Put Student Name(s) ↓	Put Student IDs ↓	Due Date	Grade Weight
Ishan Aakash Patel	146151238	As Posted	12%

Name	Project1: Build Your own Forensic Workstation
Main Goal	Set up your functional forensic workstation to conduct forensics investigations using variety of popular tools

Instructions	• It is an Individual assignment. Put your name + Student ID in the empty spaces above.
	• Submit via the BB relevant link ONLY. NO submission via email please. Be sure to submit the final version file ONLY.
	• Show me genuine signs of your work is done on your machine. This includes:
	  ○ Screenshots that show your desktop background with Date/Time
	  ○ Show me a pop-up bx that shows "your name + IP."
	  ○ Show your logged in account, if applicable
	  ○ Optional: Show your photo.
	• Use this same template to include your work in the specified fields below. Submit in PDF.
	• Submit your report name with the name: CYT215-Project1-Student Name & ID

Students Work required for this activity	1. You will follow instructions to setup your own forensics workstation on your machine. 2. You will check that your forensics workstation is functioning. 3. You will use your workstation for memory & malware analysis. 4. You will Prepare Your Target System (your own machine). You will Build Your basic Lab.
How to start	• Read thoroughly & follow the instructions mentioned in this link. The instructions will guide you to a step-by-step of how you complete your work successfully https://bluecapesecurity.com/build-your-forensic-workstation/ • Take an image of your machine memory. Use this link for guidance: https://dfirmadness.com/case-001-memory-analysis/ -- UPDATE INSTEAD ANALYSIS the MEMORY DUMP provided using Volatility. • You can use any installed tool or focus on common tools for memory analysis e.g.: Volatility; Cyber Triage; Rekall; Redline;
Important	• Your target system should be your own machine
Students Reports	1. Take screenshots of all your works steps 2. Show you have practiced 3 relevant tools to investigate memory. 3. For Memory analysis, anaylsis the Memory dump provided and solve the challenges - https://github.com/stuxnet999/MemLabs/tree/master/Lab%203 – Follow this guide: https://oviche.github.io/2023/10/MemLabs3/ 4. Write a detailed report of personal learning experience (free writing).
Grading Rubrics	• 5 Marks for completing the setup of your forensics workstation successfully. • 5 Marks for using your workstation for memory analysis. You can use Volatility to solve the challenge - https://github.com/stuxnet999/MemLabs/tree/master/Lab%203 • 2 Marks for your detailed personal learning experience (free writing)
	• If you do NOT use this template or delete any part of it or use any other template, you will be degraded.

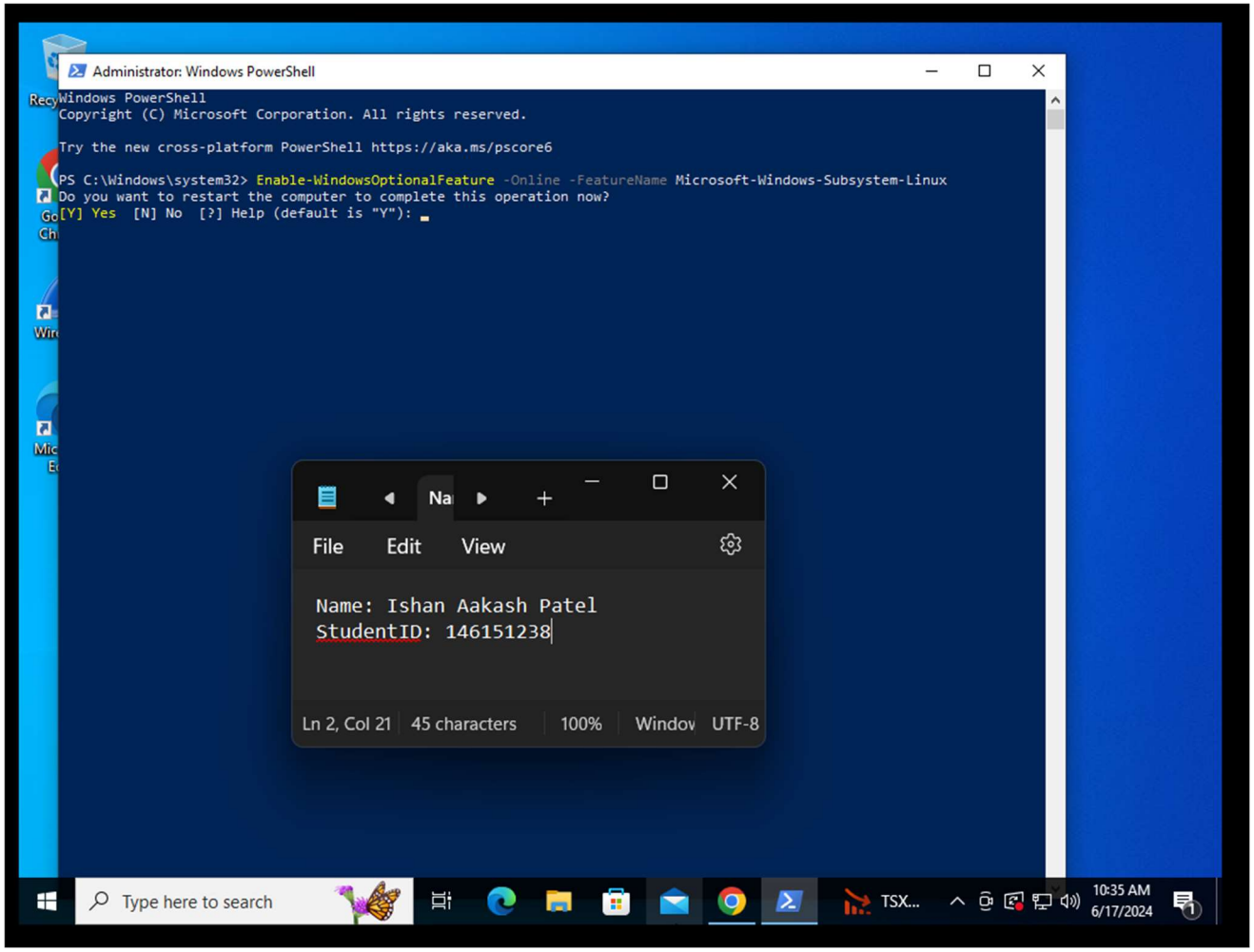
Grading Alerts	• If you do NOT follow the fie naming convention, you will be degraded. • If you do NOT submit your file in PDF; you will be degraded. • If you do NOT show your account real name (when applicable); you will be degraded. • If you do NOT show your machine desktop background (with date & time) and IP, you will be degraded. • If you do NOT write (in your own words) your learning experience for the activity practices, you will be degraded.

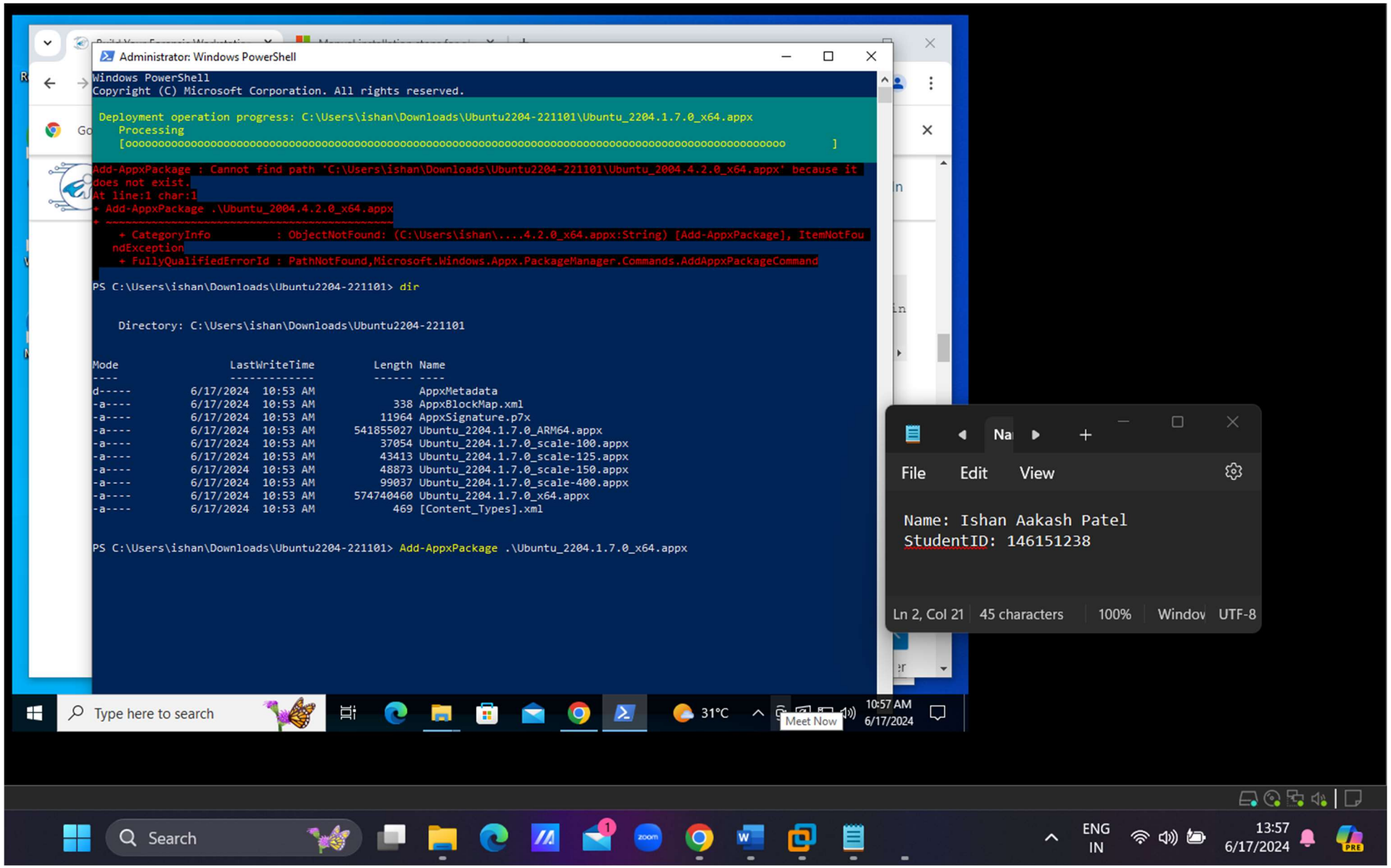
Part 1 – Building my own Workstation

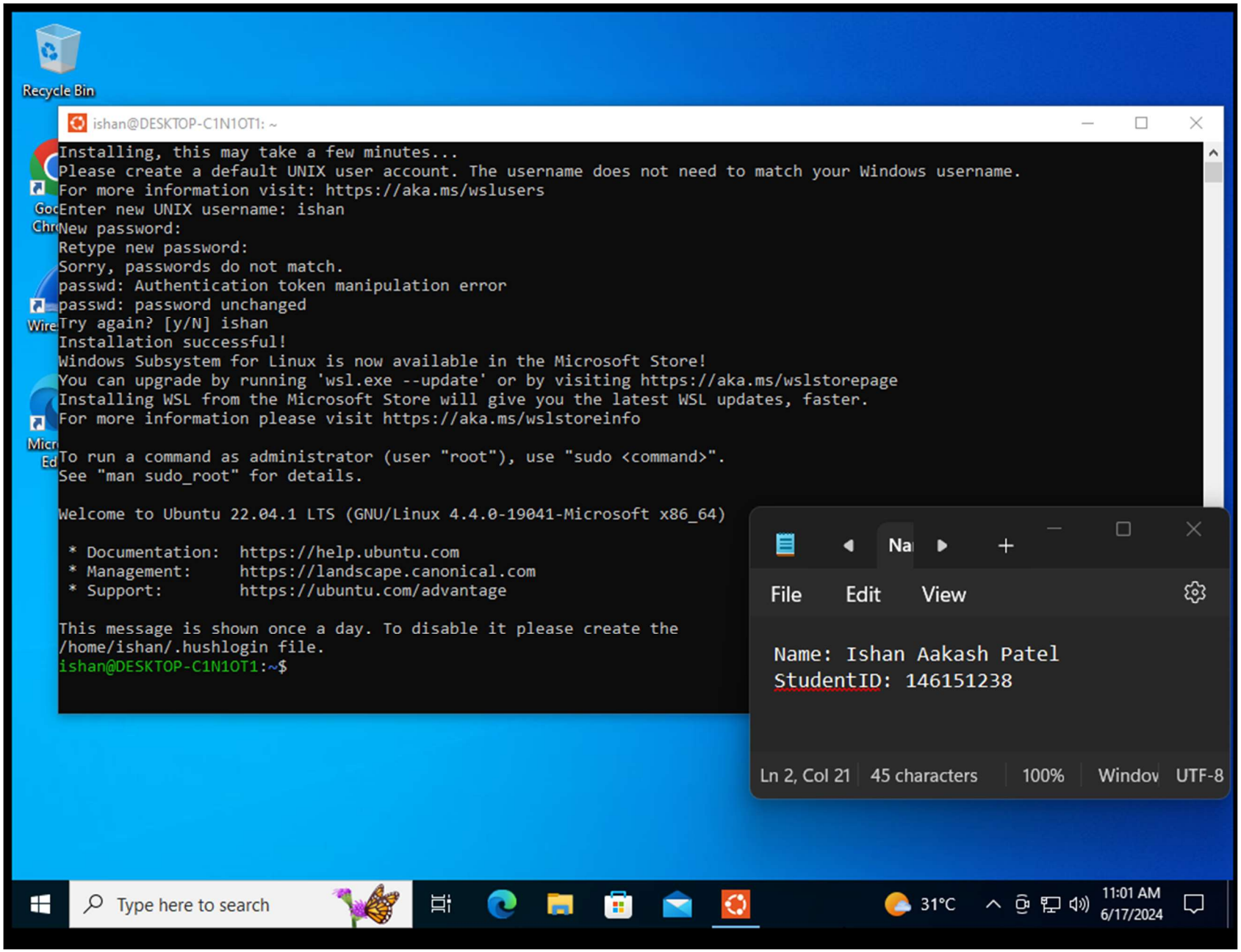
I had a Windows 10 already installed in my VMware Workstation Pro, So I had decided to build the forensics lab inside this vm.

First, installing WSL.

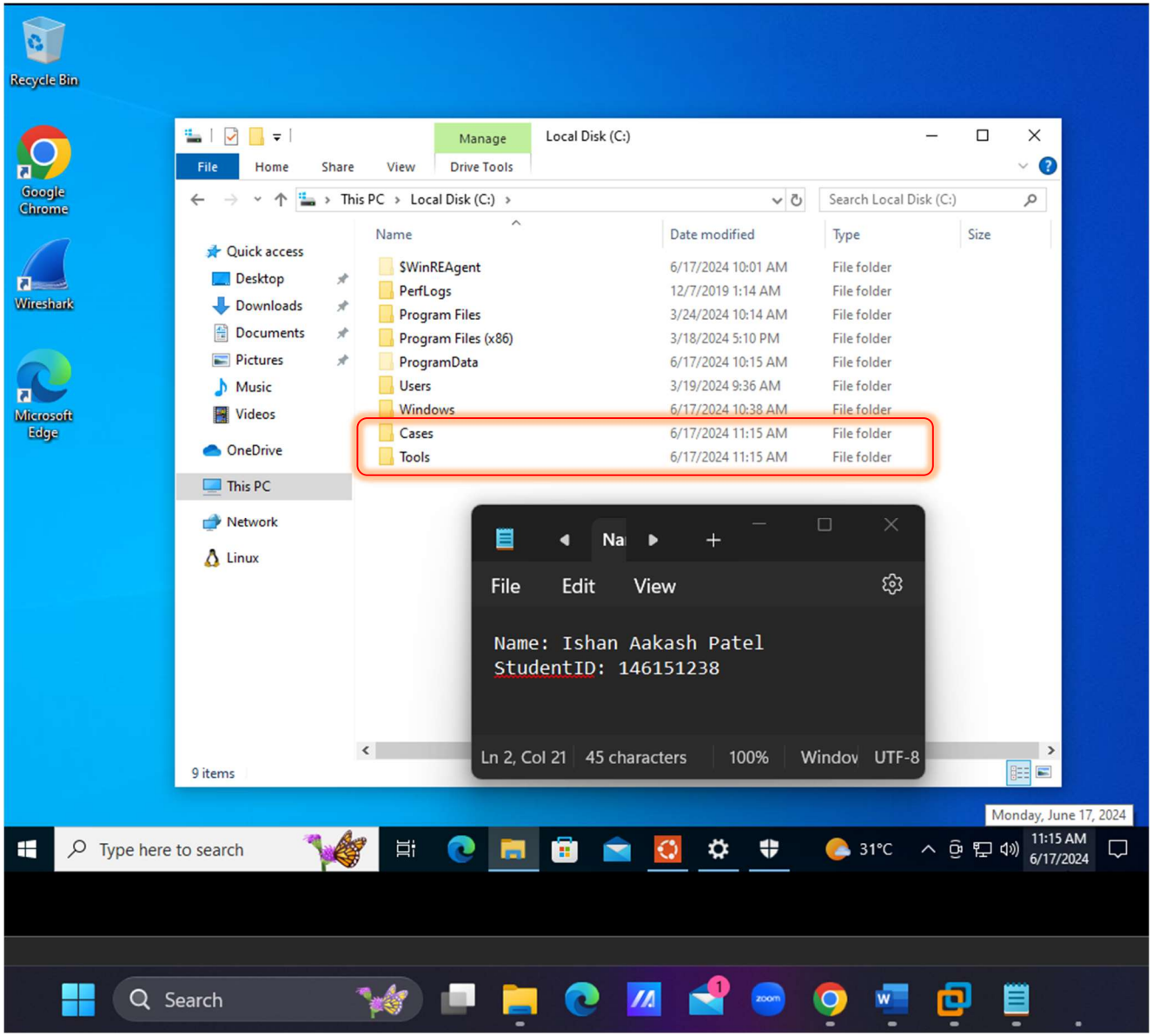
1) WSL



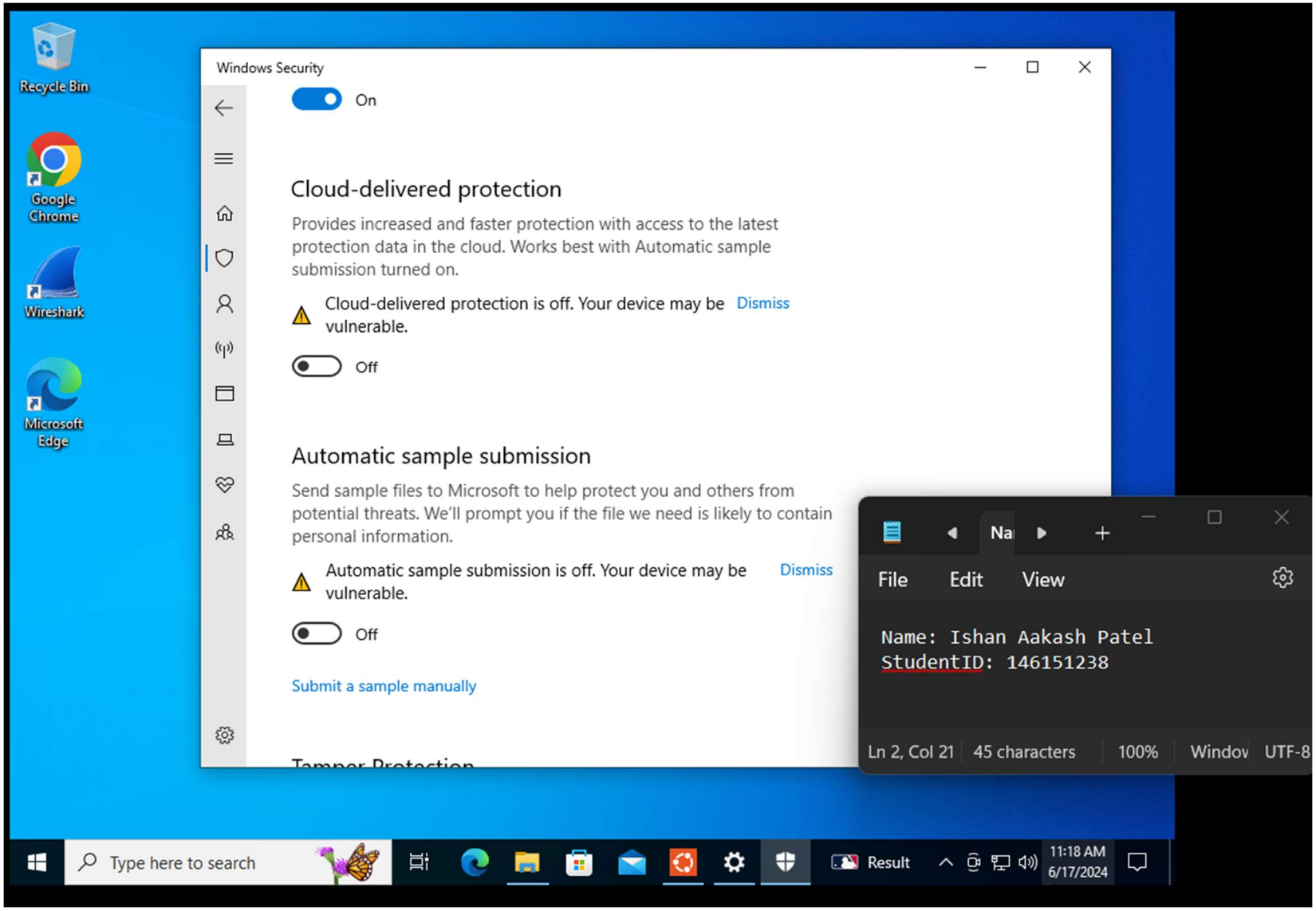

Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Deployment operation progress: C:\Users\ishan\Downloads\Ubuntu2204-221101\Ubuntu_2204.1.7.0_x64.appx
Processing
Add-AppxPackage : Cannot find path 'C:\Users\ishan\Downloads\Ubuntu2204-221101\Ubuntu_2004.4.2.0_x64.appx' because it does not exist.
At line:1 char:1
+ Add-AppxPackage .\Ubuntu_2004.4.2.0_x64.appx
+ CategoryInfo : ObjectNotFound: (C:\Users\ishan\...4.2.0_x64.appx:String) [Add-AppxPackage], ItemNotFoundException
+ FullyQualifiedErrorId : PathNotFound,Microsoft.Windows.Appx.PackageManager.Commands.AddAppxPackageCommand
PS C:\Users\ishan\Downloads\Ubuntu2204-221101> dir
Directory: C:\Users\ishan\Downloads\Ubuntu2204-221101
Mode LastWriteTime Length Name
d----- 6/17/2024 10:53 AM AppxMetadata
-a---- 6/17/2024 10:53 AM 338 AppxBlockMap.xml
-a---- 6/17/2024 10:53 AM 11964 AppxSignature.p7x
-a---- 6/17/2024 10:53 AM 541855027 Ubuntu_2204.1.7.0_ARM64.appx
-a---- 6/17/2024 10:53 AM 37054 Ubuntu_2204.1.7.0_scale-100.appx
-a---- 6/17/2024 10:53 AM 43413 Ubuntu_2204.1.7.0_scale-125.appx
-a---- 6/17/2024 10:53 AM 48873 Ubuntu_2204.1.7.0_scale-150.appx
-a---- 6/17/2024 10:53 AM 99037 Ubuntu_2204.1.7.0_scale-400.appx
-a---- 6/17/2024 10:53 AM 574740460 Ubuntu_2204.1.7.0_x64.appx
-a---- 6/17/2024 10:53 AM 469 [Content_Types].xml
PS C:\Users\ishan\Downloads\Ubuntu2204-221101> Add-AppxPackage .\Ubuntu_2204.1.7.0_x64.appx
File Edit View
Name: Ishan Aakash Patel
StudentID: 146151238
Ln 2, Col 21 45 characters 100% UTF-8
Type here to search
31°C
Meet Now
10:57 AM
6/17/2024
Search
ENG
IN
13:57
6/17/2024


Recycle Bin
ishan@DESKTOP-C1N1OT1: ~
Installing, this may take a few minutes...
Please create a default UNIX user account. The username does not need to match your Windows username.
For more information visit: https://aka.ms/wslusers
Enter new UNIX username: ishan
New password:
Retype new password:
Sorry, passwords do not match.
passwd: Authentication token manipulation error
passwd: password unchanged
Try again? [y/N] ishan
Installation successful!
Windows Subsystem for Linux is now available in the Microsoft Store!
You can upgrade by running 'wsl.exe --update' or by visiting https://aka.ms/wslstorepage
Installing WSL from the Microsoft Store will give you the latest WSL updates, faster.
For more information please visit https://aka.ms/wslstoreinfo
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 4.4.0-19041-Microsoft x86_64)
* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/advantage
This message is shown once a day. To disable it please create the
/home/ishan/.hushlogin file.
ishan@DESKTOP-C1N1OT1:~$
File
Edit
View
Name: Ishan Aakash Patel
StudentID: 146151238
Ln 2, Col 21
45 characters
100%
UTF-8
Type here to search
31°C
11:01 AM
6/17/2024

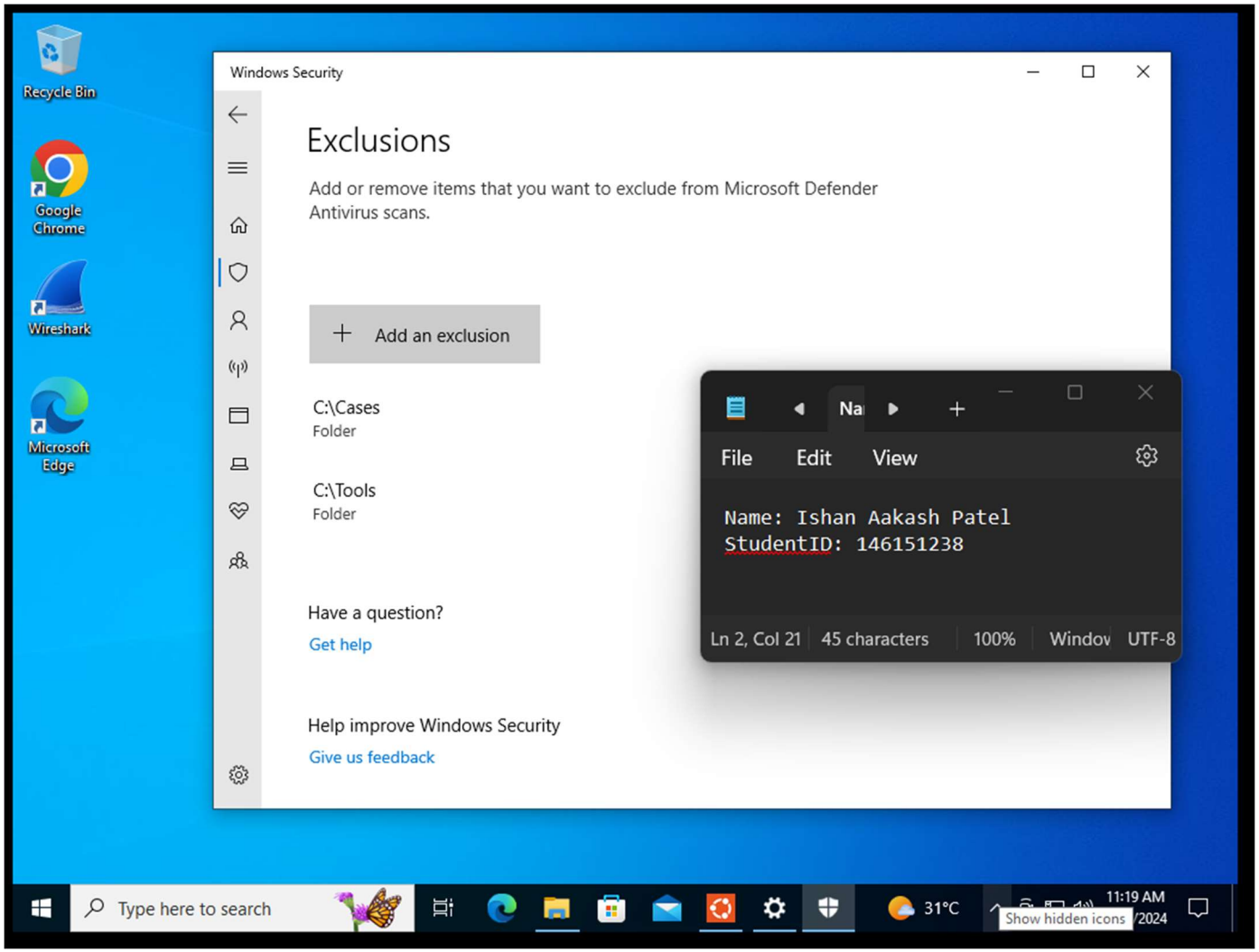
2) Create Cases & Tools in C Drive



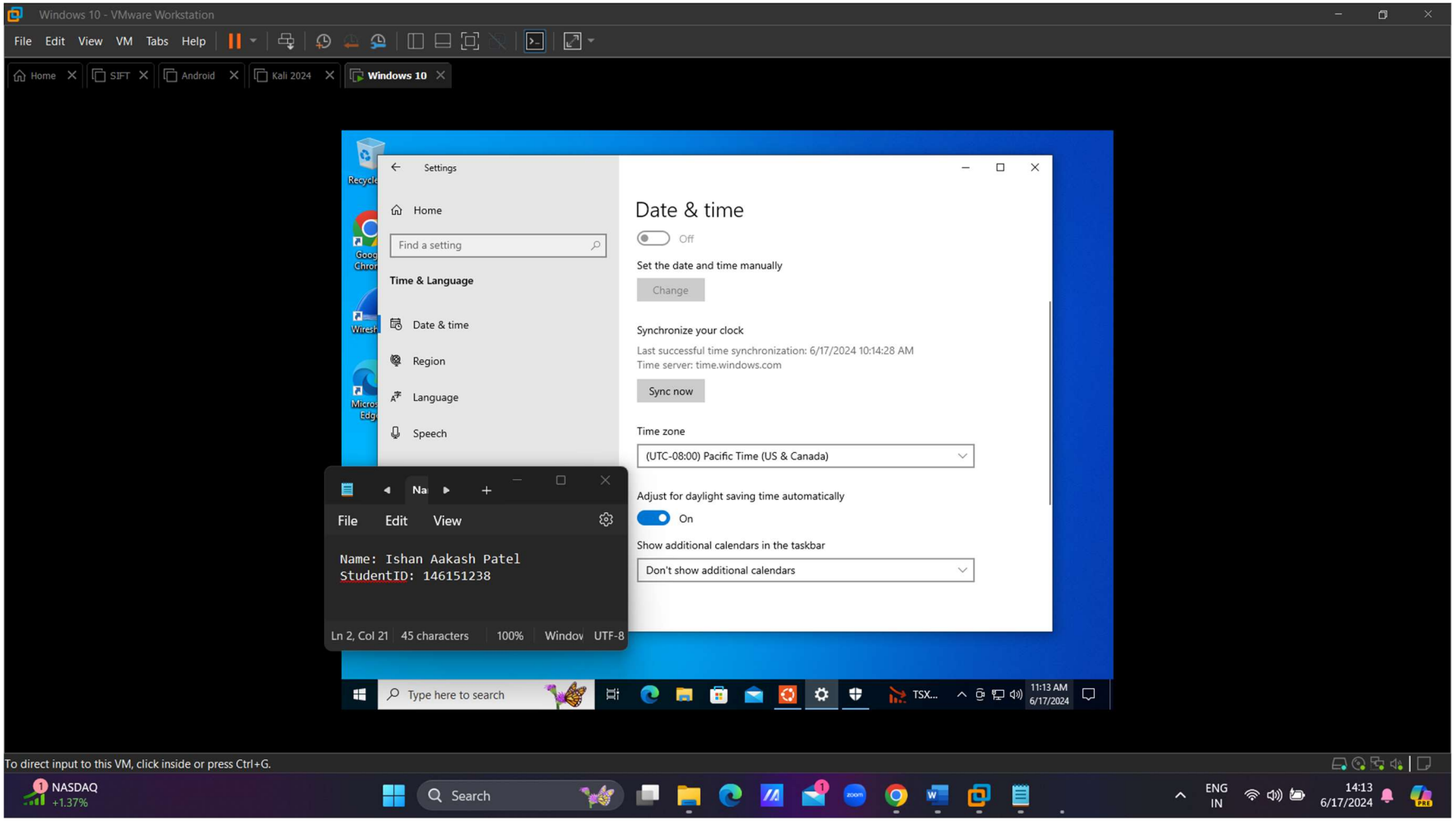
3) Configure Microsoft Defender



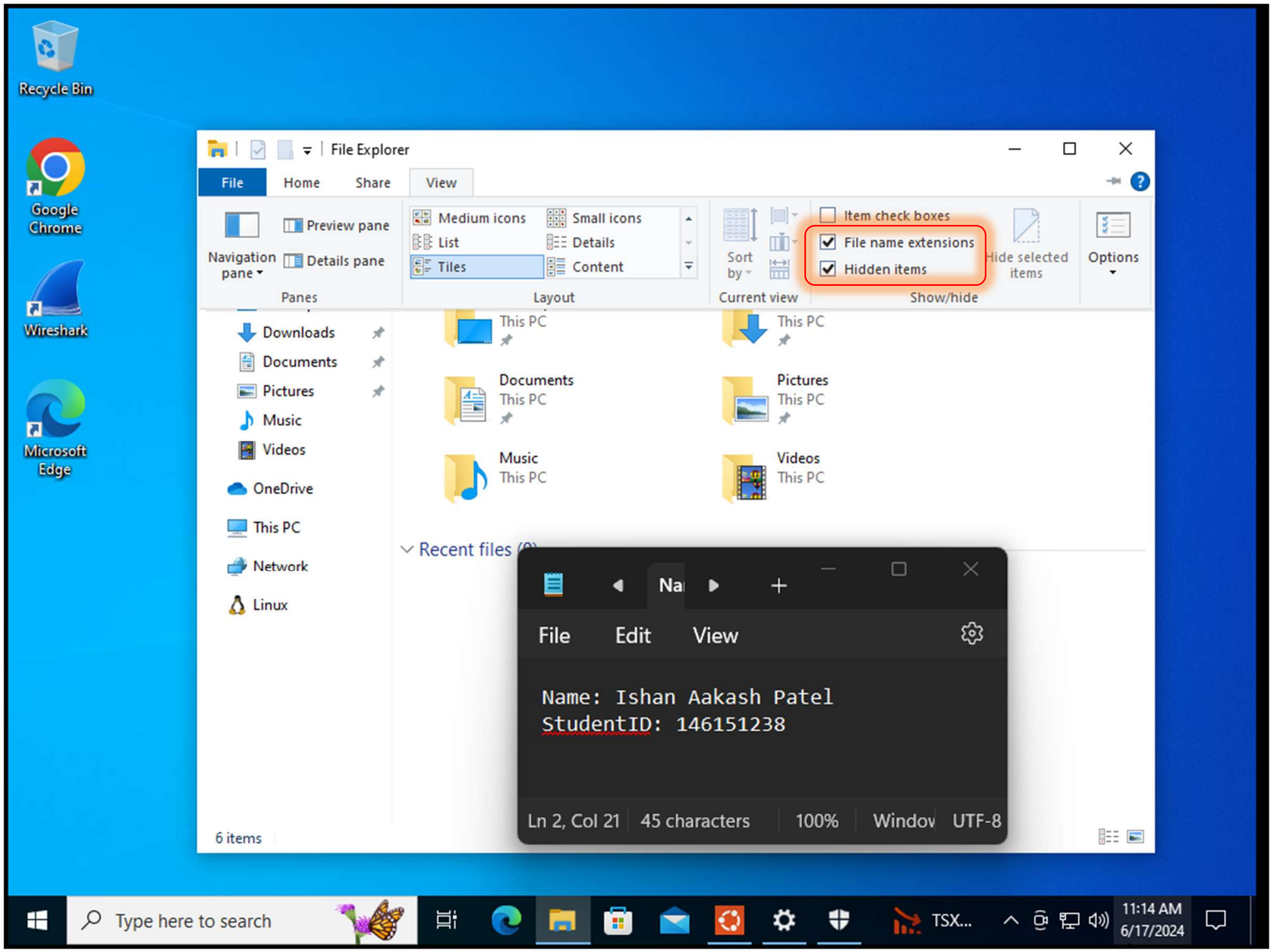
4) Exclude the Working Directories (Cases and Tools)



5) Set timezone to UTC



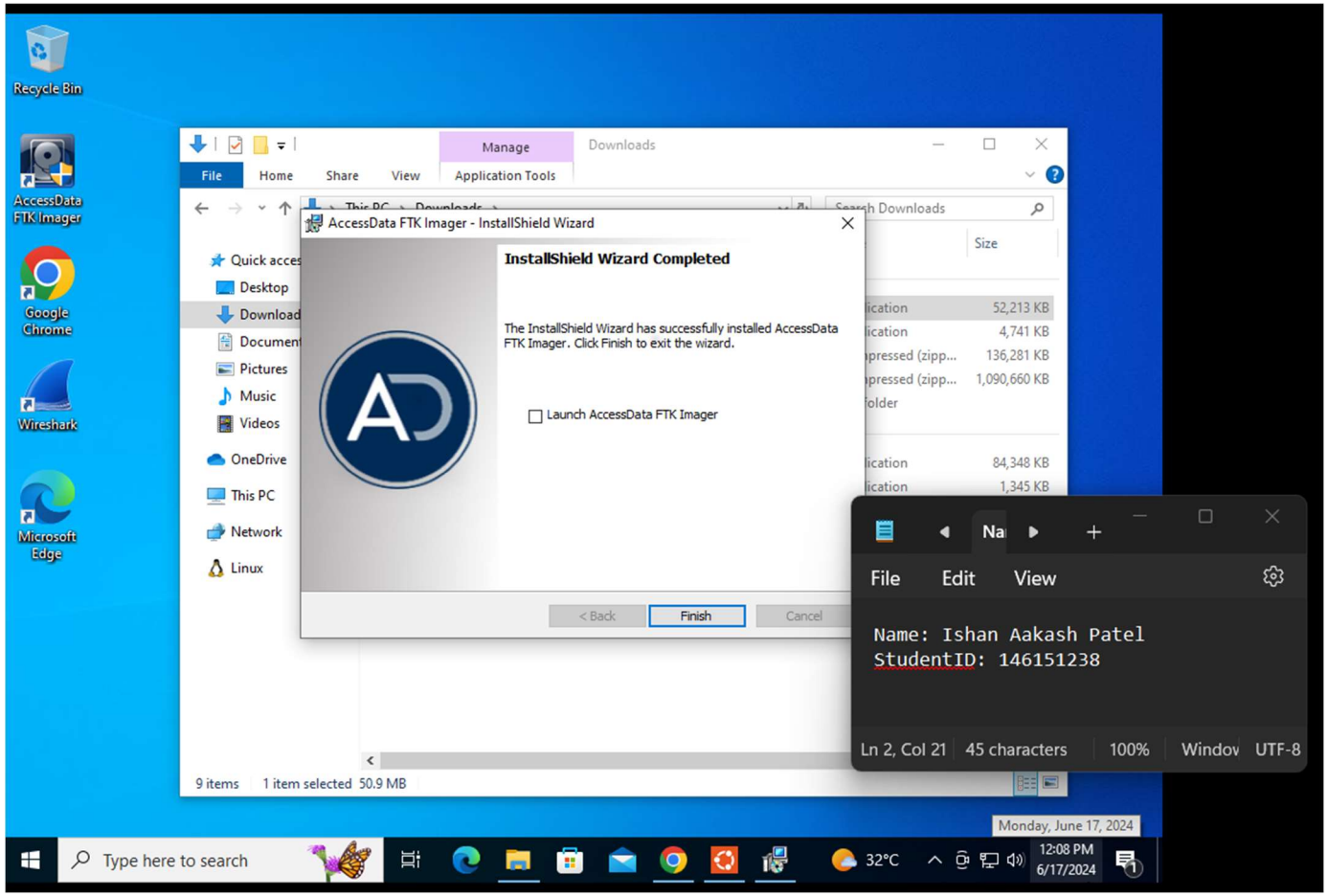
6) Show Hidden Files



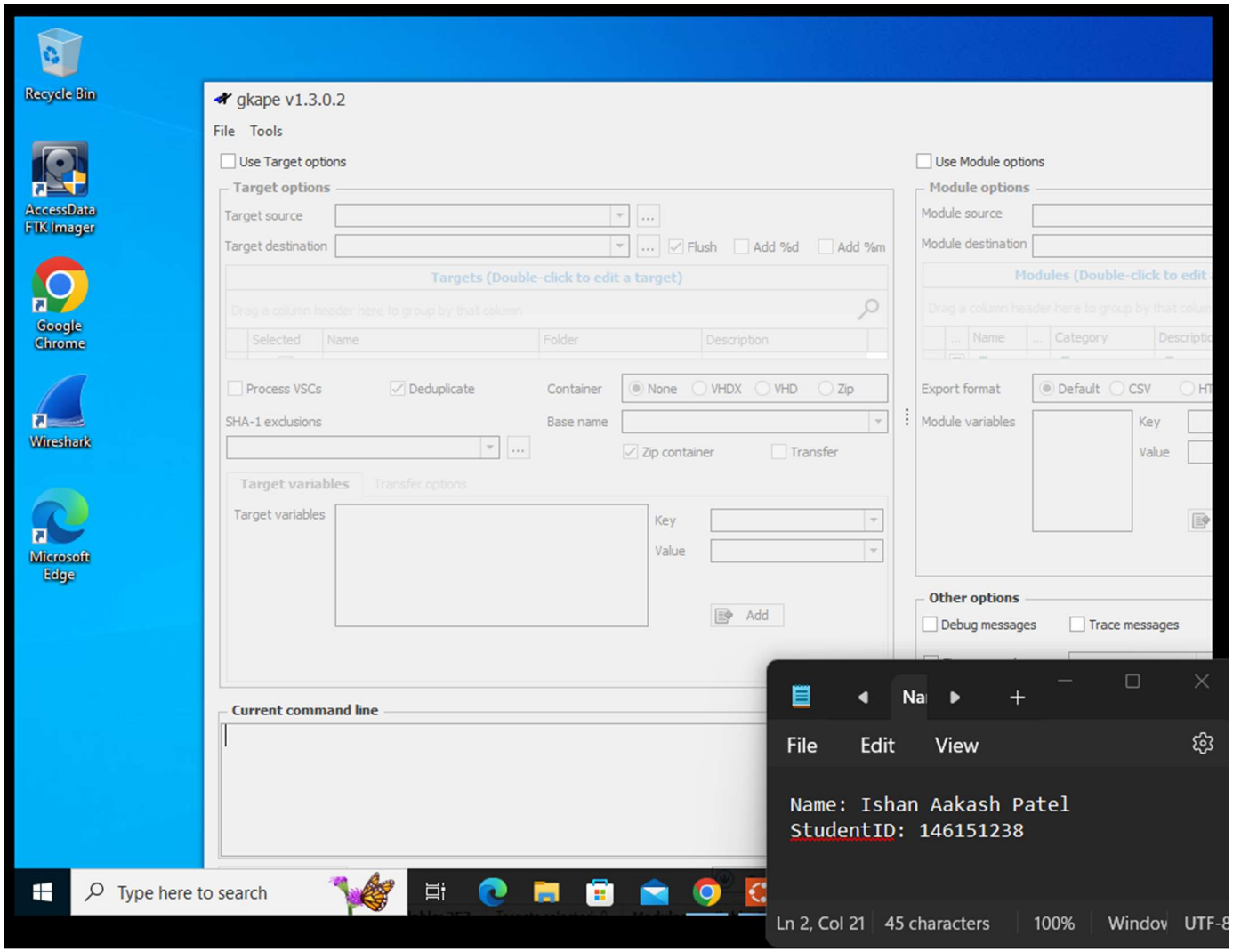
Now we will Install forensics tools.

Tools

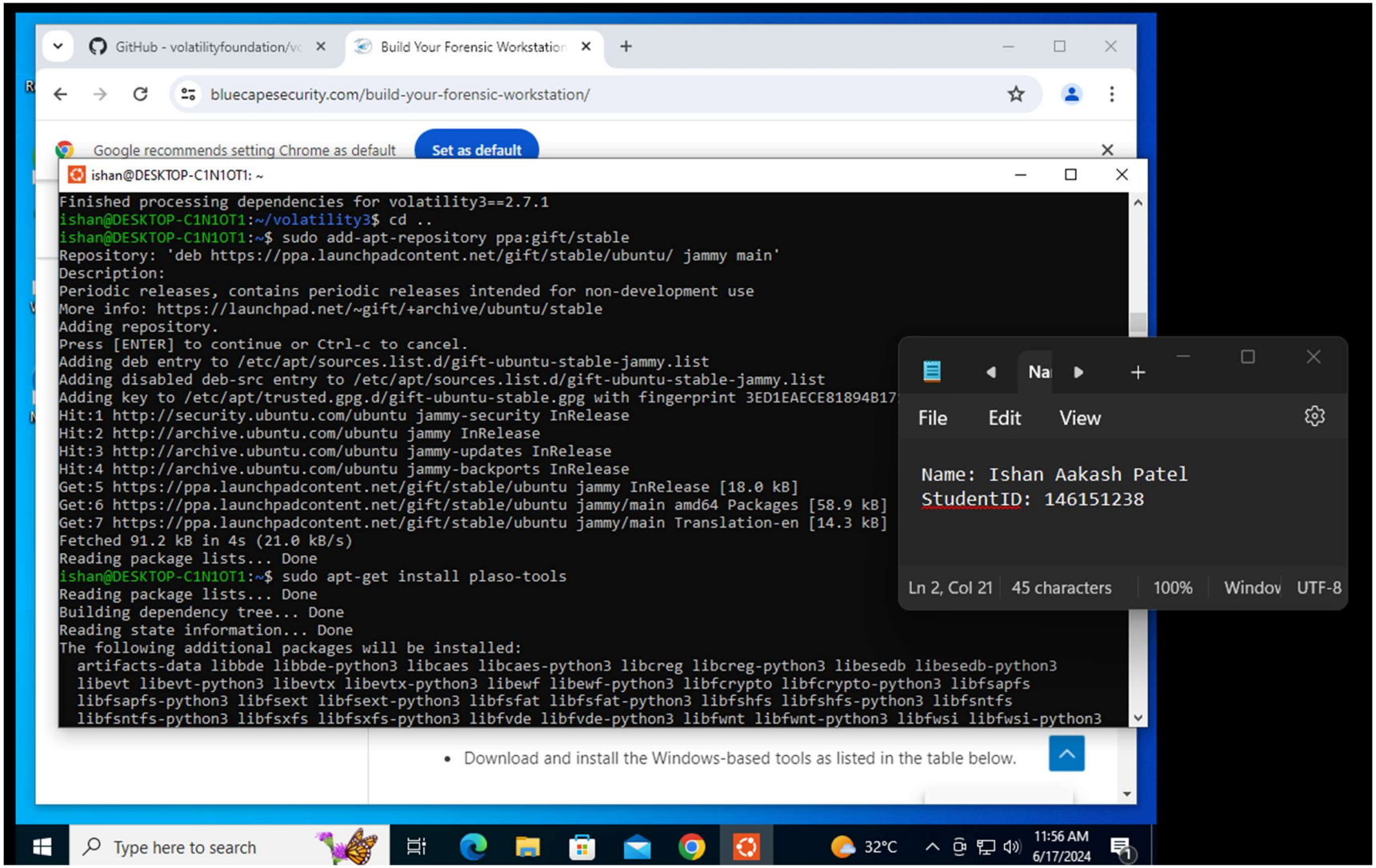
1) FTK Imager : FTK Imager, developed by AccessData, is a forensic imaging tool used to acquire and preview forensic images of computer systems and digital devices. It allows users to create exact bit-for-bit copies of hard drives, USB drives, and other storage media. Key features include the ability to view the contents of forensic images, mount images as drive letters, and export files and folders from images. FTK Imager also supports the generation of hash values for data integrity verification and can capture volatile memory. It is widely used in digital forensics for evidence preservation and analysis.

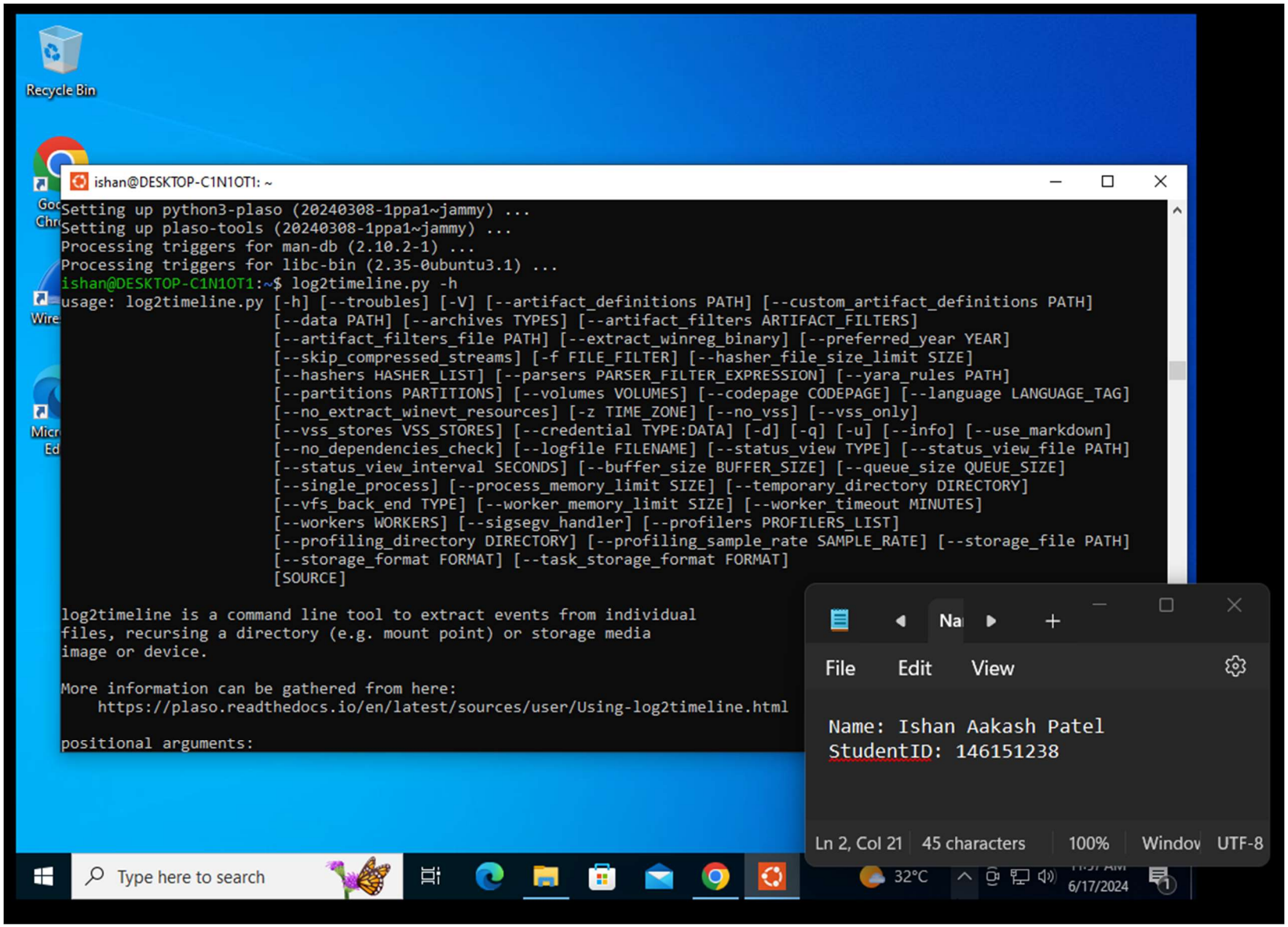


2) KAPE : KAPE (Kroll Artifact Parser and Extractor) is a digital forensics tool designed to quickly and efficiently collect and process forensic artifacts from Windows systems. Developed by Eric Zimmerman, KAPE streamlines the collection of key forensic artifacts, making it particularly useful for incident response and triage.

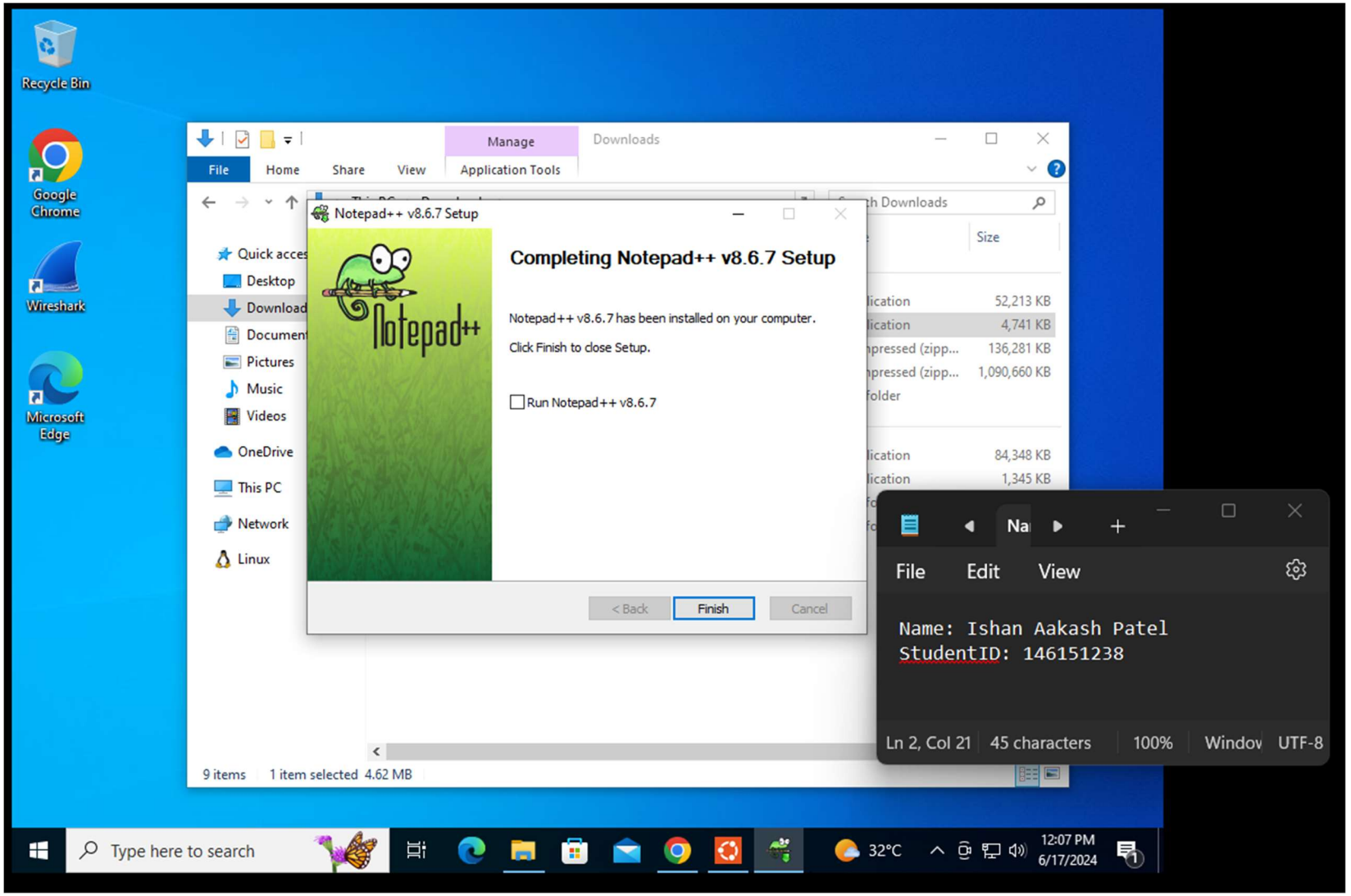


3) Log2timeline : Log2timeline, part of the Plaso project, is an open-source tool used in digital forensics for creating a timeline of events from various log files and forensic artifacts. It simplifies the process of aggregating different types of data sources to produce a single, unified timeline that can be analyzed to understand what happened on a system. First install Plaso tools

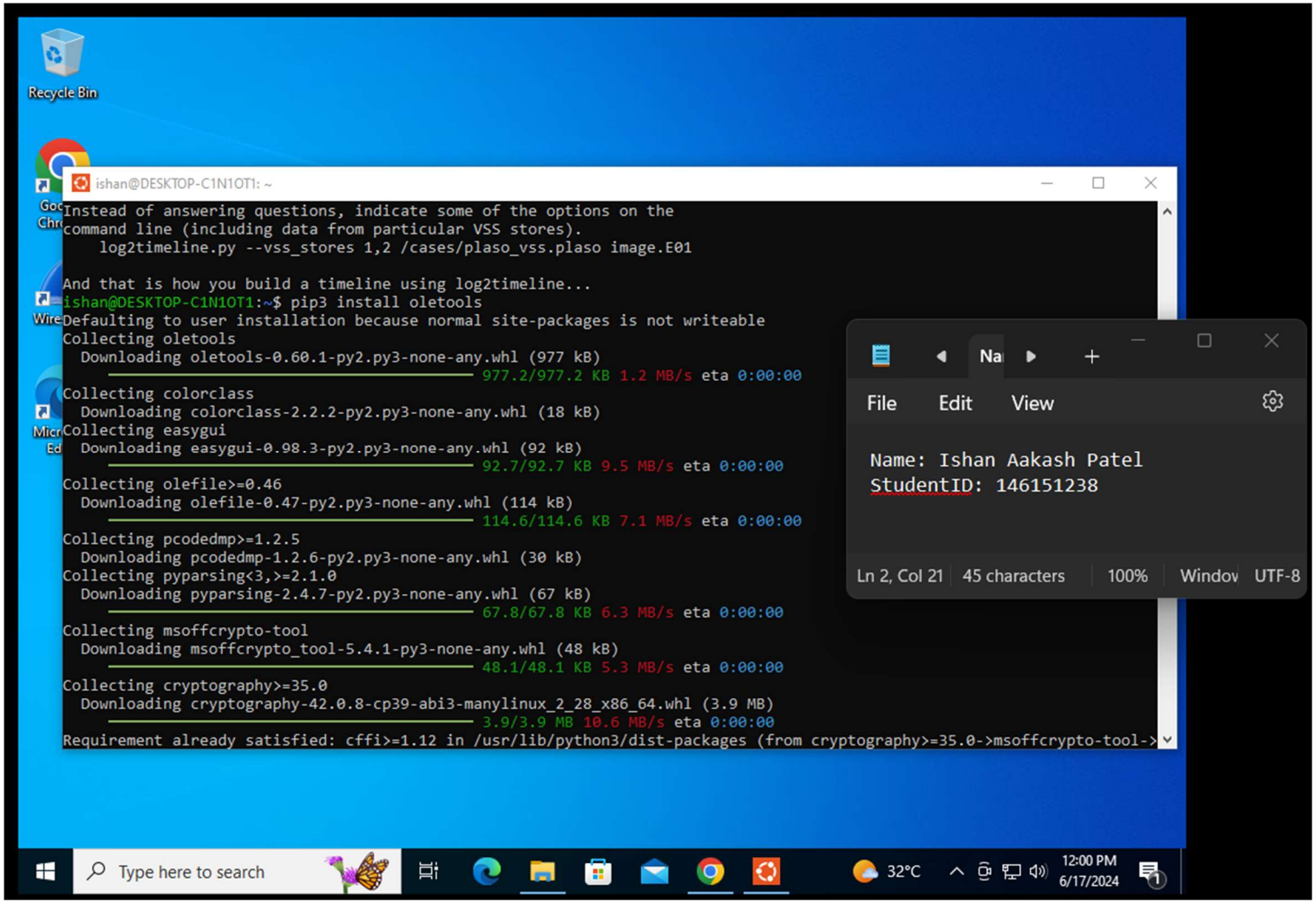


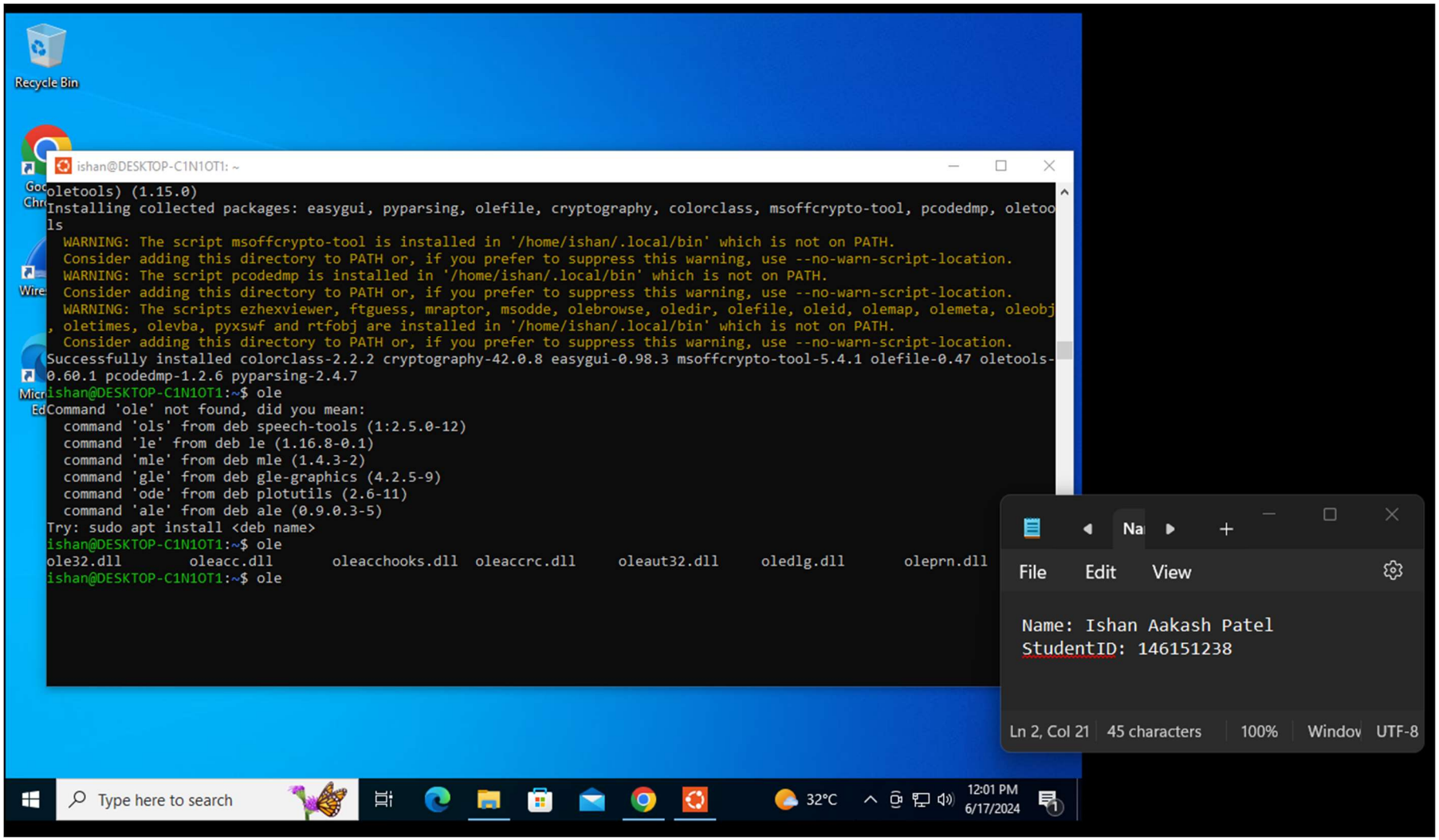

Recycle Bin
ishan@DESKTOP-C1N1OT1: ~
Setting up python3-plaso (20240308-1ppa1~jammy) ...
Setting up plaso-tools (20240308-1ppa1~jammy) ...
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for libc-bin (2.35-0ubuntu3.1) ...
ishan@DESKTOP-C1N1OT1:~$ log2timeline.py -h
usage: log2timeline.py [-h] [--troubles] [-V] [--artifact_definitions PATH] [--custom_artifact_definitions PATH]
[--data PATH] [--archives TYPES] [--artifact_filters ARTIFACT_FILTERS]
[--artifact_filters_file PATH] [--extract_winreg_binary] [--preferred_year YEAR]
[--skip_compressed_streams] [-f FILE_FILTER] [--hasher_file_size_limit SIZE]
[--hashers HASHER_LIST] [--parsers PARSER_FILTER_EXPRESSION] [--yara_rules PATH]
[--partitions PARTITIONS] [--volumes VOLUMES] [--codepage CODEPAGE] [--language LANGUAGE_TAG]
[--no_extract_winevt_resources] [-z TIME_ZONE] [--no_vss] [--vss_only]
[--vss_stores VSS_STORES] [--credential TYPE:DATA] [-d] [-q] [-u] [--info] [--use_markdown]
[--no_dependencies_check] [--logfile FILENAME] [--status_view TYPE] [--status_view_file PATH]
[--status_view_interval SECONDS] [--buffer_size BUFFER_SIZE] [--queue_size QUEUE_SIZE]
[--single_process] [--process_memory_limit SIZE] [--temporary_directory DIRECTORY]
[--vfs_back_end TYPE] [--worker_memory_limit SIZE] [--worker_timeout MINUTES]
[--workers WORKERS] [--sigsegv_handler] [--profilers PROFILERS_LIST]
[--profiling_directory DIRECTORY] [--profiling_sample_rate SAMPLE_RATE] [--storage_file PATH]
[--storage_format FORMAT] [--task_storage_format FORMAT]
[SOURCE]
log2timeline is a command line tool to extract events from individual
files, recursing a directory (e.g. mount point) or storage media
image or device.
More information can be gathered from here:
https://plaso.readthedocs.io/en/latest/sources/user/Using-log2timeline.html
positional arguments:
File
Edit
View
Name: Ishan Aakash Patel
StudentID: 146151238
Ln 2, Col 21
45 characters
100%
UTF-8
Type here to search
32°C
6/17/2024

4) Notepad++ : Notepad++ is a free, open-source text and source code editor for Windows. It supports multiple programming languages and features syntax highlighting, code folding, and a tabbed interface. Notepad++ is known for its speed, lightweight design, and extensive plugin ecosystem, making it a popular choice for programmers and text editors alike.

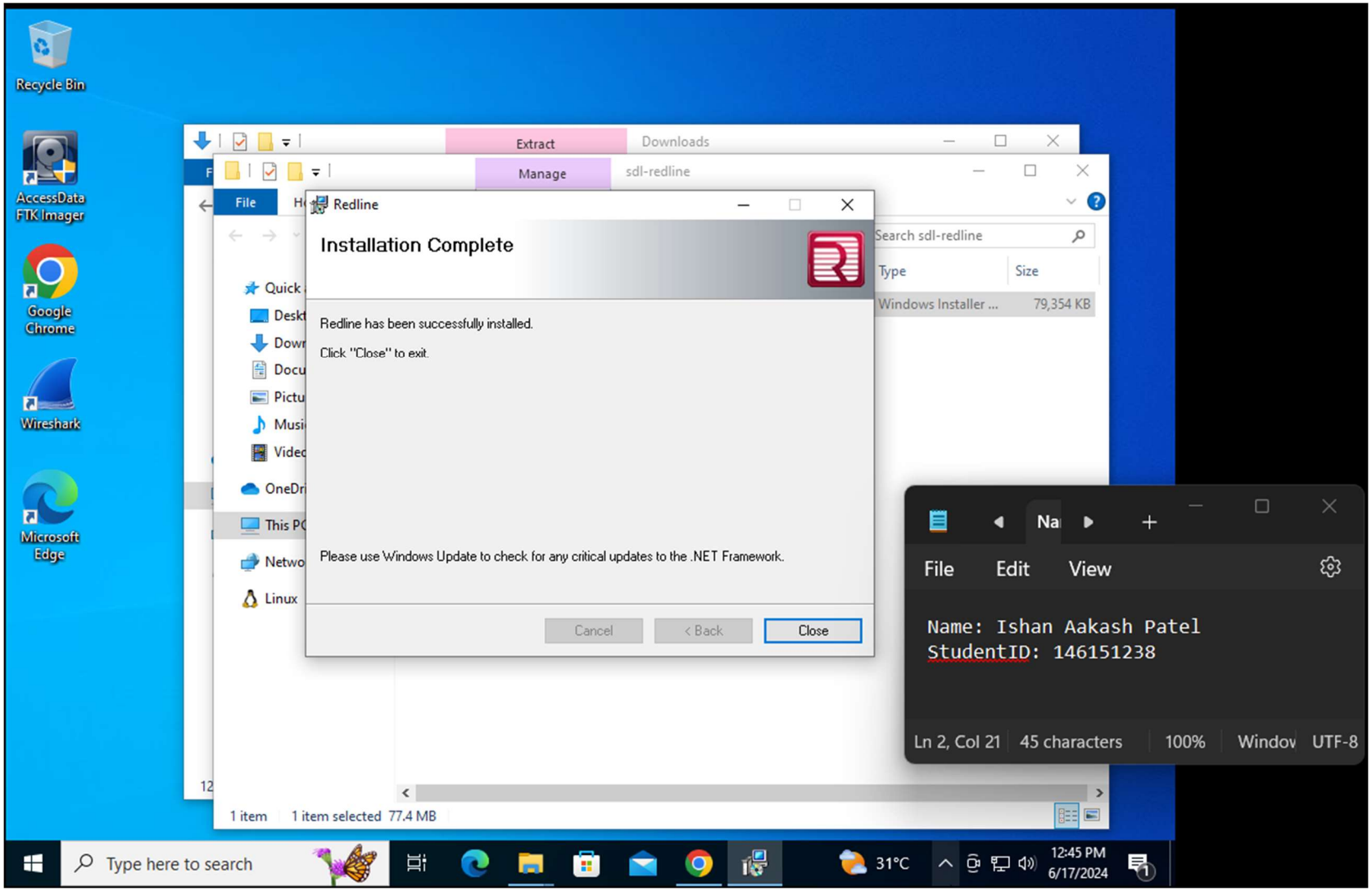


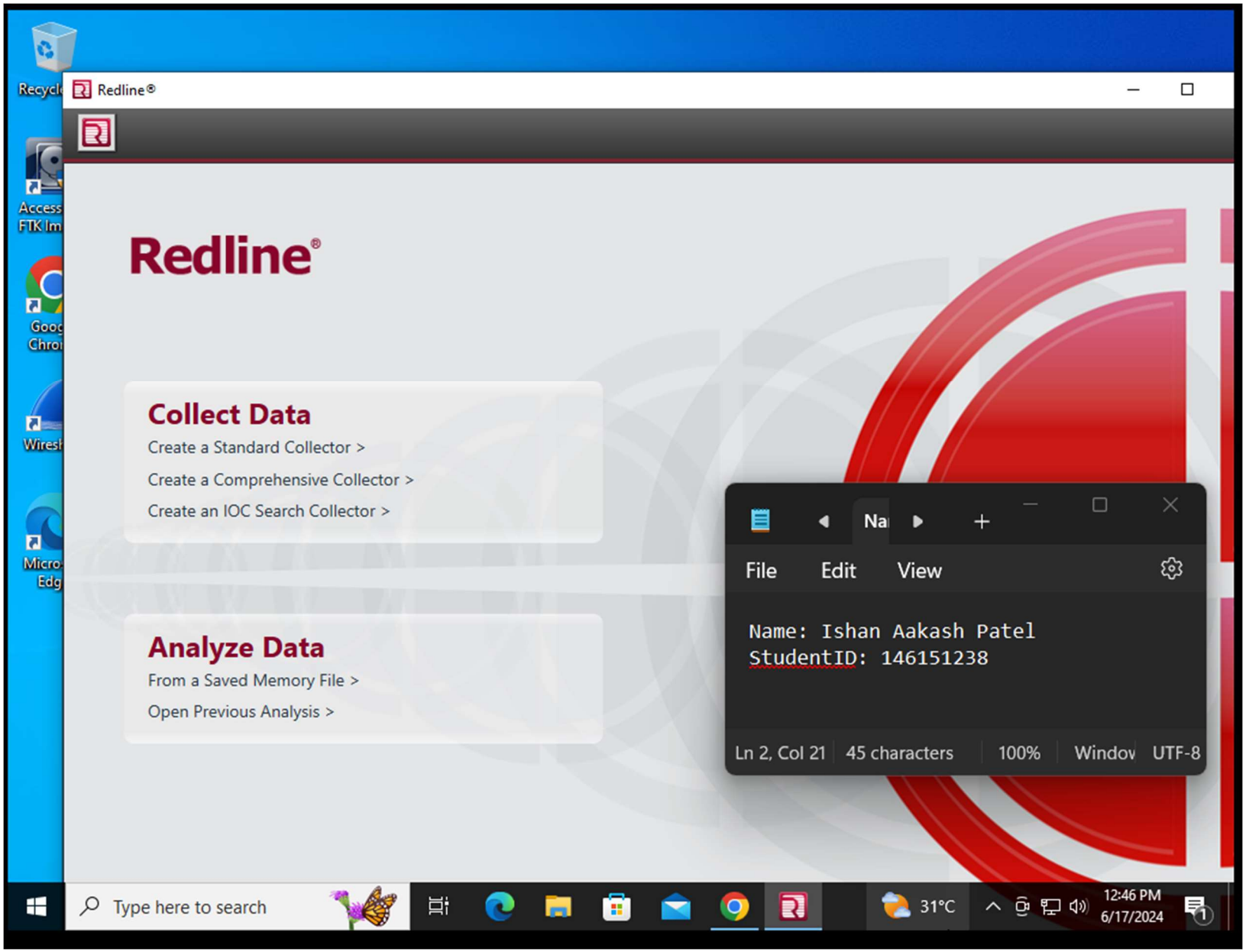
5) Ole tools : OLE forensics involves analyzing Object Linking and Embedding (OLE) objects within digital documents to uncover hidden data, metadata, and potential malicious content. This type of analysis is crucial in digital forensics, especially when examining documents such as Microsoft Office files, which often use OLE for embedding various objects.



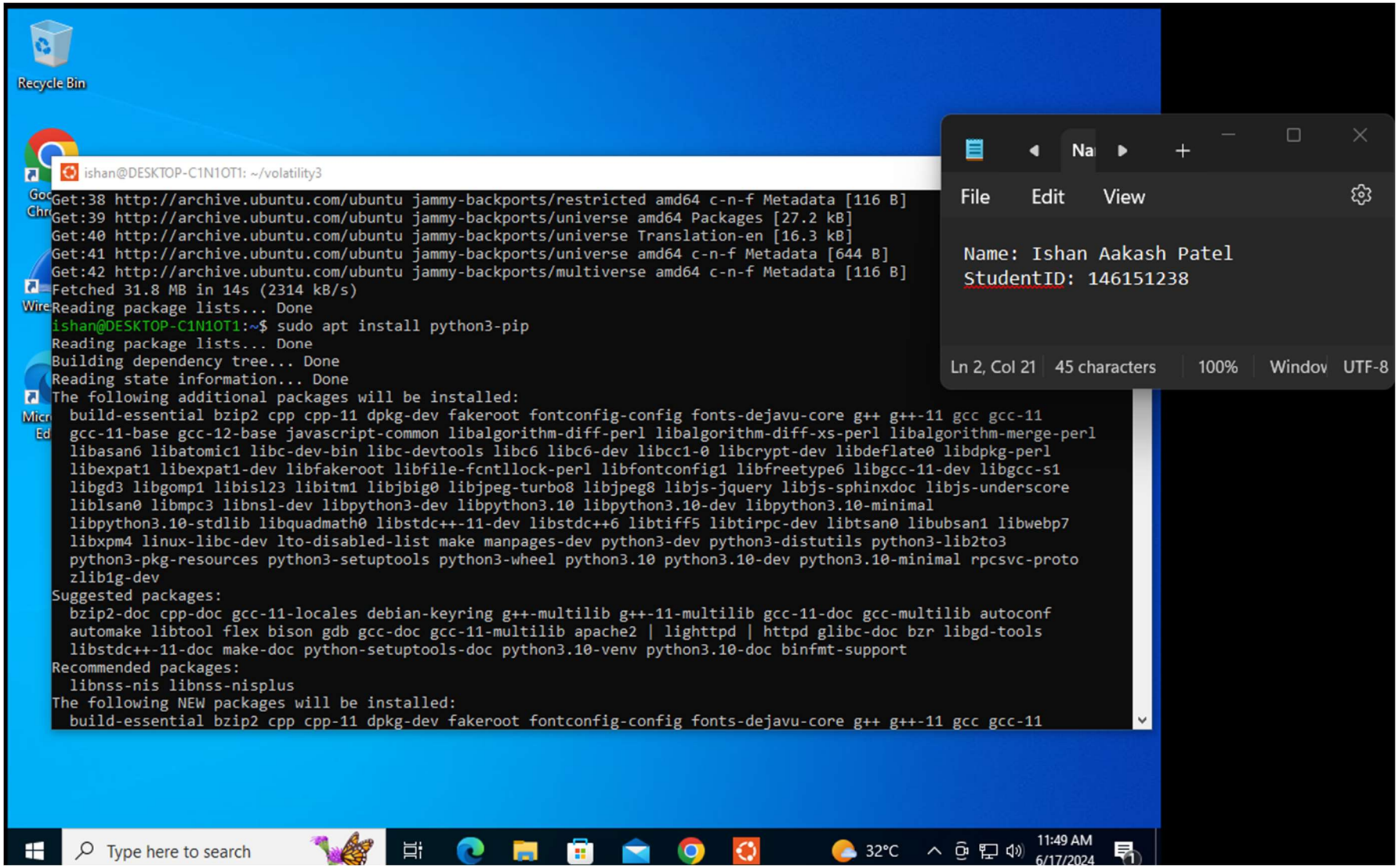

Recycle Bin
ishan@DESKTOP-C1N1OT1: ~
oletools) (1.15.0)
Installing collected packages: easygui, pyparsing, olefile, cryptography, colorclass, msoffcrypto-tool, pcodedmp, oletools
WARNING: The script msoffcrypto-tool is installed in '/home/ishan/.local/bin' which is not on PATH.
Consider adding this directory to PATH or, if you prefer to suppress this warning, use --no-warn-script-location.
WARNING: The script pcodedmp is installed in '/home/ishan/.local/bin' which is not on PATH.
Consider adding this directory to PATH or, if you prefer to suppress this warning, use --no-warn-script-location.
WARNING: The scripts ezhexviewer, ftguess, mraptor, msodde, olebrowse, oledir, olefile, oleid, olemap, olemeta, oleobj, oletimes, olevba, pyxswf and rtfobj are installed in '/home/ishan/.local/bin' which is not on PATH.
Consider adding this directory to PATH or, if you prefer to suppress this warning, use --no-warn-script-location.
Successfully installed colorclass-2.2.2 cryptography-42.0.8 easygui-0.98.3 msoffcrypto-tool-5.4.1 olefile-0.47 oletools-0.60.1 pcodedmp-1.2.6 pyparsing-2.4.7
ishan@DESKTOP-C1N1OT1:~$ ole
Command 'ole' not found, did you mean:
command 'ols' from deb speech-tools (1:2.5.0-12)
command 'le' from deb le (1.16.8-0.1)
command 'mle' from deb mle (1.4.3-2)
command 'gle' from deb gle-graphics (4.2.5-9)
command 'ode' from deb plotutils (2.6-11)
command 'ale' from deb ale (0.9.0.3-5)
Try: sudo apt install <deb name>
ishan@DESKTOP-C1N1OT1:~$ ole
ole32.dll oleacc.dll oleacchooks.dll oleaccrc.dll oleaut32.dll oledlg.dll oleprn.dll
ishan@DESKTOP-C1N1OT1:~$ ole
File Edit View
Name: Ishan Aakash Patel
StudentID: 146151238
Ln 2, Col 21 45 characters 100% Windov UTF-8
Type here to search
32°C
12:01 PM
6/17/2024

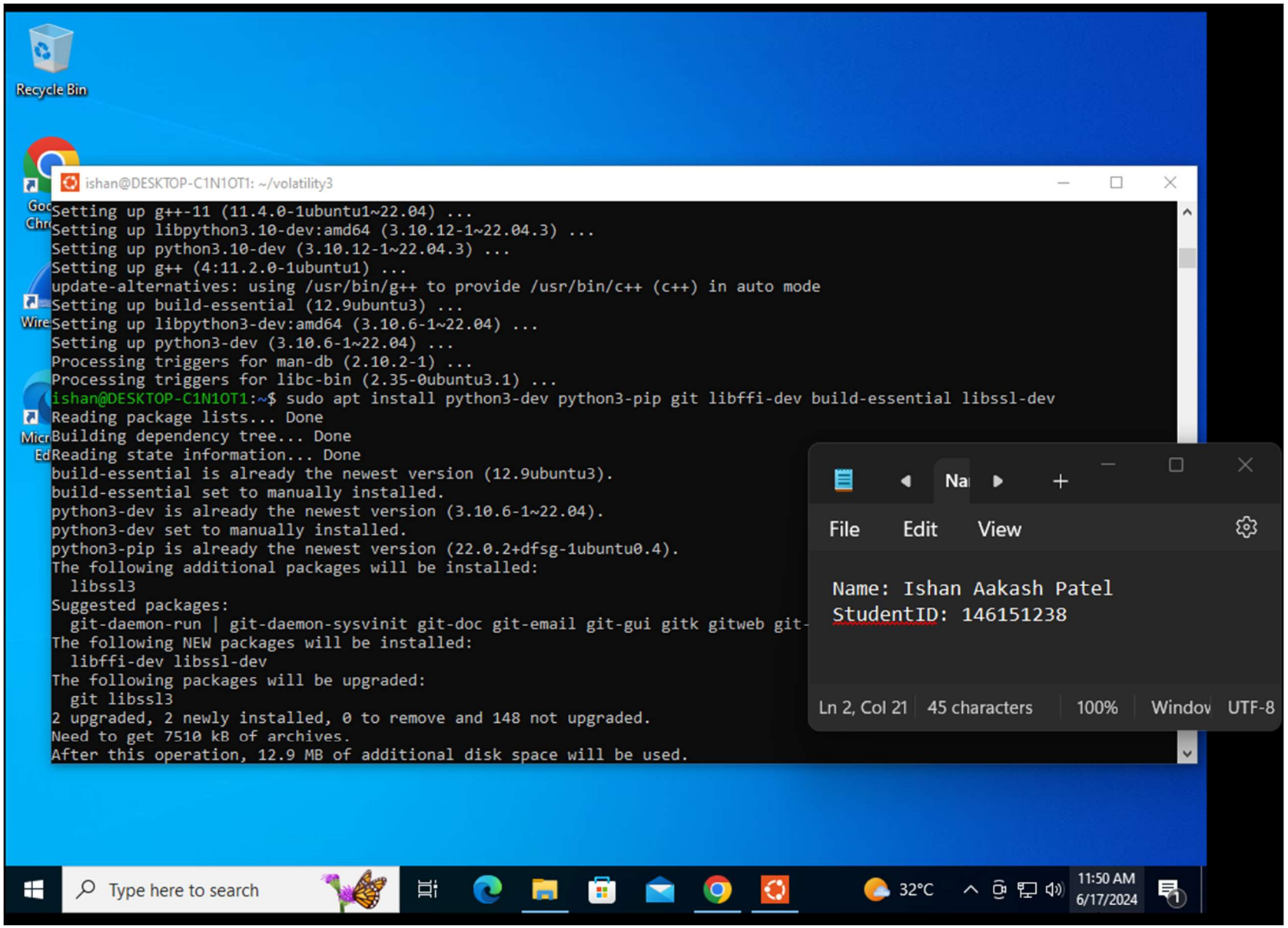
6) Redline : Redline, developed by FireEye, is a free endpoint security tool designed for memory and file analysis on Windows systems. It assists in identifying and analyzing potential security incidents by providing comprehensive information about the system's state.

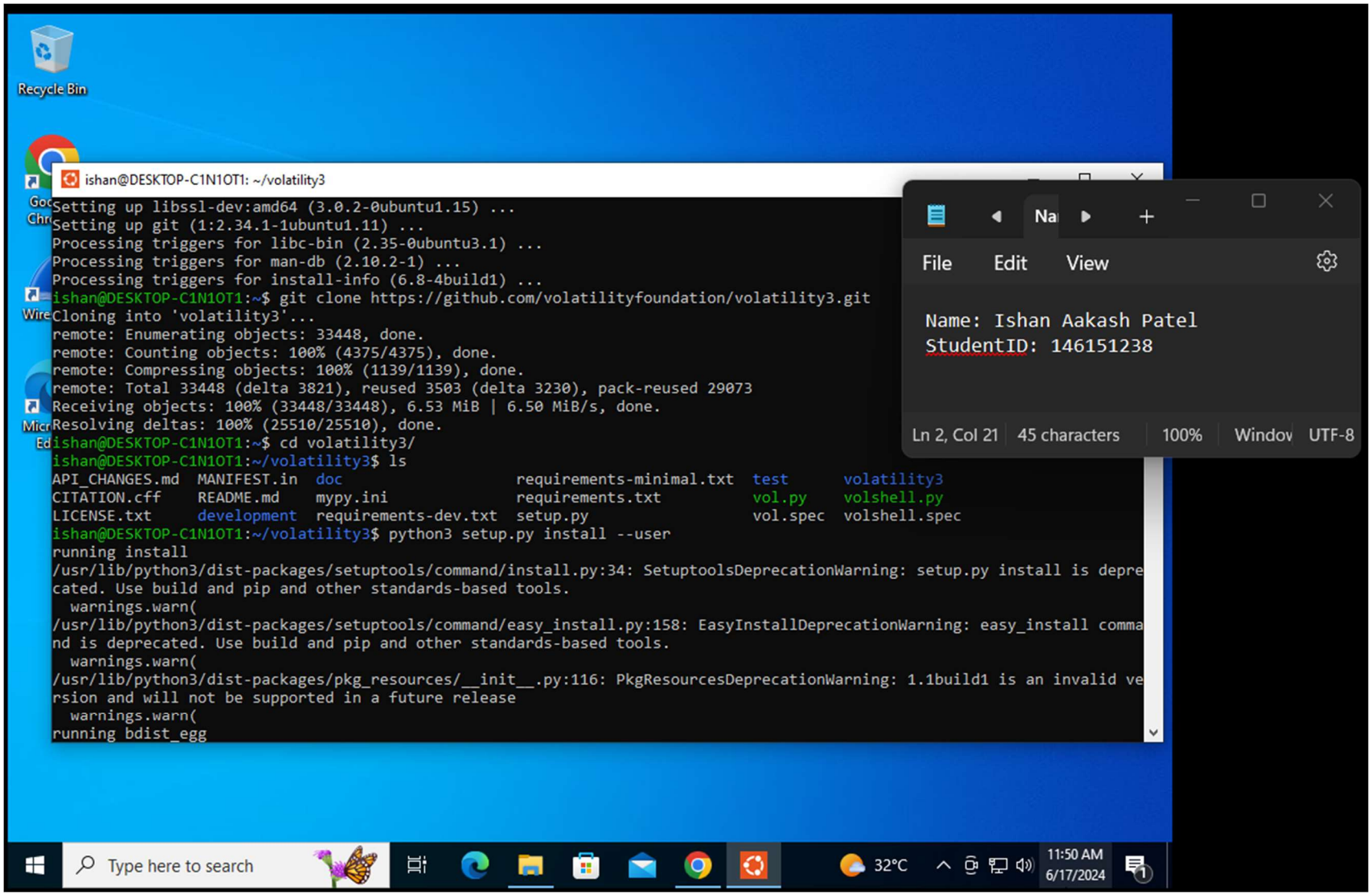


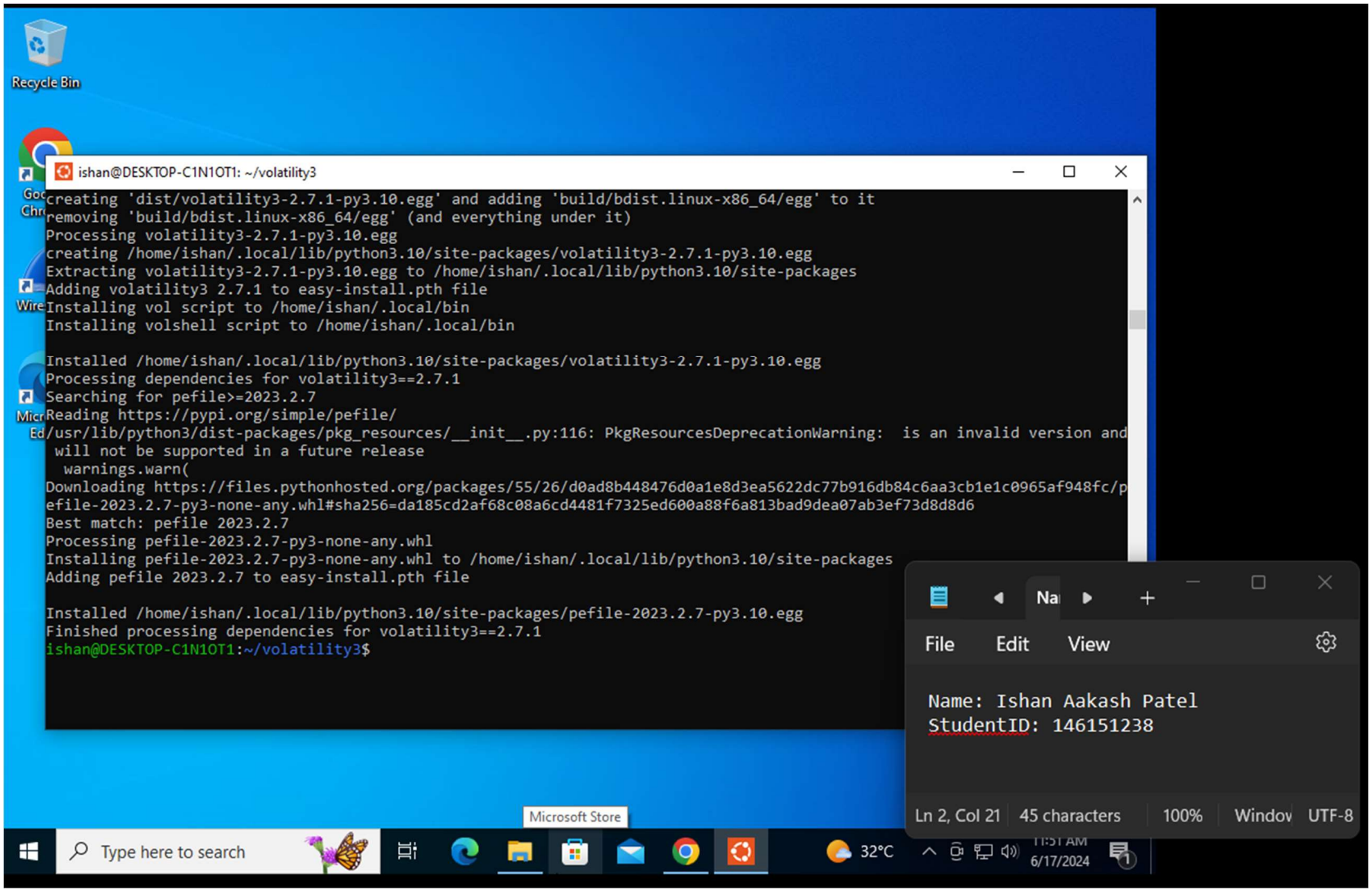

Redline®
Redline®
Collect Data
Create a Standard Collector >
Create a Comprehensive Collector >
Create an IOC Search Collector >
Analyze Data
From a Saved Memory File >
Open Previous Analysis >
File
Edit
View
Name: Ishan Aakash Patel
StudentID: 146151238
Ln 2, Col 21
45 characters
100%
UTF-8
Type here to search
31°C
12:46 PM
6/17/2024

7) Volatility




Recycle Bin
ishan@DESKTOP-C1N1OT1: ~/volatility3
Setting up g++-11 (11.4.0-1ubuntu1~22.04) ...
Setting up libpython3.10-dev:amd64 (3.10.12-1~22.04.3) ...
Setting up python3.10-dev (3.10.12-1~22.04.3) ...
Setting up g++ (4:11.2.0-1ubuntu1) ...
update-alternatives: using /usr/bin/g++ to provide /usr/bin/c++ (c++) in auto mode
Setting up build-essential (12.9ubuntu3) ...
Setting up libpython3-dev:amd64 (3.10.6-1~22.04) ...
Setting up python3-dev (3.10.6-1~22.04) ...
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for libc-bin (2.35-0ubuntu3.1) ...
ishan@DESKTOP-C1N1OT1:~$ sudo apt install python3-dev python3-pip git libffi-dev build-essential libssl-dev
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
build-essential is already the newest version (12.9ubuntu3).
build-essential set to manually installed.
python3-dev is already the newest version (3.10.6-1~22.04).
python3-dev set to manually installed.
python3-pip is already the newest version (22.0.2+dfsg-1ubuntu0.4).
The following additional packages will be installed:
libssl3
Suggested packages:
git-daemon-run | git-daemon-sysvinit git-doc git-email git-gui gitk gitweb git-
The following NEW packages will be installed:
libffi-dev libssl-dev
The following packages will be upgraded:
git libssl3
2 upgraded, 2 newly installed, 0 to remove and 148 not upgraded.
Need to get 7510 kB of archives.
After this operation, 12.9 MB of additional disk space will be used.
File
Edit
View
Name: Ishan Aakash Patel
StudentID: 146151238
Ln 2, Col 21
45 characters
100%
UTF-8
Type here to search
32°C
11:50 AM
6/17/2024


Recycle Bin
ishan@DESKTOP-C1N1OT1: ~/volatility3
Setting up libssl-dev:amd64 (3.0.2-0ubuntu1.15) ...
Setting up git (1:2.34.1-1ubuntu1.11) ...
Processing triggers for libc-bin (2.35-0ubuntu3.1) ...
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for install-info (6.8-4build1) ...
ishan@DESKTOP-C1N1OT1:~$ git clone https://github.com/volatilityfoundation/volatility3.git
Cloning into 'volatility3'...
remote: Enumerating objects: 33448, done.
remote: Counting objects: 100% (4375/4375), done.
remote: Compressing objects: 100% (1139/1139), done.
remote: Total 33448 (delta 3821), reused 3503 (delta 3230), pack-reused 29073
Receiving objects: 100% (33448/33448), 6.53 MiB | 6.50 MiB/s, done.
Resolving deltas: 100% (25510/25510), done.
ishan@DESKTOP-C1N1OT1:~$ cd volatility3/
ishan@DESKTOP-C1N1OT1:~/volatility3$ ls
API_CHANGES.md MANIFEST.in doc requirements-minimal.txt test volatility3
CITATION.cff README.md mypy.ini requirements.txt vol.py volshell.py
LICENSE.txt development requirements-dev.txt setup.py vol.spec volshell.spec
ishan@DESKTOP-C1N1OT1:~/volatility3$ python3 setup.py install --user
running install
/usr/lib/python3/dist-packages/setuptools/command/install.py:34: SetuptoolsDeprecationWarning: setup.py install is deprecated. Use build and pip and other standards-based tools.
warnings.warn(
/usr/lib/python3/dist-packages/setuptools/command/easy_install.py:158: EasyInstallDeprecationWarning: easy_install command is deprecated. Use build and pip and other standards-based tools.
warnings.warn(
/usr/lib/python3/dist-packages/pkg_resources/__init__.py:116: PkgResourcesDeprecationWarning: 1.1build1 is an invalid version and will not be supported in a future release
warnings.warn(
running bdist_egg
File Edit View
Name: Ishan Aakash Patel
StudentID: 146151238
Ln 2, Col 21 45 characters 100% Windov UTF-8
Type here to search
32°C
11:50 AM
6/17/2024



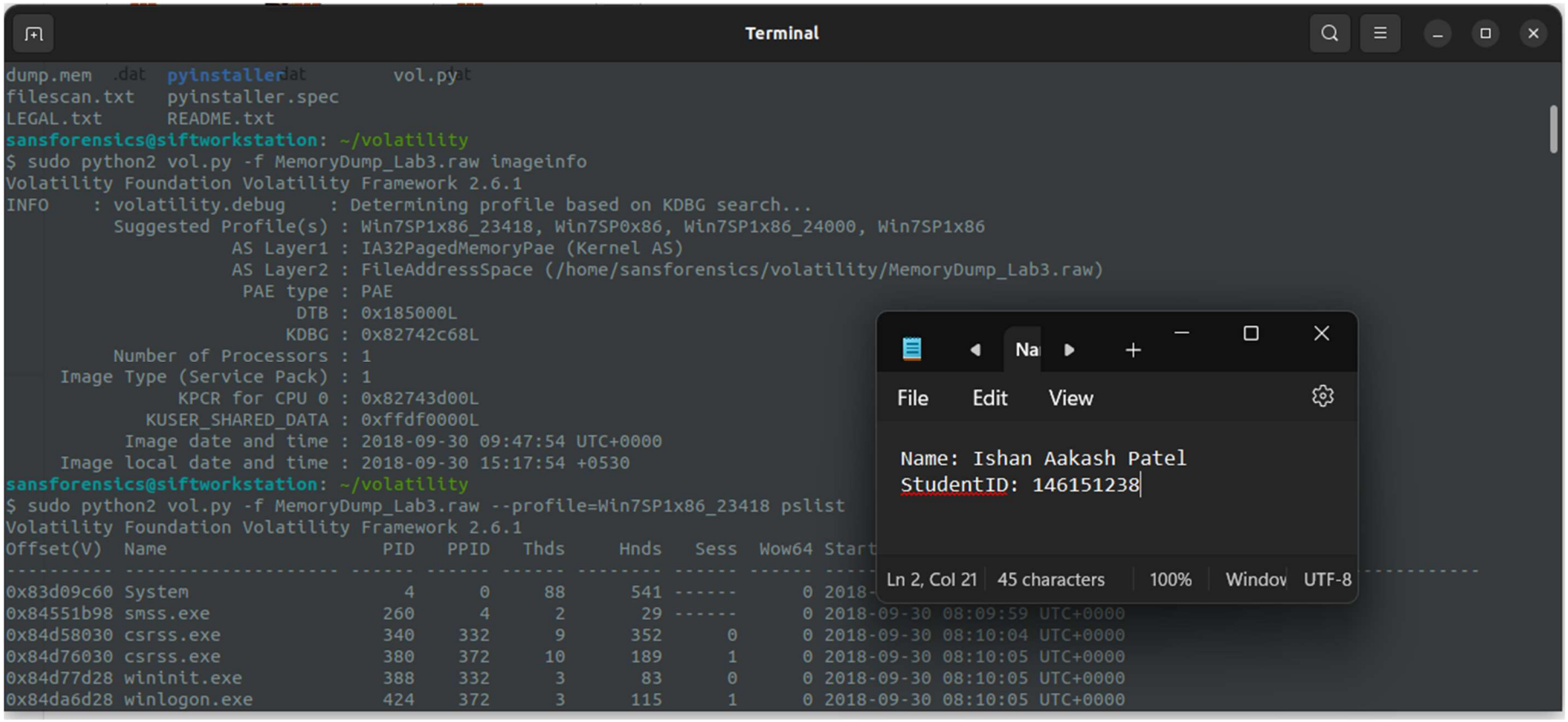
Part -1 is completed here as I no longer have to install the malware and scan my own image.

Part – 2 Memlabs Challenge 3

I have done this challenge in SIFT workstation.

Step 1 : Get the image info

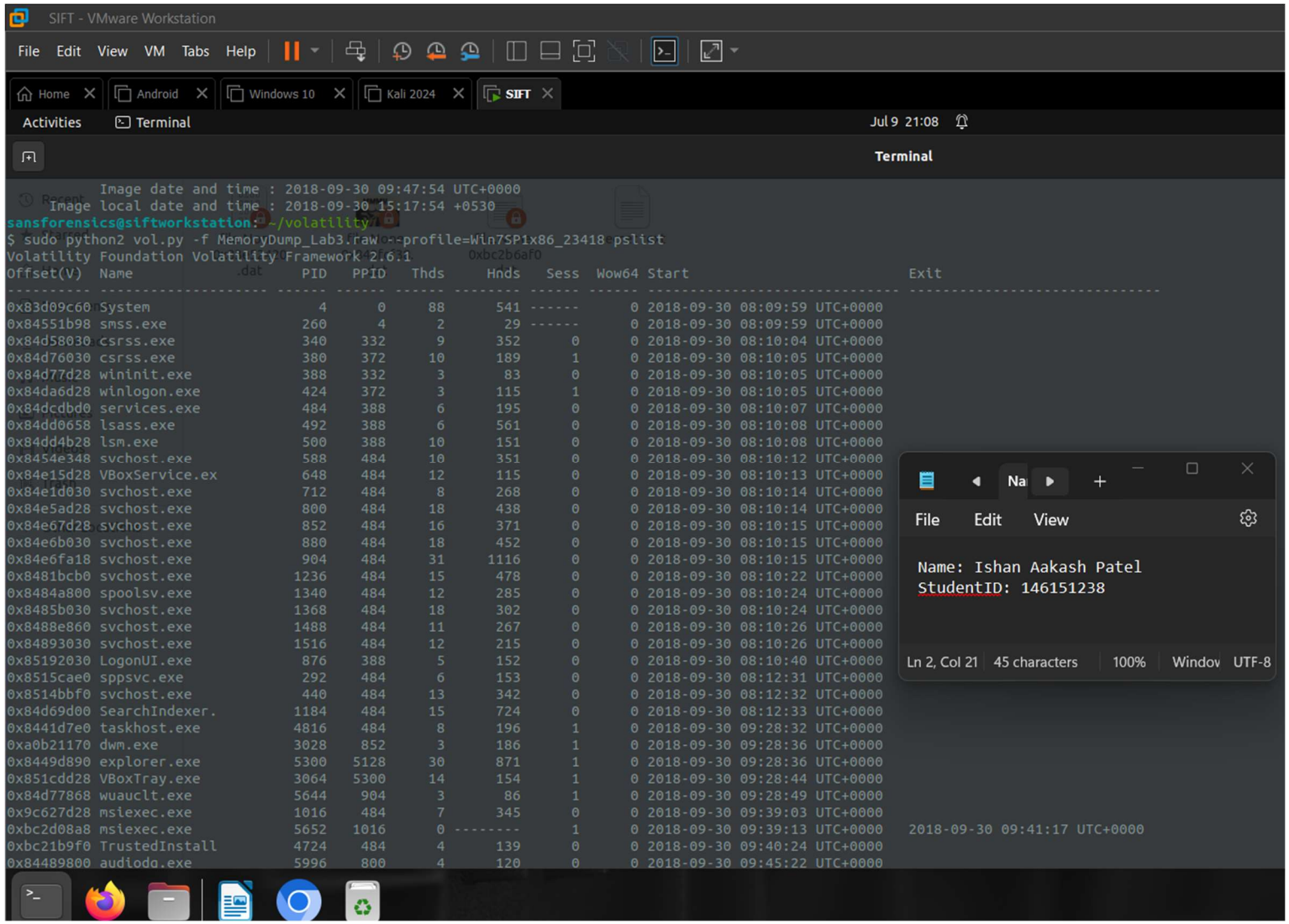
Command : sudo python2 vol.py -f MemoryDump_Lab3.raw imageinfo

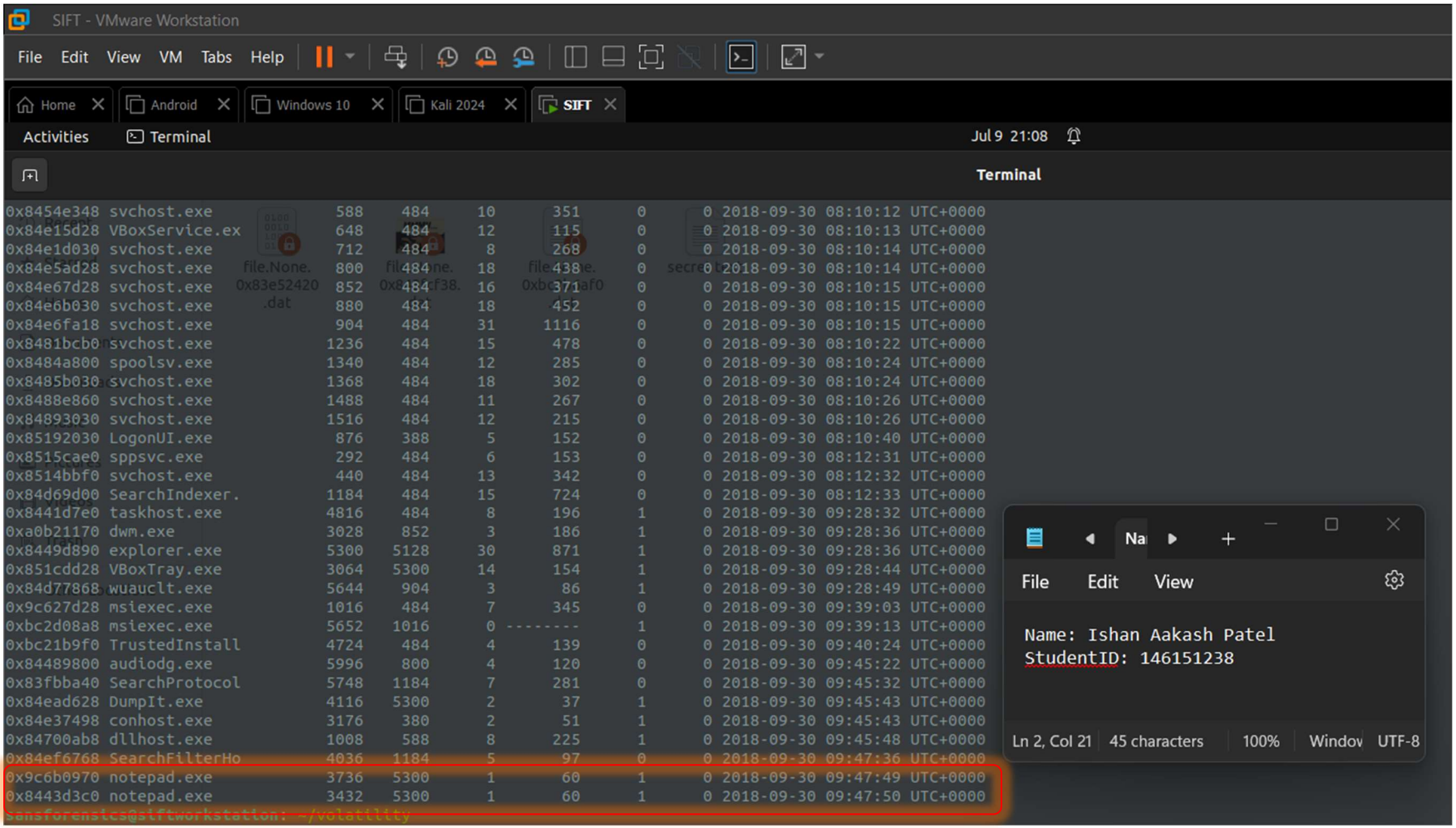


First profile is Win7SP1x86_23418

Step 2: Using pslist find the malicious script

Command : sudo python2 vol.py -f MemoryDump_Lab3.raw --Win7SP1x86_23418 pslist

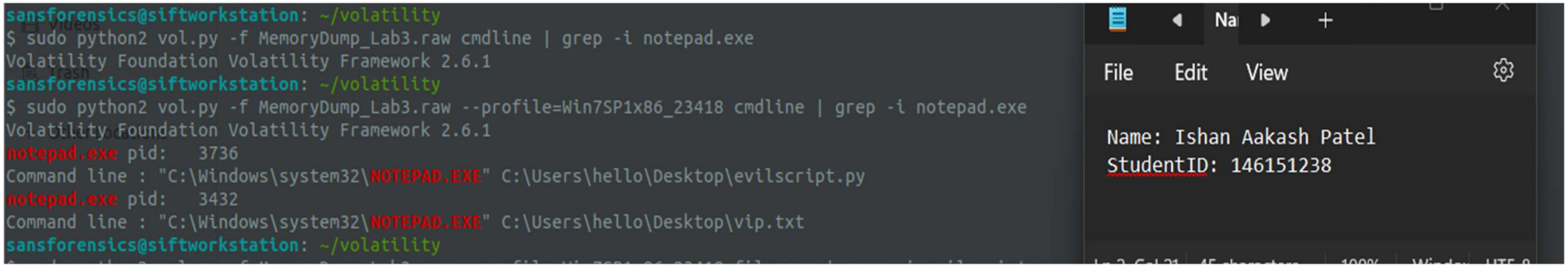




Two instances found of notepad.exe, now lets look at the command-line of this…

Step 3 : Command-line arguments of notepad processes using cmdline plugin.

Command : sudo python2 vol.py -f MemoryDump_Lab3.raw cmdline | grep -I notepad.exe

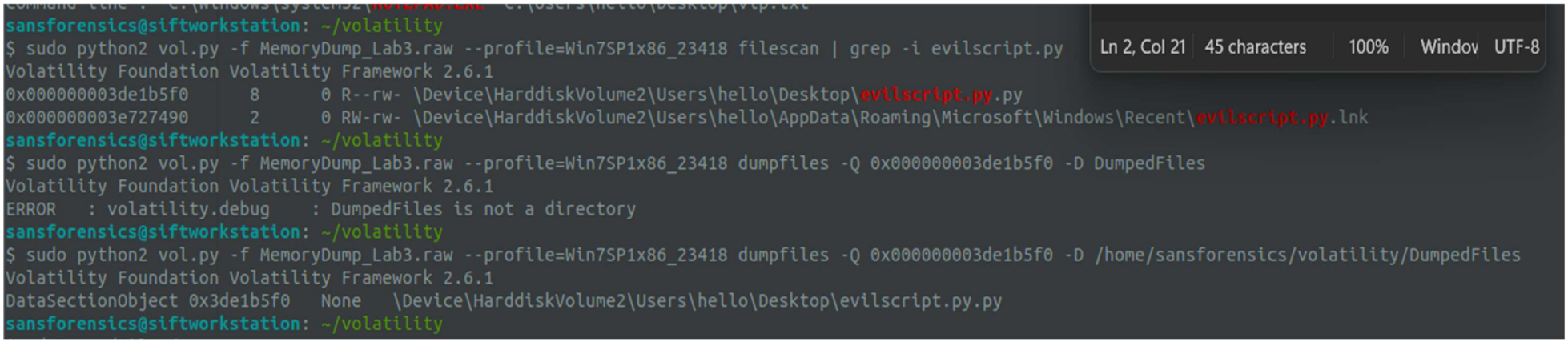


In the above outpust we found two malicious scripts 1) evilscript.py 2) vip.txt, now lets dump them…

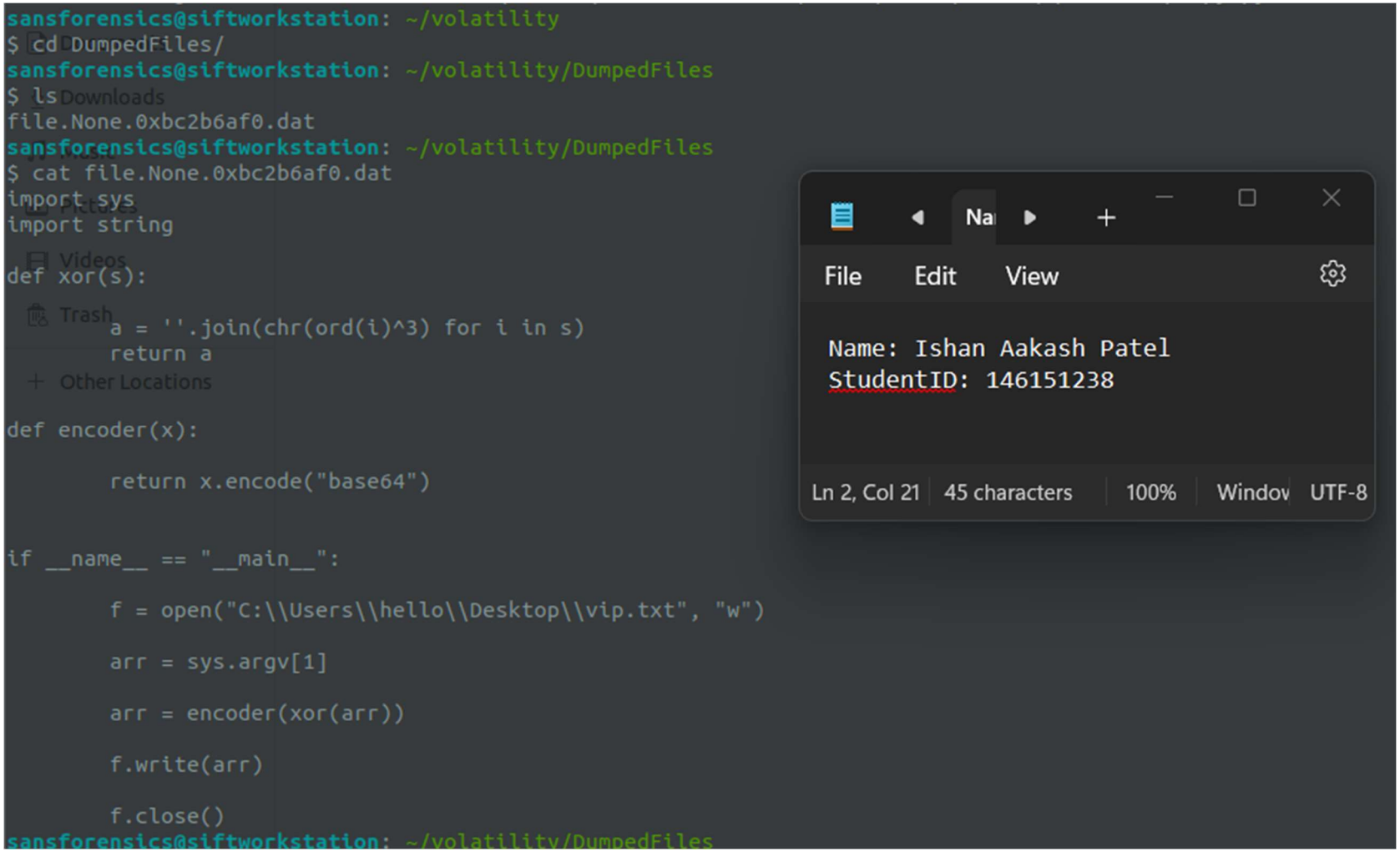
Step 4 : Dump the malicious scripts

Command : 1) sudo python2 vol.py -f MemoryDump_Lab3.raw --Win7SP1x86_23418 filescan | grep -i evilscript.py

2) sudo python2 vol.py -f MemoryDump_Lab3.raw --profile=Win7SP1x86_23418 dumpfiles -Q 0x000000003de1b5f0 -D /home/sansforensics/volatility/DumpedFiles



Step 5 : Open the script (evilscript.py)



The script receives a one-string argument as input and then xor its characters with value 0x3. Then, apply base64 encoding on the result and save it to the text file “vip.txt”.

Step 6 : Dump vip.txt

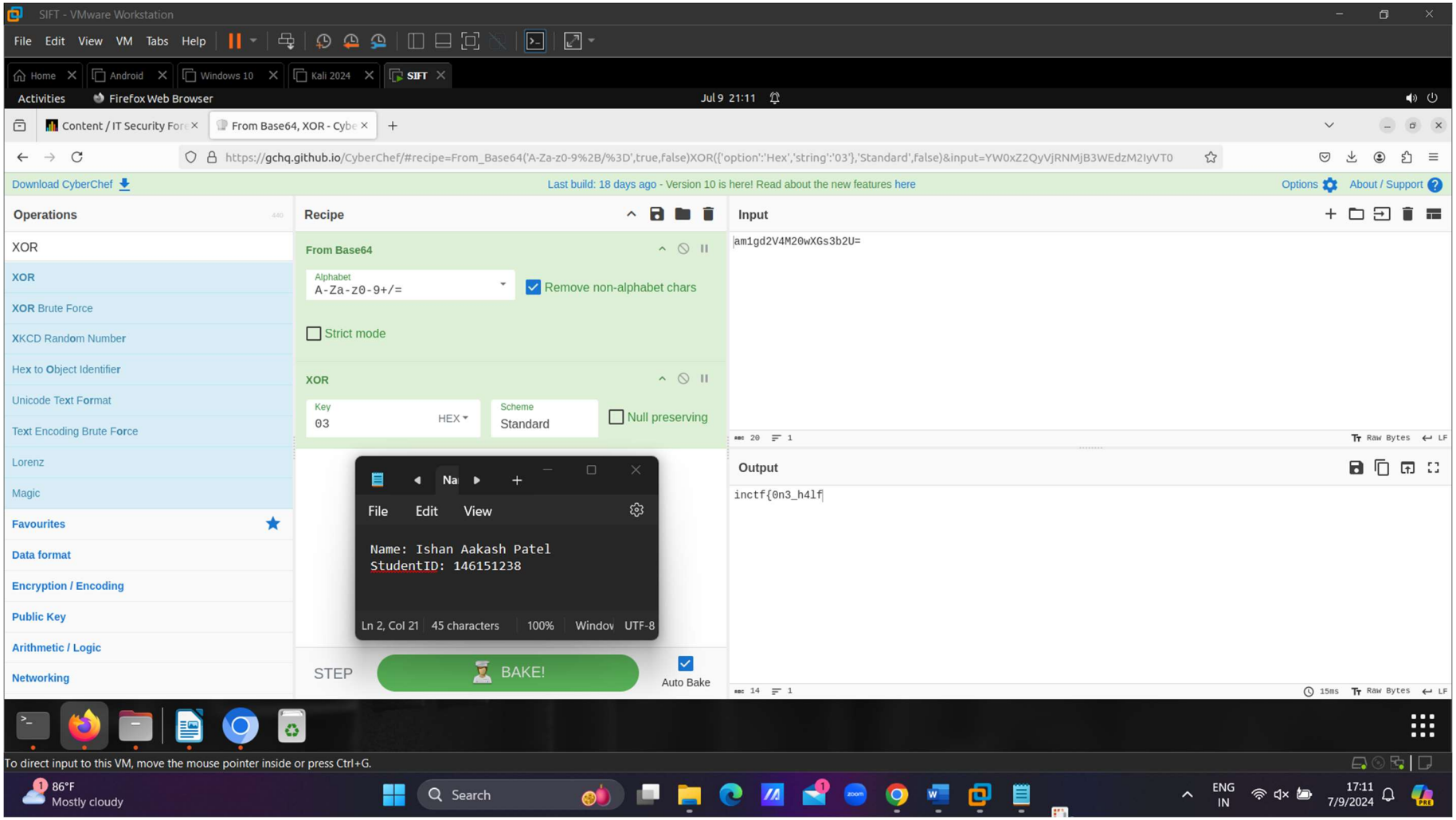
Command : 1) sudo python2 vol.py -f MemoryDump_Lab3.raw --Win7SP1x86_23418 filescan | grep -i vip.txt

2) sudo python2 vol.py -f MemoryDump_Lab3.raw --profile=Win7SP1x86_23418 dumpfiles -Q 0x000000003e727e50

-D /home/sansforensics/volatility/DumpedFiles

Now we can convert this string to flag through cyberchef.

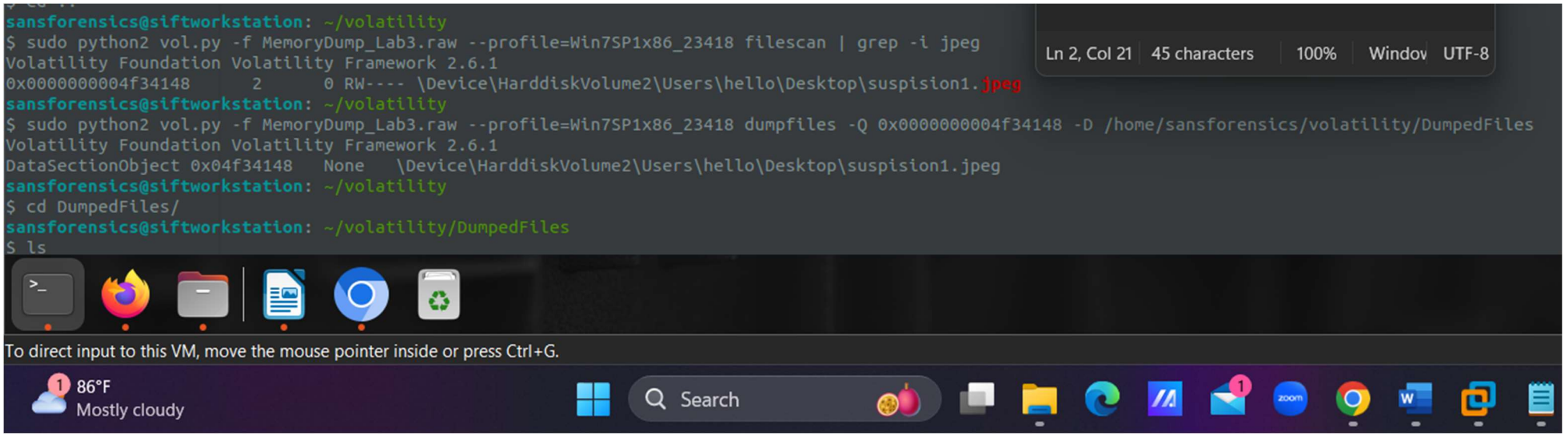
Step 7 : Cyberchef



Half flag : inctf{0n3_h4lf

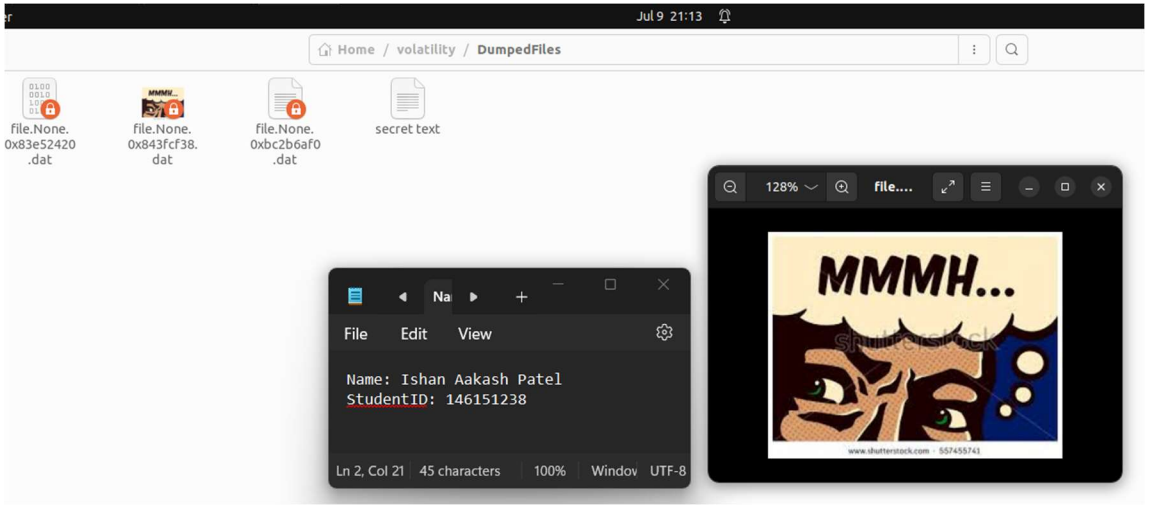
Step 8 : Recovering the hidden data: The second clue suggests using the steghide tool, which is designed for hiding data within images or audio files. According to the documentation, steghide supports JPEG, BMP, WAV, and AU file formats. Using these extensions, I searched for the file that conceals the data. I got lucky with a JPEG file, as shown in the following command.

Command : sudo python2 vol.py -f MemoryDump_Lab3.raw --profile=Win7SP1x86_23418 filescan | grep -i jpeg



After dumping the file here is the output.

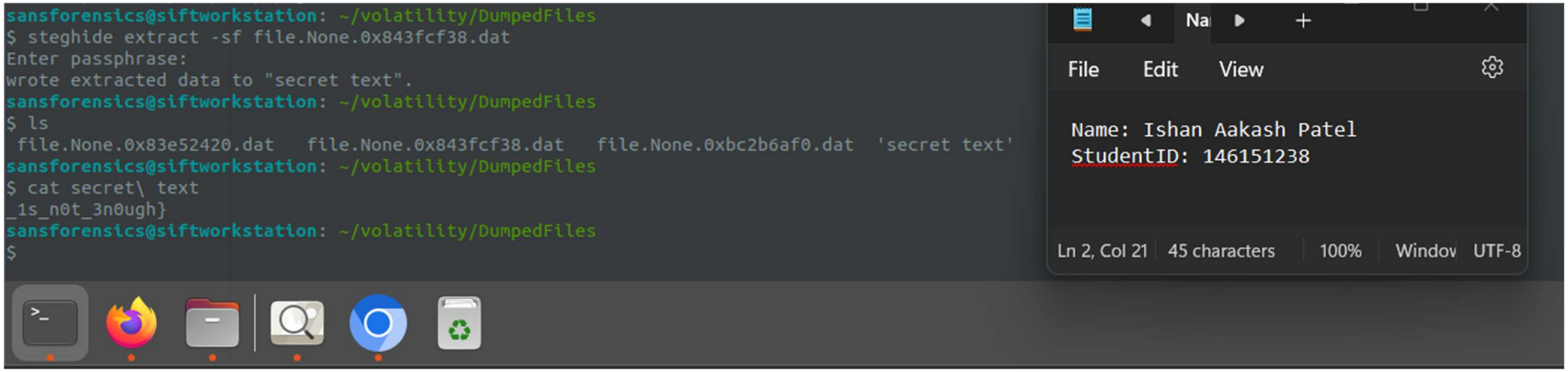
Step 9 : Output



Step 10 : Now use steghide tool to extract the hidden data.

Password was the first half of the flag

Second flag : _1s_n0t_3n0ugh}



Learning Experience

To complete my forensics workstation setup and memory analysis project, I followed a detailed guide provided by Blue Cape Security. Setting up the workstation involved installing essential tools like Volatility and Cyber Triage, ensuring they were functional for memory and malware analysis. Throughout the process, I documented each step with screenshots, ensuring to capture my desktop with the date/time visible, and showing my logged-in account details.

Using Volatility, I analyzed memory dumps from the MemLabs challenges, applying techniques learned from online resources. This practical exercise not only honed my technical skills but also deepened my understanding of forensic investigation methodologies. Reflecting on this experience, I gained insights into the complexities of digital forensics, particularly in analyzing memory artifacts to uncover potential security breaches or malicious activities. This project underscored the importance of meticulous documentation and methodical analysis in cybersecurity investigations, skills that are crucial in today's digital landscape.