into one file: cat file1.txt file2.txt > file3.txt. This will combine the contents of the files "file1.txt" and "file2.txt" into a single file "file3.txt".

- Demonstrate examples of when the cat command is used to create a file, view the contents of a file, redirect information to another file, and merge multiple files into one;



Create a file with text and view the contents of the file.



Redirect the contents of a file to another file.



Add the contents of one file to the end of another file.



Combine the contents of multiple files into a single file.

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

Number the lines of a file: Use the "-n" or "--number" option.

cat -n file.txt

This command will display the contents of the file "file.txt" with the lines numbered.

Displays non-printable characters: Use the "-v" or "--show-nonprinting" option.

cat -v file.txt

This command will display the contents of the file "file.txt", showing nonprinting characters such as tabs, carriage returns, etc.

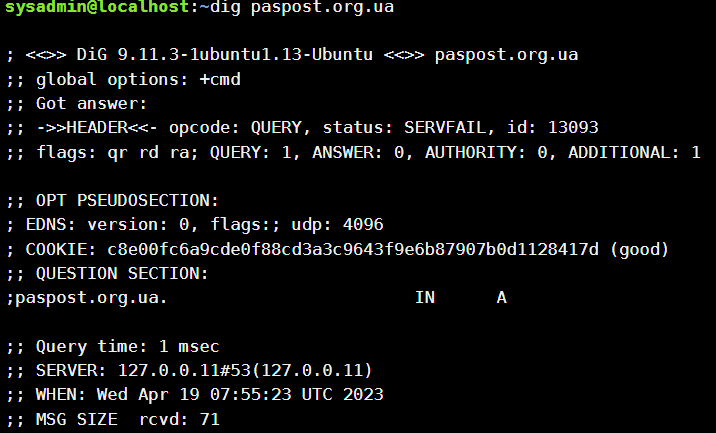
Removes blank lines: Use the "-s" or "--squeeze-blank" option.

cat -s file.txt

This command will display the contents of the "file.txt" file, removing any duplicate lines.

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

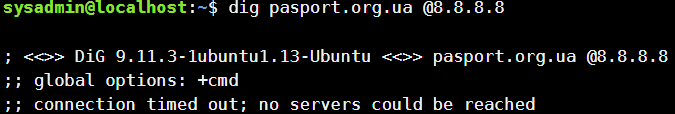
Obtaining information about a domain name.



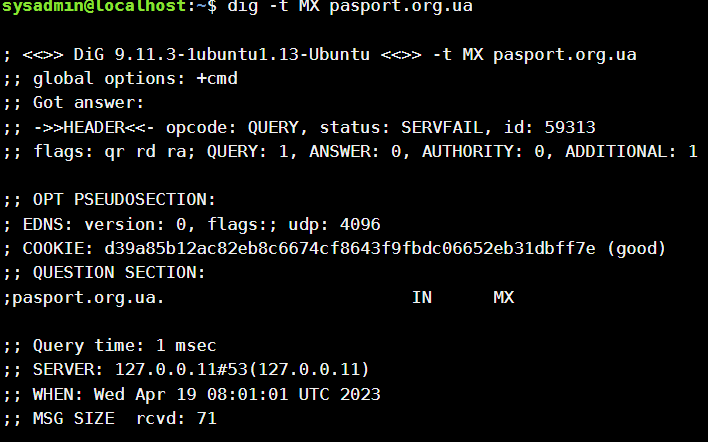
Displaying the DNS cache



Testing the connection to the DNS server

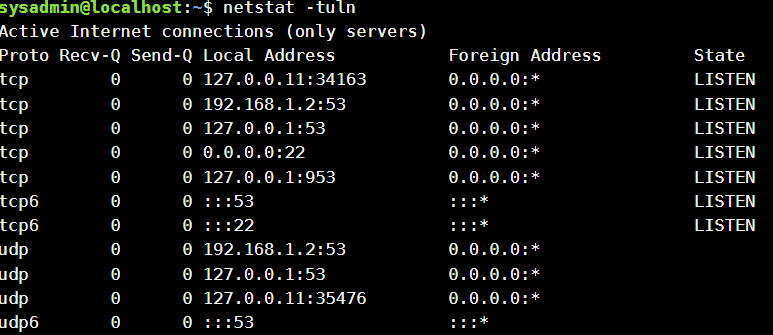


Using different types of DNS records

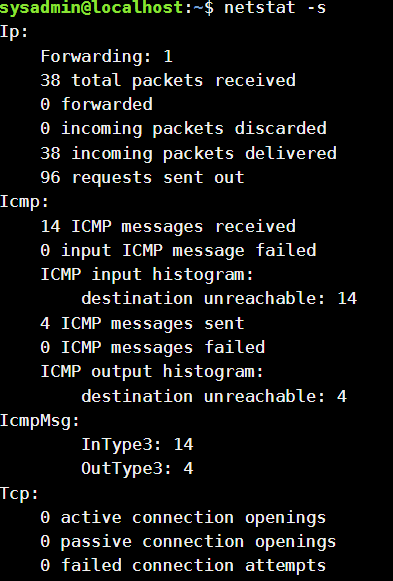


- describe the capabilities of the netstat command and give examples;.

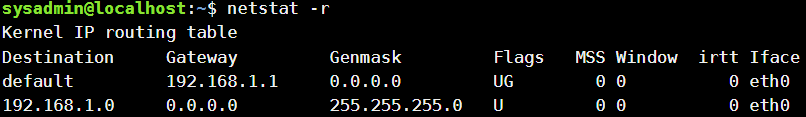
Display active network connections



Display of network statistical



Displaying network routes



Control questions

**The material was prepared by student Kulikovska Maria (@Smith5004)**

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

**Cat** command is a well known Unix utility that reads files sequentially. Writing them to conventional output. The name is derived from its function for concatenating and listing the documents. **Tac** (that is “cat” backwards) concatenates every record to traditional output much like the cat command. However in opposite: line-by means of-line, printing the last line first. This article explains about “How to use ‘cat’ and ‘tac’ commands with examples”.

1. *What do the ss command do?*

The ss command is designed to display socket statistics and supports all major packet and socket types. Designed to replace the command and similar in function to netstat, it also shows much more information and has more features.

The main reason the user will use this sscommand is to see what connections are currently established between his local computer and remote computers, statistics about these connections, etc.

1. *What is the difference between the ps --forest and pstree commands?*

The ps --forest and pstree commands are designed to display information about the processes in a tree structure. There are a few differences between these commands, however.

The ps --forest command displays the tree of the processes associated with the current terminal, including the parent processes and their descendants. It uses the output format of the ps command, which displays the list of processes in a hierarchical structure with each child process aligned to its parent. Thus ps --forest outputs information about processes with a multi-level hierarchical structure.

The pstree command, on the other hand, shows the complete tree of processes running on the system, including parent processes, child processes and their relationships. In contrast to ps --forest, pstree uses its own output format which shows the process tree as a textual graph.

1. *Where are the system settings stored?*

On Linux, system settings are stored in different directories, depending on the specific configuration files used by the system and programs.

The most common directories where system settings are stored are /etc, /usr/share, /var, /opt, and /usr/local.

1. *Where can I find the programs installed in the system that are available to the user?*

In the following directories you can find the programs installed in the system that are available to the user: /usr/bin, /usr/local/bin, /opt, ~/bin.

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

In the following directories you can find the installed system programs and programs intended for execution by the superuser: /bin, /sbin, /usr/bin, /usr/sbin, /usr/local/bin, /opt.

1. *Explain the purpose of the ping, ifconfig, traceroute commands.*

ping

Pinging is the act of sending packets to a device and measuring the time it takes to get a reply, in other words, it is a way to check if there is an established connection between two devices. It is typically used by writing ping followed by the address you are trying to reach. Example is shown below.

traceroute

traceroute is another handy networking command. This command is similar to ping, except it prints the route that it takes a packet to reach its destination.

ifconfig

This command is most used to check the IP address of your device as shown below. The highlighted section of the image below is the IP address of this device.

1. *What are the names of network interfaces in Linux?*

Network interface names are dynamic by default and are assigned on a first-come, first-served numerical basis: eth0, eth1, eth2, and so on.

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

Using the interface name (eth1) as an argument with the "ifconfig" command will show you the details of a particular network interface: ifconfig eth1.

Conclusion

In this lab, we learned about the basic structures for storing system data - processes, memory, log files, and kernel status messages, the FHS standard, and network configuration.