**"КИЇВСЬКИЙ ФАХОВИЙ КОЛЕДЖ ЗВ’ЯЗКУ"**

**Лабораторна робота 5**

з дисципліни «Операційні системи»

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

Виконали: студенти **3** курсу, групи **КСМ-13А**

**Засенко Олександр**

(прізвище та ініціали)

**Дзюбенко Дмитро**

(прізвище та ініціали)

**Сторожук Костянтин**

(прізвище та ініціали)

Київ  2023

**ЗМІСТ**

1 …………………………………………………………………………………...

2 ……………………………….

Conclusions……………………………………………………..16

**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

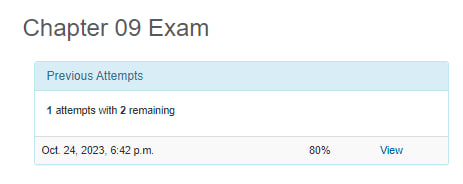
**Завдання для попередньої підготовки**

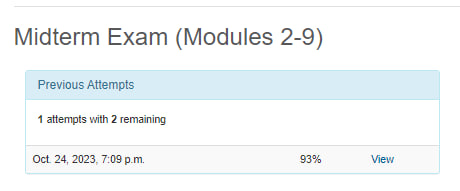
***Готував матеріал студент Zasenko***

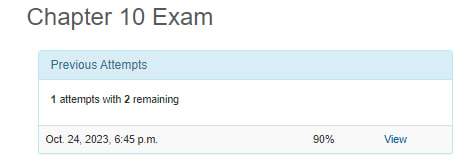
Невеликий словник базових англійських термінів з питань призначення команд та їх параметрів.

|  |  |
| --- | --- |
| Command | Usage |
| bzip2 afile | This will compress the file and give it the name &quot;afile.bz2&quot;. |
| bzip2 -s afile | While bzip2 implements numbered flags, they mean something somewhat different than they do with gzip. |
| bzip2 -1 file | To decompress a bzip compressed file, you can pass the -d flag: |
| bzip2 -d file.bz2 | This will give back an uncompressed file called &quot;file&quot;. |
| xz file | This will process the file and produce a file called &quot;file.xz&quot;. |
| xz -c test &gt; test.xz | For the numbered flags, xz uses the lower numbers to indicate faster compression. |
| xz -e -9 large\_file | This will take a long time and in the end, may not show very significant gains, but if you need that  functionality, the option is available. |

**Пройдіть тестування у курсі NDG Linux Essentials за такими темами:**







На базі розглянутого матеріалу я дізнався:

1. tar:

* Призначення: tar (Tape Archive) використовується для створення архівів, які можуть містити кілька файлів або каталогів.

Основні параметри:

* -c: Створює архів.
* -x: Розпаковує архів.
* -v: Виводить детальний вихід під час роботи.
* -f file: Вказує ім'я файлу архіву.

Встановити tar можна зазвичай за допомогою пакунків встановлення вашого дистрибутиву Linux (наприклад, apt-get install tar для системи Debian/Ubuntu).

1. xz:

* Призначення: xz - це утиліта стиснення файлів, яка використовує алгоритм стиснення LZMA2.

Основні параметри:

* -z: Стиснути файл (створити файл з розширенням .xz).
* -d: Розпакувати файл (розархівувати файл .xz).

Для встановлення xz використовуйте пакунковий менеджер вашої операційної системи.

1. zip:

* Призначення: zip - утиліта для створення та розпакування архівів в форматі ZIP.

Основні параметри:

* -r: Рекурсивне створення архіву для каталогів.
* -u: Оновлення архіву, додаючи нові файли.
* -d: Видалити файли з архіву.

Для встановлення zip використовуйте пакунковий менеджер вашої операційної системи.

1. bzip2:

* Призначення: bzip2 - це утиліта для стиснення файлів за допомогою алгоритму BZip2.

Основні параметри:

* -c: Вивести стиснений вміст на стандартний вивід.
* -d: Розпакувати файл BZip2.

Для встановлення bzip2 використовуйте пакунковий менеджер вашої операційної системи.

1. gzip:

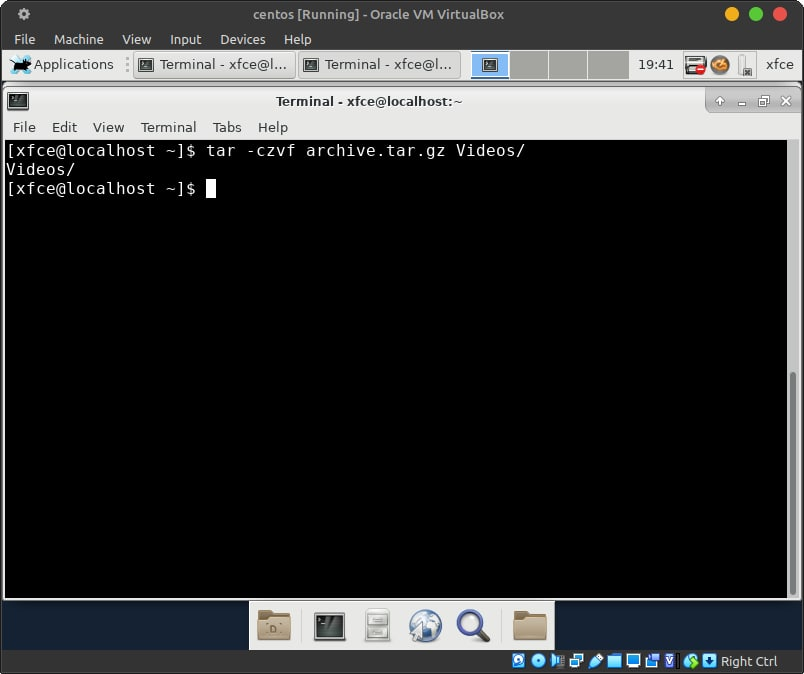
* Призначення: gzip - утиліта стиснення файлів за допомогою алгоритму GZip.

Основні параметри:

* -d: Розпакувати файл GZip.
* -c: Вивести стиснений вміст на стандартний вивід.

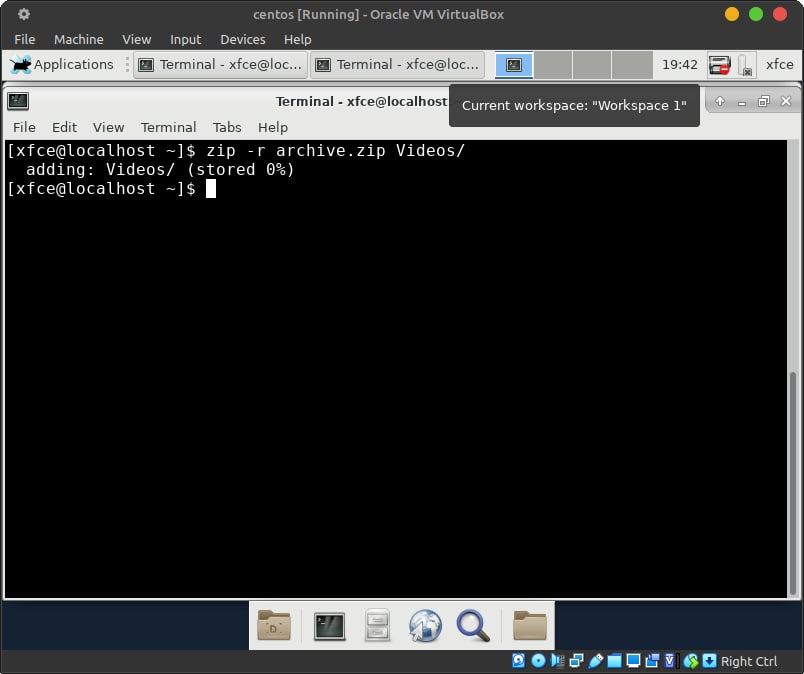
Для встановлення gzip використовуйте пакунковий менеджер вашої операційної системи.

1. Використання tar та gzip для створення стисненого архіву:



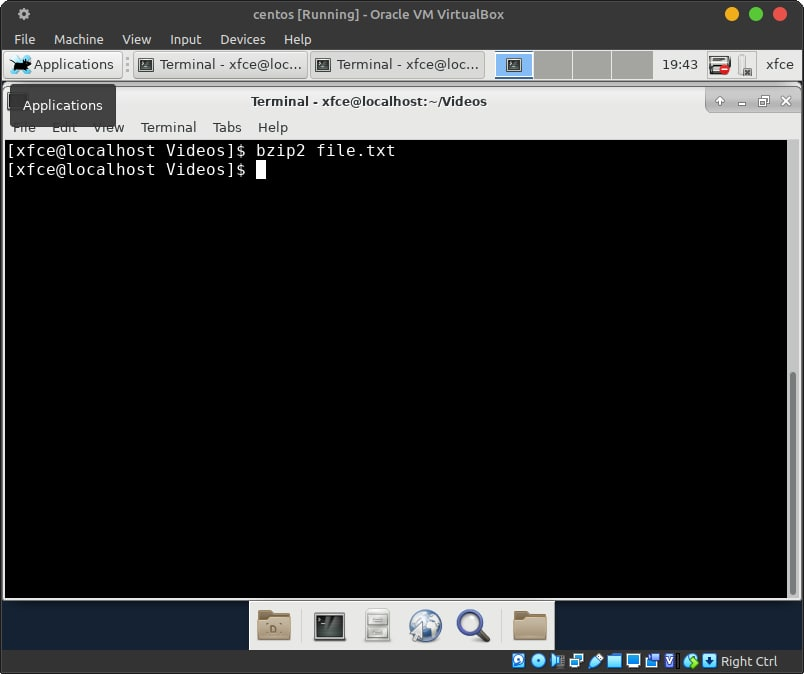
У цьому прикладі tar створює архів (archive.tar) з вмісту Videos/, а gzip стискає цей архів, створюючи файл archive.tar.gz.

1. Використання zip для створення ZIP-архіву:



За допомогою команди zip, ви можете створити ZIP-архів (archive.zip), який містить вміст Videos/. Ключ -r означає рекурсивне включення файлів та каталогів у ZIP-архів.

1. Використання bzip2 для стискання файлу:



Команда bzip2 стискає файл file.txt, створюючи стиснутий файл file.txt

1. cat:

* Призначення: cat (від "concatenate") використовується для відображення вмісту файлів та конкатенації їх разом.

Основні параметри:

* Вивести вміст одного або декількох файлів: cat file1.txt file2.txt
* Конкатенувати файли: cat file1.txt file2.txt > combined.txt

cat зазвичай встановлено на більшості дистрибутивів Linux за замовчуванням.

1. less:

* Призначення: less - це програма для перегляду текстових файлів, яка дозволяє прокручувати вміст вгору та вниз та виконувати пошук.

Основні параметри:

* Відкрити файл для перегляду: less file.txt
* Прокручувати текст вниз (клавіша "Space") та вгору (клавіша "B").
* Вийти з програми (клавіша "q").

less зазвичай встановлено на більшості дистрибутивів Linux за замовчуванням.

1. more:

* Призначення: more - це інший переглядач текстових файлів, який також дозволяє прокручувати вміст вгору та вниз.

Основні параметри:

* Відкрити файл для перегляду: more file.txt
* Прокручувати текст вниз (клавіша "Space") та вгору (клавіша "Enter").
* Вийти з програми (клавіша "q").

more також встановлено на більшості дистрибутивів Linux за замовчуванням.

1. head:

* Призначення: head використовується для виведення перших кількох рядків текстового файлу.

Основні параметри:

* Вивести перші N рядків файлу: head -n N file.txt
* Вивести перші 10 рядків за замовчуванням: head file.txt

head також встановлено на більшості дистрибутивів Linux за замовчуванням.

1. tail:

* Призначення: tail використовується для виведення останніх кількох рядків текстового файлу.

Основні параметри:

* Вивести останні N рядків файлу: tail -n N file.txt
* Вивести останні 10 рядків за замовчуванням: tail file.txt

tail також встановлено на більшості дистрибутивів Linux за замовчуванням.

Командна оболонка (наприклад, Bash в середовищі Linux) працює з каналами, потоками та фільтрами для обробки та обміну даними. Основні принципи роботи такі:

1. Потоки (Streams):

* STDIN (стандартний ввід): Це вхідний потік, через який команда може приймати дані. Зазвичай використовується клавіша Enter або редирект від файлу.
* STDOUT (стандартний вивід): Це вихідний потік, через який команда виводить результати своєї роботи. Зазвичай це виведено на термінал або може бути перенаправлено до файлу.
* STDERR (стандартна помилка): Це також вихідний потік, але для виведення помилок та повідомлень про помилки. Зазвичай це виводиться на термінал, але може бути перенаправлено для логування.

1. Канали (Pipes):

* Канали дозволяють об'єднувати роботу кількох команд разом, передаючи вихідний потік однієї команди як вхідний потік іншої команди.
* Синтаксис для каналу це вертикальна черта |.

1. Фільтри:

* Фільтри - це команди, які обробляють та змінюють дані, передавані через потоки. Вони можуть виконувати фільтрацію, заміну, сортування, обрізку тощо.
* Фільтри часто використовуються разом з каналами для складних операцій обробки даних.

Команда grep призначена для пошуку та виведення текстових рядків, які відповідають певному шаблону або регулярному виразу у вихідному тексті. Основні використання grep включають:

1. Пошук тексту в файлі або потоці даних: grep дозволяє здійснювати пошук певного текстового рядка або шаблону у вмісті текстового файлу або потоку даних, наприклад, якщо ви шукаєте певне слово або фразу у лог-файлі або документі.
2. Використання регулярних виразів: grep підтримує використання регулярних виразів для більш складних пошуків, таких як пошук всіх рядків, які відповідають певному шаблону, наприклад, всі числа чи email-адреси в текстовому файлі.
3. **Виведення відповідних рядків**: grep виводить всі рядки, які відповідають шаблону або регулярному виразу, на стандартний вивід. Ви можете також перенаправити результат до файлу або використовувати його у скриптах та інших командах.
4. **Пошук у багатьох файлах**: grep може застосовуватися до декількох файлів або включати опції для рекурсивного пошуку в каталогах.

**Хід роботи  
*Готував матеріал студент Dziubenko***

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.

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. Створіть таблицю для опису цих команд\*\*\*

|  |  |
| --- | --- |
| Назва команди | Її призначення та функціональність |
| mkdir mybackups | Creating a new directory "mybackups" in the user's home directory |
| tar -cvf mybackups/udev.tar | The command "tar" is used to combine multiple files into one file. In this case, |
| /etc/udev | the contents of the "/etc/udev" directory will be saved in an archive called "udev.tar" in the "mybackups" directory. The "-c" option instructs the "tar" command to create a tar file. The "-v" option stands for "verbose," which tells the "tar" command to show what it is doing. The "-f" option is used to specify the name of the tar file. |
| ls mybackups | List the contents of the "mybackups" directory. |
| cp /usr/share/dict/words | Copy the file "words" from "/usr/share/dict" to the current directory. |
| gzip words | Compress the file "words" using gzip, creating "words.gz." |
| gunzip words.gz | Decompress the file "words.gz" to restore "words." |
| bzip2 words | Compress the file "words" using bzip2, creating "words.bz2." |
| bunzip2 words.bz2 | Decompress the file "words.bz2" to restore "words." |
| xz words | Compress the file "words" using xz, creating "words.xz." |
| unxz words.xz | Decompress the file "words.xz" to restore "words." |
| zip words.zip words | Create a ZIP archive named "words.zip" containing the "words" file. |
| unzip -l udev.zip | List the contents of the ZIP archive "udev.zip" without extracting. |
| rm -r etc | Recursively remove the "etc" directory and its contents. |
| echo “Hello World” | Display the text "Hello World." |
| cat mymessage | Display the content of the file "mymessage." |
| find ~ -name “\*bash\*” | Search for files with "bash" in their names within the home directory (~). |
| tr a-z A-Z | Translate lowercase letters to uppercase. |
| cut -d: -f1 /etc/passwd | sort | Extract and sort usernames from the "/etc/passwd" file using ":" as the delimiter. |
| more /etc/passwd | Display the contents of the "/etc/passwd" file one page at a time. |
| Less /etc/passwd | Display the contents of the "/etc/passwd" file in a pager for reading. |
| head /etc/passwd | Display the first few lines of the "/etc/passwd" file. |
| tail /etc/passwd | Display the last few lines of the "/etc/passwd" file. |
| grep sshd passwd | Search for the string "sshd" in the "passwd" file. |
| egrep | This is a command that would typically be followed by a regular expression for pattern matching. It's not specified here. |

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

- створити файл з розширенням .tar;

tar -cvf file.tar file1 file2

- створити файл з розширенням .tar, що складається з декількох файлів і каталогів одночасно;

tar -cvf file.ter file1 directory1

- перегляду вмісту файлу;

tar -tvf file.tar

- витягти вміст файлу tar;

tar -xvf file.tar

- створити архівний файл tar, стиснений за допомогою bzip;

tar -cvjf file.tar.bz2 file1 file2

- витягти вміст файлу tar bzip;

tar -xvjf file.tar.bz2

- створити архівний tar файл, стисненого за допомогою gzip;

tar -cvzf file.tar.gz file1 file2

- витягти вміст файлу tar gzip.

tar -xvzf file.tar.gz

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&gt; file | Виведення stdout команди cmd у файл file |
| cmd &gt; file | Те ж, що і попередня дія, виведення stdout у файл file |
| cmd 2&gt; file | Виведення stderr команди cmd у файл file. |
| cmd &gt;&gt; file | Додавання stdout команди cmd до вмісту файлу file (замість перезапису). |
| cmd &amp;&gt; file | Виведення як stdout, так і stderr команди cmd у файл file. |
| cmd &gt; file 2&gt;&amp;1 | иведення stdout команди cmd у файл file, а також stderr (2) в той же файл, що і stdout (1). |
| cmd &gt;&gt; file 2&gt;&amp;1 | Додавання як stdout, так і stderr команди cmd до вмісту файлу file. |
| cmd 2&gt;&amp;1 &gt; /dev/null | Виведення stdout команди cmd в /dev/null (повне відкидання), при цьому stderr залишається. |
| cmd 2&gt; /dev/null | Виведення stderr команди cmd в /dev/null (повне відкидання stderr). |
| cmd1 | cmd2 | Перенаправлення stdout команди cmd1 на вхід команди cmd2 (конвеєр). |
| cmd1 2&gt;&amp;1 | cmd2 | Перенаправлення stdout та stderr команди cmd1 на вхід команди cmd2 (конвеєр), тобто обидва потоки виведення команди cmd1 використовуються командою cmd2 |

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 &quot;It is a new story.&quot; &gt; story | The echo command prints the string "It is a new story." to the standard output (stdout).   * The output of the echo command is redirected to the file story. | It uses the > operator, which redirects stdout to a file. |
| $ date &gt; date.txt | The date command displays the current date and time to stdout.   * The output of the date command is redirected to the file date.txt. | It uses the > operator, which redirects stdout to a file. |
| $ cat file1 file2 file3 &gt; bigfile | The cat command concatenates the contents of files file1, file2, and file3 and outputs them to stdout.   * The output of the cat command is redirected to the file bigfile. | It uses the > operator, which redirects stdout to a file. |
| $ls -l &gt;&gt; directory | The ls -l command lists files and directories in "long listing" format and outputs to stdout.   * The output of the ls -l command is appended to the end of the file directory. | It uses the >> operator, which appends stdout to a file. |
| $ sort &lt; file1\_unsorted &gt; file2\_sorted | The sort command sorts the content of the file file1\_unsorted.   * The content of file1\_unsorted is redirected to the stdin of the sort command. * The sorted result is saved to the file file2\_sorted. | It uses the < operator, which redirects stdin from a file. |
| $ find -name &#39;\*.txt&#39; &gt; file.txt 2&gt; /dev/null | The find command searches for files with the extension .txt in the current directory and its subdirectories.   * The search result (stdout) is redirected to the file file.txt. * The stderr is redirected to /dev/null, effectively discarding error messages. | It uses the > and 2> operators, which redirect stdout and stderr, respectively. |
| $ cat file1\_unsorted | sort &gt; file2\_sorted | The cat command outputs the content of file1\_unsorted to stdout.   * The output of cat is piped (|) to the stdin of the sort command, which sorts the data. | The sorted result is saved to the file file2\_sorted. |
| $ cat myfile | grep student | wc -l | This is a sequence of commands: cat, grep, and wc -l.   * cat outputs the content of myfile to stdout. * The output of cat is piped to the stdin of the grep command, which searches for lines containing the word "student." | The result of grep is then piped to the stdin of the wc -l command, which counts the number of lines. |

Контрольні запитання:

**Conclusions**