“Київський фаховий коледж зв’язку”

Циклова комісія Комп’ютерної та програмної інженерії

**ЗВІТ ПО ВИКОНАННЮ**

**ЛАБОРАТОРНОЇ РОБОТИ №6**

з дисципліни: «Операційні системи»

**Тема:** **“Команди Linux для архівування та стиснення даних. Робота з текстом”**

Виконали студенти

групи КСМ-13а

MathematicainsLivesMatter: Дзизиль Д. Є., Чех І. В.

Перевірив викладач

Сушанова В. С.

Київ 2023

**Мета роботи:**

**1.** Отримання практичних навиків роботи з командною оболонкою Bash.

**2.** Знайомство з базовими командами для архівування та стиснення даних.

**3.** Знайомство з базовими діями при роботі з текстом у терміналі.

**Матеріальне забезпечення занять**

1. ЕОМ типу IBM PC.

2. ОС сімейства Windows (Windows 7).

3. Віртуальна машина – Virtual Box (Oracle).

4. Операційна система GNU/Linux – CentOS.

5. Сайт мережевої академії Cisco netacad.com та його онлайн курси по Linux.

**Завдання для попередньої підготовки**

***Готував матеріал студент Дзизиль Д.Є.***

1. На базі розглянутого матеріалу дайте відповіді на наступні питання:
   1. Яке призначення команд *tar*, *xz*, *zip*, *bzip*, *gzip*? Зробіть короткий опис кожної команди та виділіть їх основні параметри. Яким чином їх можна встановити.

* Gzip is used to compress and decompress compressed files. It has the following parameters: -l — provides information about the file (how much space the uncompressed file takes up, but how much is compressed), -d — unpack the compressed file. sudo yum install gzip.
* Tar is used for archiving and unzipping files, viewing archive contents without unzipping. Has such parameters as: -c — create an archive, -f ARCHIVE — operate on this archive, -z — compress (or decompress) an archive using gzip, -j — compress (or decompress) an archive using bzip, -t — expand a list of files in the archive, -x — unpack files from the archive, -v — a detailed list of processed files. sudo yum install tar.
* ZIP is used to compress and archive files. It has the following options: -r — add files to the archive, -d — delete a file from the archive. sudo yum install zip.
* Xz is used to compress files. Has such options as: -c — compress the file, -d — unpack the compressed file. sudo yum install xz.
* Bzip is used to compress files. It has the following parameters: -z — compress the file, -d — unpack the compressed file. sudo yum install bzip2.
  1. Наведіть три приклади реалізації архівування та стискання даних різними командами.
* tar -cf alpha\_files.tar alpha\*
* gzip -l alpha\_files.tar.gz
* zip alpha\_files.zip alpha\*
  1. Яке призначення команд cat, less, more, head and tail? Зробіть короткий опис кожної команди та виділіть їх основні параметри. Яким чином їх можна встановити.
* Cat is used to create and display text files, and to merge copies of text files. It has such parameters as: -n — number the lines, file — the name of the file or files to be output. sudo yum install cat.
* Less is used to view large text files page by page. It has the following parameters: file — the name of the file to view, /pattern — search for a text pattern in the file. sudo yum install less.
* More is used to view files, has fewer functions than less. Parameters: file — the name of the file to view. sudo yum install more.
* Head is used to output the first few lines from a file. file — the name of the file from which the first lines should be output, -n — the number of lines to output. sudo yum install head.
* Tail is used to output the last few lines from a file. –f — view changes in the file in real time, -n — number of lines to output. sudo yum install tail.
  1. Поясніть принципи роботи командної оболонки з каналами, потоками та фільтрами.
* Pipes:

Principle of operation: A channel is a mechanism that allows you to transfer the output of one command to the input of another command without the need to save the intermediate result in a file. Channels are created using a vertical bar | and are used to combine several commands into one data processing chain.

* Streams:

How it works: In Linux, there are three main data streams: standard input (stdin), standard output (stdout), and standard error (stderr). The command shell allows you to redirect these input and output streams for commands. For example, you can redirect the output of a command to a file or pass a file as input to a command.

* Filters:

How it works: A filter is a command that processes input data and outputs the changed data to standard output. Filters are often used in conjunction with channels to manipulate data. Examples of filters include grep for searching text, sed for editing text, and awk for processing text data.

* 1. Яке призначення команди grep?

The grep command is a powerful tool for searching and filtering data in Linux and Unix-like operating systems. It enables users to search for text strings in a file or filter output from other commands using pattern strings. The grep command has several essential purposes, including searching for strings by pattern, output filtering, and others.

**Хід роботи**

1. Початкова робота в CLI-режимі в Linux ОС сімейства Linux:

1.1. Запустіть віртуальну машину VirtualBox, оберіть CentOS та запустіть її. Виконайте вхід в систему під користувачем: CentOS, пароль для входу: reverse (якщо виконуєте ЛР у 401 ауд.) та запустіть термінал.

1.2. Запустіть віртуальну машину Ubuntu\_PC (якщо виконуєте завдання ЛР через академію netacad)

1.3. Запустіть свою операційну систему сімейства Linux (якщо працюєте на власному ПК та її встановили) та запустіть термінал.

***Готував матеріал студент Дзизиль Д.Є.***

1. Опрацюйте всі приклади команд, що представлені у лабораторних роботах курсу ***NDG Linux Essentials - Lab 9: Archiving and Compression*** та ***Lab 10: Working With Text.***

|  |  |
| --- | --- |
| Назва команди | Її призначення та функціональність |
| mkdir mybackups | Creating a new mybackups directory in the user's home directory |
| tar -cvf mybackups/udev.tar /etc/udev | The tar command is used to combine multiple files into a single file. In this case, the contents of the /etc/udev directory will be saved in the udev.tar archive in the mybackups directory. The -c option tells the tar command to create a tar file. The -v option stands for "verbose", which tells the tar command to demonstrate what it is doing. The -f option is used to specify the name of the tar file. |
| ls mybackups | Lists the files and directories in the "mybackups" directory. |
| tar -tvf mybackups/udev.tar | Used to list the files and directories contained in the `udev.tar' archive located in the "mybackups" directory. |
| Tar -zcvf mybackups/udev.tar.gz /etc/udev | Creates an archive of files and directories from /etc/udev and saves it as a compressed tar.gz archive. |
| ls -lh mybackups | Lists the files and directories in the "mybackups" directory, along with size and permission information. |
| cd mybackups | Changes the current working directory to "mybackups" |
| ls | This command lists the files and directories in a directory. |
| tar -xvf udev.tar.gz | This command uses tar to extract the udev.tar.gz archive, which is located in the current directory ("mybackups"). |
| ls etc | These files are extracted from the dev.tar.gz archive, which is located in the current directory ("mibaccups"). |
| ls etc/udev | This command lists the files and directories that are in the "etc/udev" directory, which also appeared after the archive was unzipped. |
| ls etc/udev/rules.d | This command lists the files and directories that are in the "rules.d" subdirectory, which also appeared after the archive was unzipped. |
| tar -rvf udev.tar /etc/hosts | This command adds (or updates) the /etc/hosts file to the existing udev.tar archive. |
| cp /usr/share/dict/words . | Copies the file /usr/share/dict/words to the current working directory. A dot (.) indicates the current directory as the copy destination. |
| ls -l words | Displays information about the words file, including size and permissions. The words file has just been copied with the word dictionary. |
| gzip words | Compresses the words file using the gzip compression program. After executing this command, you get a compressed words.gz file. |
| ls -l words.gz | Displays information about the compressed words.gz file, including size and permissions. |
| gunzip words.gz | Extracts the compressed words.gz file using gunzip. After executing this command, you get the words file in its standard, uncompressed form. |
| ls -l words | Prints information about the extracted words file, including size and permissions. |
| bzip2 words | Uses the Bzip2 compression program to compress the words file. After executing this command, the words file is compressed and gets the extension .bz2. |
| ls -l words.bz2 | Prints information about the compressed words.bz2 file, including its size and permissions after compression. |
| bunzip2 words.bz2 | Uses the Bzip2 decompressor to decompress the words.bz2 compressed file. After executing this command, the words.bz2 file is unzipped and returned to its normal state as a words file. |
| xz words | Uses the XZ compression program to compress the words file. After executing this command, the words file is compressed and gets a .xz extension. |
| ls -l words.xz | Prints information about the compressed words.xz file, including its size and permissions, before unzipping. |
| unxz words.xz | Uses the XZ decompressor to decompress the compressed words.xz file. After executing this command, the words.xz file is unzipped and returned to its normal state as a words file. |
| zip words.zip words | Creates a compressed words.zip file that contains the contents of the words file. You use the zip utility to create a ZIP archive. A words file is added to this archive. |
| ls -l words.zip | Prints information about the compressed words.zip file, including its size and permissions. |
| zip -r udev.zip /etc/udev | Creates a compressed udev.zip archive that contains the entire contents of the /etc/udev directory and its subdirectories. The -r option indicates recursive compression, that is, the entire contents of the directory and subdirectories are included in the archive. |
| ls -l udev.zip | Prints information about the compressed udev.zip archive, including its size and permissions. |
| unzip -l udev.zip | Displays information about the contents of the udev.zip archive, namely a list of files and directories contained in the archive, without unpacking them. |
| rm -r etc | Deletes the etc directory and all its subdirectories and files. The -r option indicates recursive deletion, meaning all subdirectories and files will also be deleted. |
| unzip udev.zip | Extracts the udev.zip archive and restores the contents you deleted in the previous command. As a result, you will have the etc directory and its subdirectories and the files that were in the udev.zip archive. |
| echo "Hello World" | This command prints the string "Hello World" to standard output (usually the screen). |
| echo "Hello World" > mymessage | This command outputs the string "Hello World" and redirects it to a file called "mymessage". As a result, this file "mymessage" will be created or overwritten, and the contents of this file will be the string "Hello World". |
| cat mymessage | This command prints the contents of the file "mymessage" to standard output. |
| echo "Greetings" > mymessage | This command outputs the string "Greetings" and redirects it to a file called "mymessage". |
| echo "How are you?" >> mymessage | This command adds the line "How are you?" to the contents of the file "mymessage" without overwriting the previous contents. |
| find ~ -name "\*bash\*" | This command searches for all files and directories in the user's home directory that have "bash" in their name. Found objects will be displayed on the screen. |
| find /etc -name hosts | This command looks for a file(s) named "hosts" in the /etc directory. The well-known /etc/hosts file is a file that usually contains information about network addresses and hostnames on the system. The found file will be displayed on the screen. |
| find /etc -name hosts 2> err.txt | This command searches for a file named "hosts" in the /etc directory, and any error messages encountered during the search will be redirected to the err.txt file. Option 2> indicates redirection of the standard error stream (stderr) to the err.txt file. |
| cat err.txt | This command prints the contents of the err.txt file to standard output. |
| find /etc -name hosts > std.out 2> std.err | This command searches for a file named "hosts" in the /etc directory. Search results will be redirected to std.out, and any error messages encountered during the search will be redirected to std.err. The > option indicates redirection of standard output (stdout), and the 2> option redirects the standard error stream (stderr). |
| cat std.err | This command prints the contents of std.err to standard output. |
| cat std.out | This command prints the contents of std.out to standard output. |
| find /etc -name hosts > find.out 2>&1 | This command searches for a file named "hosts" in the /etc directory. The search results (if a "hosts" file is found) will be redirected to the find.out file. The important part is the 2>&1 part, which redirects the standard error stream (stderr) to the standard output (stdout). This means that if any errors occur during the search, they will also be written to the find.out file. |
| cat find.out | This command outputs the contents of the find.out file to standard output. This content may contain the search results (if the "hosts" file is found) and any error messages encountered when running the find command. You can view both search results and error messages in this file. |
| tr a-z A-Z | The command indicates the conversion of all letters from lowercase (a-z) to uppercase (A-Z). |
| tr A-Z a-z > myfile | This command converts all uppercase letters (A-Z) in the content coming from standard input to lowercase letters (a-z) and writes this changed text to a file named "myfile". |
| cat myfile | This command prints the contents of the file "myfile" to standard output. |
| ls -l /etc | more | Uses two commands to display information about files and directories in the /etc directory through the more filter program. |
| cut -d: -f1 /etc/passwd | This command uses the cut command to truncate and output the first field from each line of the /etc/passwd file. |
| cut -d: -f1 /etc/passwd | sort | This command uses the cut command to cut , sort the strings fed to it through the pipe. In this case, usernames will be sorted in ascending alphabetical order and output the first field from each line of the /etc/passwd file. |
| cut -d: -f1 /etc/passwd | sort | more | This command uses the cut command to truncate , sorts the lines fed to it through a pipe, outputs large amounts of text page by page, allowing you to scroll through the contents page by page, and outputs the first field of each line in the /etc/passwd file. |
| cat /etc/passwd | Will display the contents of the /etc/passwd file on the screen. |
| head /etc/passwd | Outputs the first few lines from the /etc/passwd file. |
| tail /etc/passwd | Outputs the last few lines from the /etc/passwd file. Typically, the /etc/passwd file contains information about system users, and the latest entries reflect new users that have been added to the system. |
| head -2 /etc/passwd | Outputs the first two lines from the /etc/passwd file. |
| ls /etc | tail -5 | Lists the last five items (files or directories) in the /etc directory. |
| head -n -20 /etc/passwd | Displays lines 1-7, excluding the last twenty lines. |
| cd /etc | You become active in the /etc directory. |
| grep sshd passwd | Will print the entire line containing the match. |
| grep root passwd | Finds all lines in a file named "passwd" that contain the word "root". |
| grep '^root' passwd | Uses grep to find lines in a file named "passwd" that begin with the word "root" (since the "^" character marks the beginning of a line). |
| grep 'sync' passwd | Uses grep to find lines in a file named "passwd" that contain the word "sync". |
| grep 'sync$' passwd | Uses grep to find lines in a file named "passwd" that end with the word "sync" (since the "$" character marks the end of a line). |
| grep '.y' passwd | Uses grep to find lines in a file named "passwd" that contain any character followed by "y". |
| grep 'sshd|root|operator' passwd | Uses grep to find lines in a file named "passwd" where the matching lines contain one of three words: "sshd", "root", or "operator". |
| grep -E 'sshd|root|operator' passwd | Uses grep with extended regular expression mode (-E) to search for lines in a file named "passwd" that contain one of three words: "sshd", "root", or "operator". |
| egrep 'no(b|n)' passwd | Uses egrep with extended regular expression mode to find lines in the "passwd" file where a word that begins with "no" and can be followed by either "b" or "n" occurs. |
| head passwd | grep '[0-9]' | This command starts by outputting the first few lines from the "passwd" file using the head command. The output strings are then filtered through the grep command using the regular expression [0-9], which looks for any string that contains at least one digit in the "passwd" file. |
| grep -E '[0-9]{3}' passwd | Uses grep with extended regular expression mode to find lines in the "passwd" file where there are sequences of three digits in a row (total of 3 digits). |

\*\*\***Скріншоти** виконання команд в терміналі можна **не представляти**, достатньо **коротко описати команди в таблиці**.

***Готував матеріал студент Чех І.В.***

1. Ознайомтесь з командою tar та за її допомогою виконати у терміналі наступні дії:

* створити файл з розширенням .tar;
* створити файл з розширенням .tar, що складається з декількох файлів і каталогів одночасно;
* перегляду вмісту файлу;
* витягти вміст файлу tar;

Зображення, що містить текст, знімок екрана, Шрифт, меню

Автоматично згенерований опис

* створити архівний файл tar, стиснений за допомогою bzip;
* витягти вміст файлу tar bzip;

Зображення, що містить текст, знімок екрана, Шрифт, чорний

Автоматично згенерований опис

* створити архівний tar файл, стисненого за допомогою gzip;
* витягти вміст файлу tar gzip.

Зображення, що містить текст, знімок екрана, чорний, Шрифт

Автоматично згенерований опис

***Готував матеріал студент Дзизиль Д.Є.***

1. Як буде відбуватись перенаправлення потоків виведення в bash для наступних дій з командами (позначено як cmd) та файлами (позначено як file):

|  |  |
| --- | --- |
| **Команда** | **Що виконує команда?** |
| cmd 1> file | Output stdout of the cmd command to the file file. |
| cmd > file | Output stdout of the cmd command to the file file. |
| cmd 2> file | Stderr output of the cmd command to the file file. |
| cmd >> file | Adding the stdout of the cmd command to the contents of file. |
| cmd &> file | Output both stdout and stderr of the cmd command to the file file. |
| cmd > file 2>&1 | Output both stdout and stderr of the cmd command to the file file. 2>&1 redirects stderr to stdout. |
| cmd >> file 2>&1 | Adding both stdout and stderr of the cmd command to the contents of the file file (compensation). 2>&1 redirects stderr to stdout. |
| cmd 2>&1 > /dev/null | cmd stdout output to /dev/null (discard). |
| cmd 2> /dev/null | stderr output of the cmd command to /dev/null (discard). |
| cmd1 | cmd2 | Redirection of cmd1 command stdout to cmd2 command stdin. |
| cmd1 2>&1 | cmd2 | Redirection of both stdout and stderr of cmd1 to stdin of cmd2 (pipelining commands). |

***Готував матеріал студент Чех І.В.***

1. Розгляньте наведені нижче приклади та поясніть, що виконують дані команди та який тип перенаправлення потоків вони використовують:

|  |  |  |
| --- | --- | --- |
| Команда  (контейнер команд) | Що виконує команда? | Який потік перенаправлення? |
| $echo "It is a new story." > story | Uses stdout redirection to write the text string "It is a new story." to the "story" file. | Redirection of stdout |
| $ date > date.txt | Uses stdout redirection to write the current date and time to the "date.txt" file. | Redirection of stdout |
| $ cat file1 file2 file3 > bigfile | Using stdout redirection to merge the contents of files "file1", "file2" and "file3" into a file "bigfile". | Redirection of stdout |
| $ls -l >> directory | Uses stdout redirection to append the output of the "ls -l" command to the contents of the "directory" file. | Redirection of stdout |
| $ sort < file1\_unsorted > file2\_sorted | The stdin redirect is used to feed the contents of "file1\_unsorted" to the sort command, and the result (sorted contents) is written to the file "file2\_sorted" using the stdout redirect. | Redirection of stdin and stdout |
| $ find -name '\*.txt' > file.txt 2> /dev/null | Uses stdout redirection to write search results for files with extension ".txt" to file "file.txt". However, redirecting stderr to "/dev/null" discards all error messages. | Redirection of stdin and stdout |
| $ cat file1\_unsorted | sort > file2\_sorted | Pipeline redirection (|) is used to pass the contents of "file1\_unsorted" to the input of the sort command, the sorting result is written to file "file2\_sorted" via stdout redirection. | Pipeline redirection |
| $ cat myfile | grep student | wc -l | Sequential pipelining is used to do the following: output the contents of the file "myfile" (cat myfile) -> search for lines with the word "student" (grep student) -> count the number of lines (wc -l). | Pipeline redirection |

**Контрольні запитання *Готував матеріал студент Чех І.В.*:**

1. Надайте порівняльну характеристику процесам стискання та архівування.

Compression is a method of reducing data size by eliminating redundant information or by using various encoding techniques without losing data.

Archiving is the process of compressing data for future storage or transmission, which includes data compression, and the creation of archive directories and structures.

* Goal:

Compression aims to reduce data size, thus saving disk space or speeding up data transfer over the network.

Archiving aims to condense data for easy storage and future use.

* Content:

Compression includes various techniques such as removing duplicates, reducing precision (if necessary), selecting data compression algorithms such as ZIP, JPEG, MP3, and others.

Archiving includes compression but can also involve creating indexes, metadata, and directories for easy access to archived data.

* Data Loss:

Compression can lead to quality loss, such as loss of detail in compressed images or audio.

Archiving aims to minimize data loss, and archives keep data unchanged at all times.

* Using:

Compression is used to optimize data storage and transfer, often in real-time.

Archiving is used for old or frequently-used data, or important data for later recovery.

1. Які програми, окрім наведених в роботі, можуть використовуватись для стискання та архівування файлів та каталогів в ОС Linux? Наведіть приклади та їх короткий опис.

* Rar:

Description: Rar is another compression and archiving format used to create compressed archives. In Linux, there are programs like rar and unrar that allow you to work with the Rar format.

Example of use: rar a archive.rar /folder/to/archive (create a Rar archive) or unrar x archive.rar (unzip a Rar archive).

* Zstandard (Zstd):

Description: Zstd is a fast compression algorithm that provides competitive compression and speed. It is ideal for modern systems and is widely used in Linux.

Example usage: zstd -19 filename.txt (compress the file) or zstd -d filename.txt.zst (unzip the compressed file).

* lzip:

Description: Lzip is a data compression program that uses the LZMA algorithm. It strives to ensure high data compression and reliability.

Example usage: lzip filename.txt (compress the file) or lzip -d filename.txt.lz (unzip the compressed file).

* Cpio:

Description: Cpio is a utility that allows you to create archives and extract files from them. It is often used in scripts and command mode to copy files and directories.

Example of use: find /folder/for/archive | cpio -o > archive.cpio (create Cpio archive) or cpio -i < archive.cpio (unzip Cpio archive).

1. Порівняйте алгоритми стискання, що використовуються в командах (програмах), використовуваних в Linux. Які з алгоритмів можна вважати найшвидшим та найефективнішим?

* XZ:

Compression algorithm: XZ uses the LZMA2 compression algorithm.

Performance: XZ provides high robustness and is considered better than other algorithms such as gzip and bzip2.

Speed: XZ is usually slower for gzip, but faster for bzip2.

* ZIP Code:

Compression algorithm: ZIP Code uses the Deflate compression algorithm, which is also used in gzip.

Performance: ZIP Code provides good robustness for many data types, but is always slightly worse than XZ or bzip2.

Speed: ZIP Code has high compression and decompression speeds similar to gzip.

* Bzip2:

Compression algorithm: Bzip2 uses the Burroughs-Wheeler algorithm.

Performance: Bzip2 provides good stability, especially for text data.

Speed: Bzip2 can be slower for gzip and zip, but can be fast for XZ.

* Gzip:

Compression algorithm: Gzip uses the DEFLATE algorithm.

Performance: Gzip provides good robustness for many data types.

Speed: Gzip has high compression and decompression speeds.

The compression quality and speed of the algorithms depends on the specific data being compressed and the chosen algorithm. In general, XZ provides the best stability and gzip and Zip are generally fast and acceptable for many tasks. Bzip2 also compresses text data well but can be slower.

1. Опишіть програмні засоби для стискання та архівування, що можуть бути використані у вашому мобільному телефоні.

* ZArchiver:

Platform: Available for Android.

Description: ZArchiver is a powerful and free application for compressing and archiving files on Android devices. It supports many formats including ZIP, RAR, 7z, TAR and others. The application allows you to create, unpack and edit archives.

* WinZip:

Platform: Available for both Android and iOS.

Description: WinZip is a popular file archiving and compression tool that is also available on mobile devices. It supports many formats including ZIP, RAR, 7z, TAR and others. WinZip allows you to create password protected archives.

* RAR:

Platform: Available for both Android and iOS.

Description: RAR is an application for creating and decompressing RAR archives on mobile devices. It also supports other formats like ZIP and TAR. RAR allows you to create archives with a password and use the function in case of recovery of archive damage.

1. Опишіть та порівняйте програмні засоби для стискання та (де)архівування даних у ОС сімейства Windows.

* WinZip:

WinZip is a highly popular software tool used for compression and archiving on Windows OS. It supports many file formats like ZIP, RAR, 7z, and more. The software is user-friendly and offers additional features like password protection, encryption, and cloud storage.

* 7-Zip:

7-Zip is an open-source and free compression and archiving software tool. It supports many file formats like 7z, ZIP, TAR, and more. 7-Zip is known for its high compression rate and speed.

* WinRAR:

WinRAR is a software tool that supports the RAR format and many other formats, including ZIP. It is used for compressing and archiving files and offers high compression rates. WinRAR also allows you to create password-protected archives.

* Windows Built-in Tools:

The Windows operating system comes with built-in tools for compressing and archiving files. These tools are accessible through Windows Explorer and can create ZIP archives and extract them without the need for additional software. However, these tools have limited capabilities when compared to specialized programs.

* PeaZip:

PeaZip is a free and open-source archiver that supports many formats, including 7z, ZIP, RAR, and more. It has both graphical and command modes and allows you to create, extract, and manage archives.

* Comparison:

While WinZip, WinRAR, and 7-Zip are popular programs with many features and capabilities, they usually cost money or have a limited free version. 7-Zip is known for its high compression rate and is free to use. Built-in Windows tools can be useful for simple file compression and archiving, but they have limited format support. PeaZip is an open-source program that offers flexibility in use and supports many formats.

1. Поясніть яким чином стиснення та архівування даних може бути використано для резервування даних. В яких ще задачах системного адміністрування воно може бути використано.

Data compression and archiving is used for various purposes in system administration and data management. Here's how these techniques can be used for data backup and other system administration tasks:

* Data Backup:

Data compression and archiving enable you to reduce the size of backup copies that you store on external media, cloud storage or other servers. This helps you save time and bandwidth required to store and restore data.

* Saving Disk Space:

Archiving and compressing files helps you save disk space, especially for old or rarely used files. This is particularly important on servers and large computing systems with limited storage options.

* Data Sharing:

Compressed and archived files can be easily moved and shared between different systems and users. This provides a convenient way to share data.

* Storage of Archival Copies:

Data compression and archiving are crucial for keeping archival copies of important data and documents that may be needed in the future to meet legal requirements or for audit purposes.

1. Яке призначення директорії файлу /dev/null?

The /dev/null directory on Unix and Unix-like operating systems such as Linux has a special purpose. In fact, /dev/null is a special file, not a directory, and its main purpose is to reject data that is output or written to it. In other words, it's a "dump" for data you no longer want to store or process.

Висновок:

Протягом виконання даної лабораторної роботи ми дізналися про різницю між архівуванням і стисненням, навчилися стискати файли і архівувати їх за допомогою tar, xz, zip, bzip, gzip.