National Technical University of Ukraine

“Igor Sikorsky Kyiv Polytechnic Institute”

Faculty of Informatics and Computer Science

Department of Information Systems and Technologies

Laboratory work № __3
from the discipline «LINUX »

Subject: « *File system structure. Ubuntu directory/file management*»

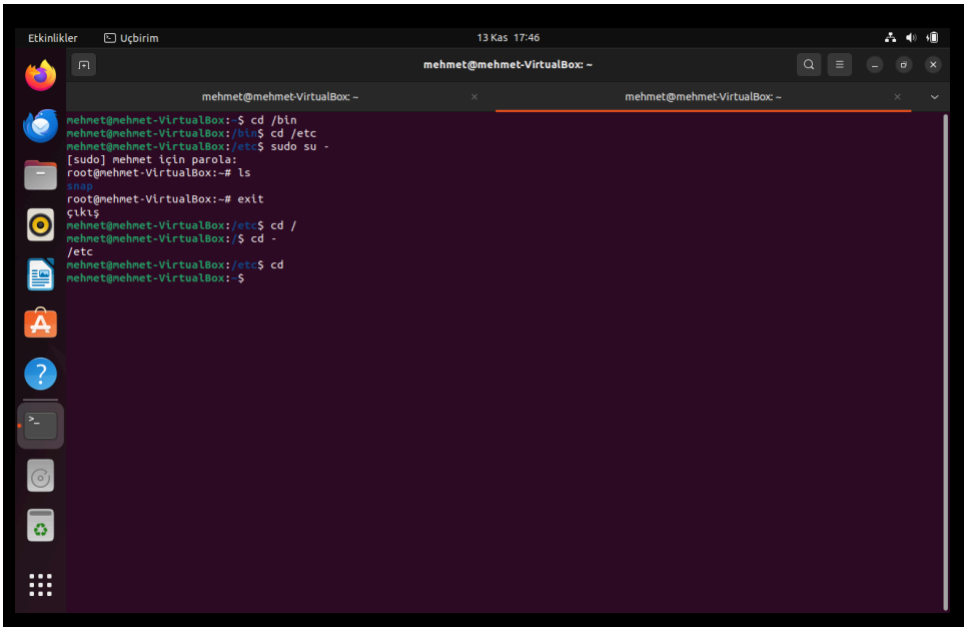
Performed by:
student of group IM-14 FIOT
Full name Mehmet KULUBECİOGLU

Checked::
Senior lecturer of Department ST
Maryna Khmeliuk

KYIV 2023

TASK SOLUTION :

Firstly we change our directory using command ‘cd’ , standard directories like /bin, /etc, /root and home are used. ‘cd - ‘ is used to change current to previous directory.

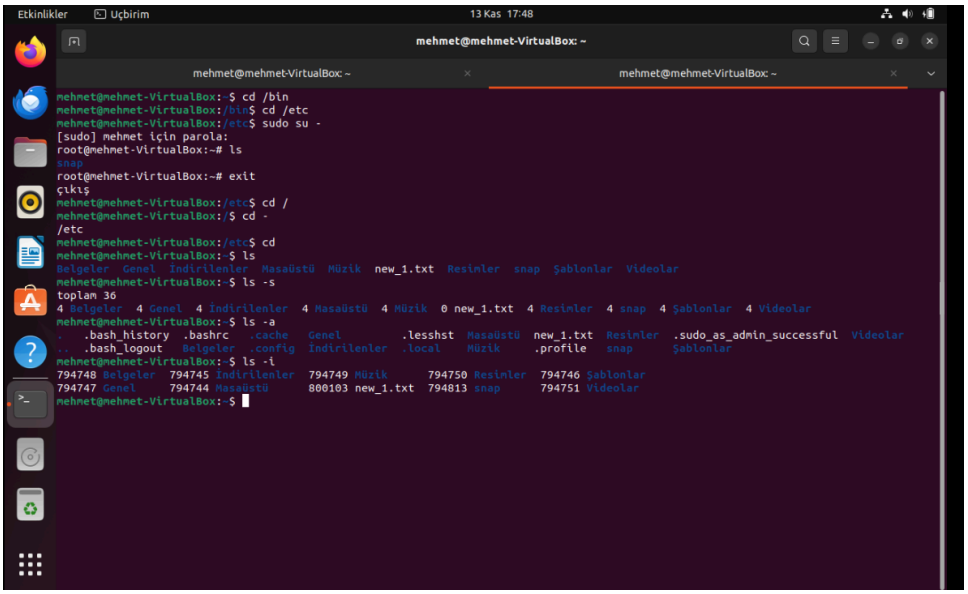


ls - we use ls for listing the contents of the current directory

-s : for order by size

-a : not ignoring dot files

-i : for inode



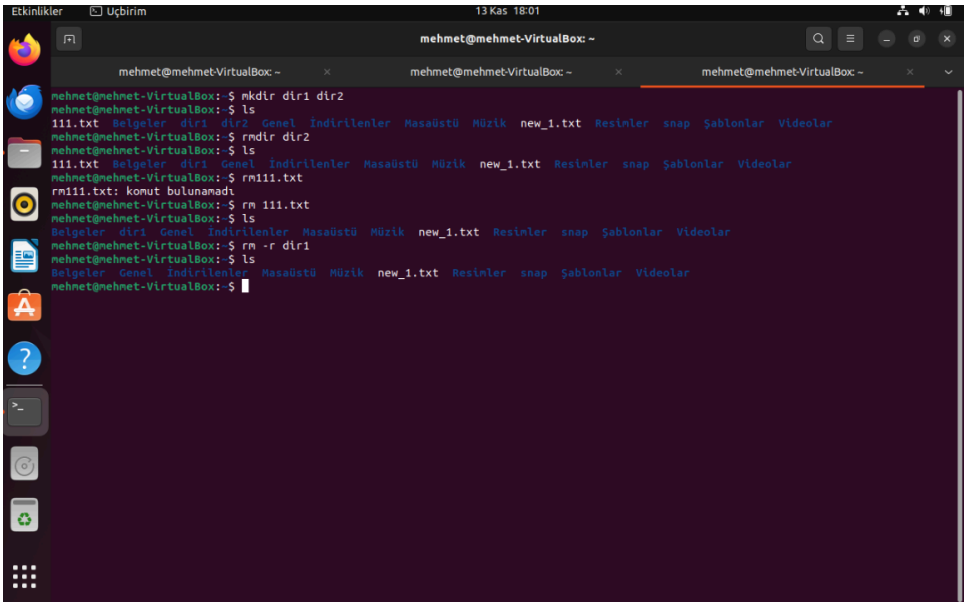
We use the ‘mkdir’ command to create new directories - dir1 and dir2.

We list the contents of our current directory and find our newly created dir1 & dir2.

Then, we remove dir2 using ‘rmdir’ command, list to check.

Accordingly, we remove file ‘111.txt’ using ‘rm’ command.

Now we remove the directory ‘dir1’ using ‘rm’: to remove and ‘-r’: for directory, together.

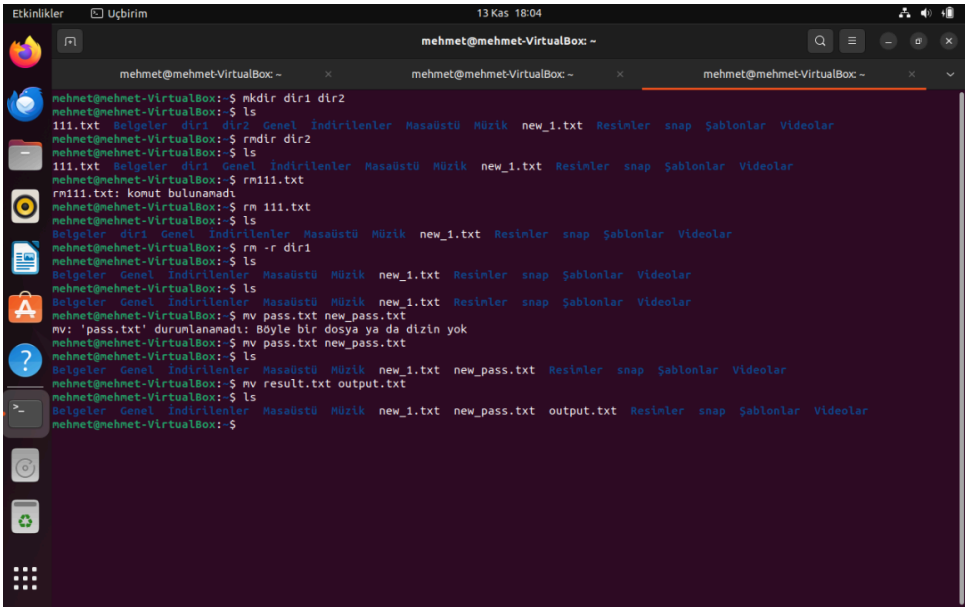


Now we see the uses of ‘mv’ command:

Firstly we list the contents in our current directory- we see pass.txt file

Then using ‘mv’ we rename it to new_pass.txt

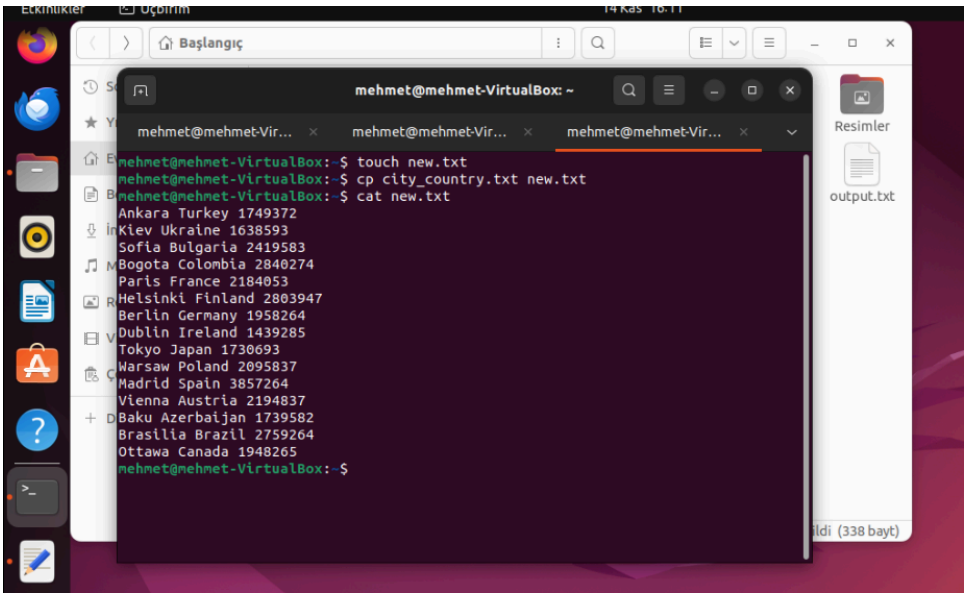
Secondly we move the result.txt file to output.txt, then we list the contents - its moved



Now we use ‘touch’ command to create new_1.txt file

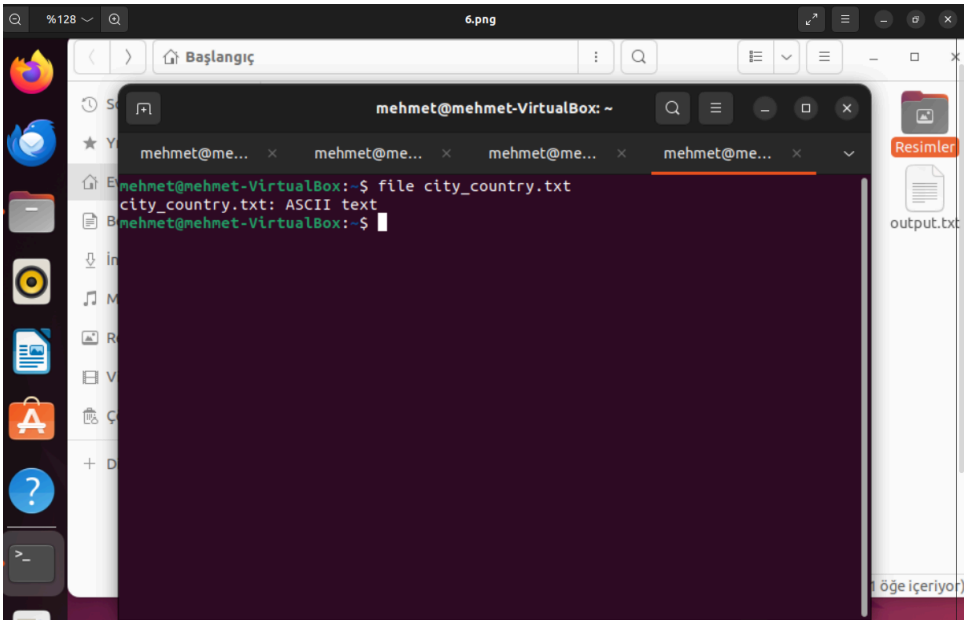
Then we copy the contents of city.txt file to the newly created file using ‘cp’

Using ‘cat’ we display the contents of the new_1.txt file.



Here we use ‘file’ command to check the type of file for ‘city.txt’

The output is shown : it is an ASCII text

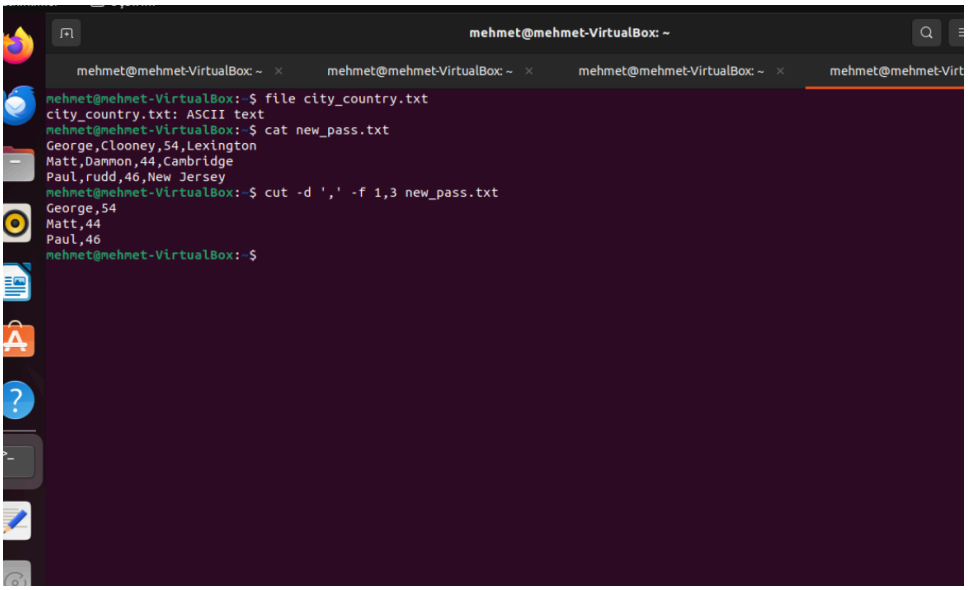


Now we concatenate the contents of new_pass.txt file

Then we use ‘cut’ to separate our required fields or contexts of the file

Here we cut delimiter (-d) comma(,) and filed(-f) first and third (1,3) from the new_pass.txt file

We see the output of the 1st & 3rd fields separated by commas



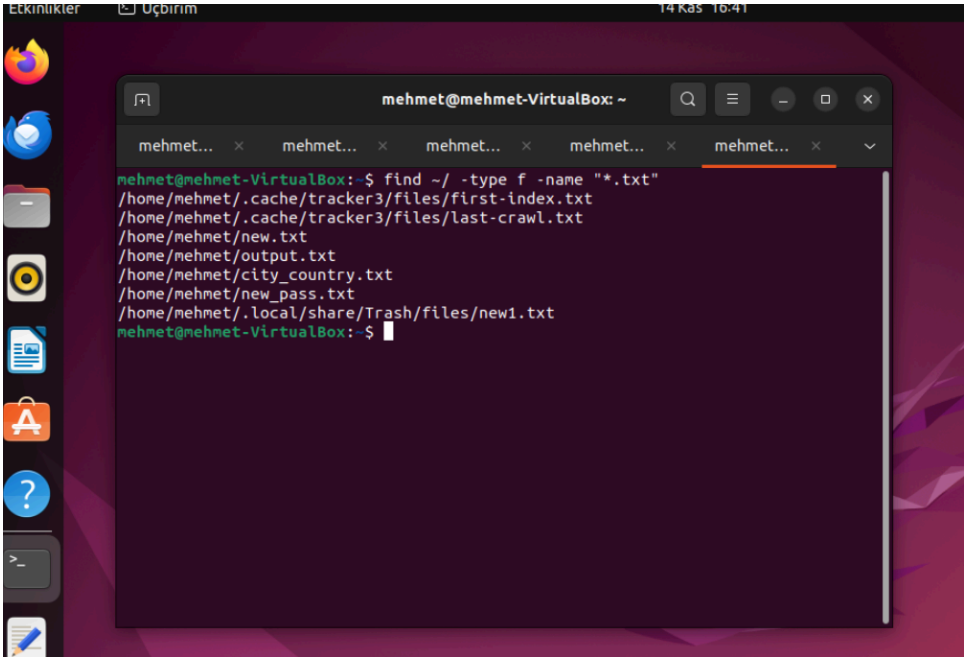
Find - we use this command to find our required files, directories,etc in a

directory hierarchy based on various criteria such as file name, type, size, and more.

~/ : specifies the starting directory for the search, which is your home directory

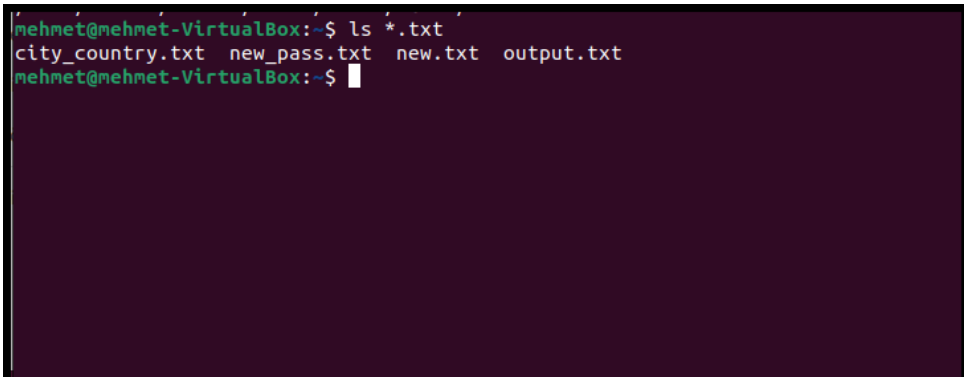
-type -f : specifies the type which is file here

-name "*.txt" specifies that we want to find files with names ending in ".txt."



Now we see an example of file gobbling:

We use ‘*’ to represent any type of characters , numbers, strings, spaces and any number of them before .txt to list all the text files in our current directory



Here's an example of how to create a tar archive (compressed with gzip) with all the necessary files in a directory and then extract them.

Firstly we create a tar file ‘my_archive.tar.gz’ with the files of dir1

-c: for creating new archive

-z: compressing the archive using gzip

-v: verbosely list the files processed

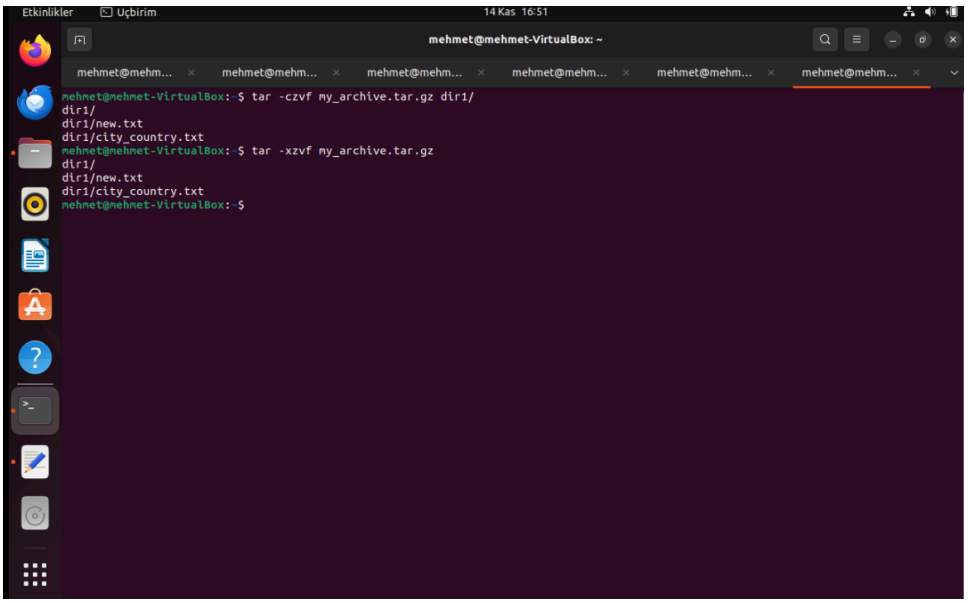
-f: specifying archive file name

Then, we extract the files from our compressed tar archive named ‘my_archive.tar.gz’

-x: for extracting

-z for decompressing

We see the files and the subdirectories extracted.



We use ‘gzip’ command to compress the city.txt file

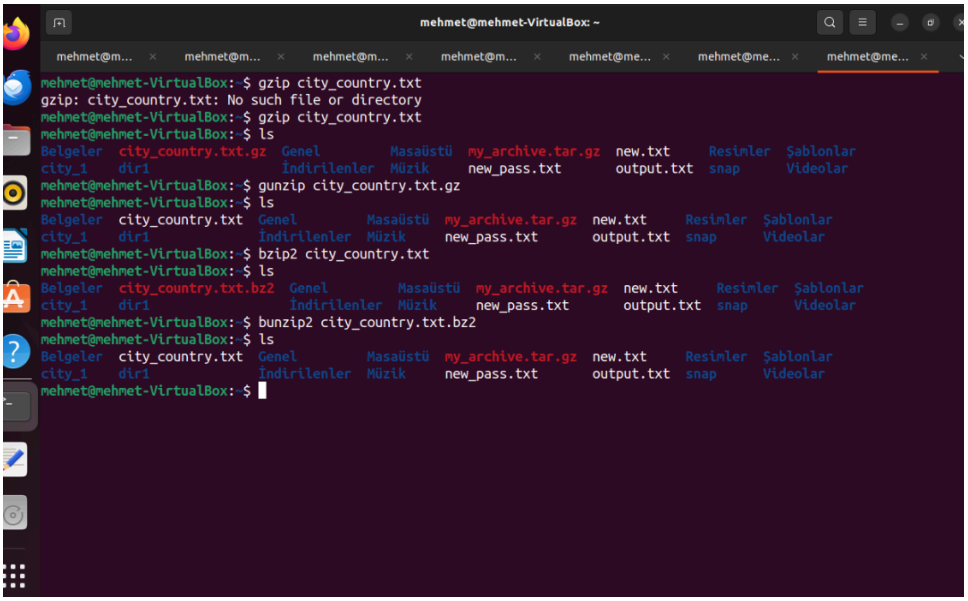
After listing the contents of our current directory ‘city.txt.gz’ - our compressed file is shown.

Then to decompress the file we use command ‘gunzip’, and list the contents to check.

Similarly we use ‘bzip2’ command to compress the city.txt file

After listing the contents of our current directory ‘city.txt.bz2’ - our compressed file is shown

To decompress the file we use ‘bunzip2’ and list the contents to check.



dd - is used for copying and converting files.

Here's a simple example of using dd to create a copy of a file.

Supposedly, we have a file named "city.txt," and we want to create a copy of it named "destination_file.txt." We can use the dd command to achieve this.

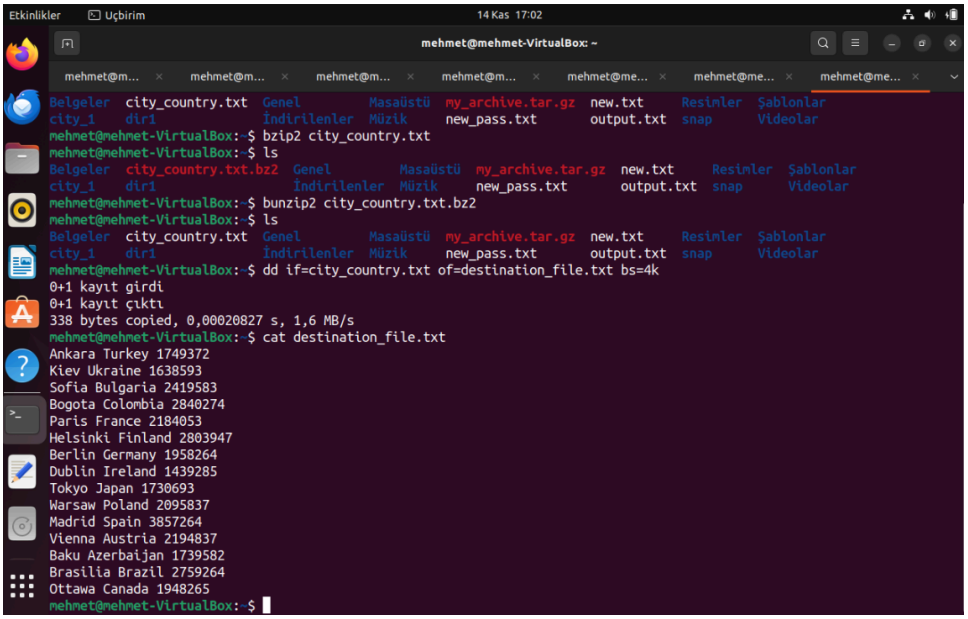
if : input file / source file

of : output file/ destination file

bs : Block size - size of data blocks to be read and written, here we have used 4KB

We see that 320 bytes is copied into our destination_file.txt

To check we concatenate the contents of ‘destination_file.txt’



Conclusion :

I have learned how to navigate the Linux command line using commands like cd, ls, create and manage directories, manipulate files with commands like rm, mv, cp, and work with file content using cat and cut. I also explored file globbing to work with multiple files. Additionally, I have gained knowledge of data compression utilities such as tar, gzip, and “bzip2” for archiving and compressing files, as well as using dd for file copying and conversion. Lastly, I have learned how to use the find command for searching files based on specific criteria. These skills are fundamental for efficient Linux system management and file manipulation.

Specifications : using the Universal Type-2 Hypervisor (UTM) and the Ubuntu 22.03.4 Linux distribution on a virtual machine within UTM.