Roll No	4716
Name	Amal Sathyan
Subject	Cyber and Information Security- II (Cyber Forensics)

Thakur Educational Trust's (Regd.)
THAKUR COLLEGE OF SCIENCE & COMMERCE
AUTONOMOUS COLLEGE, PERMANENTLY AFFILIATED TO UNIVERSITY OF MUMBAI
NAAC Accredited Grade 'A' (3rd Cycle) & ISO 9001: 2015 (Certified)
Best College Award by University of Mumbai for the Year 2018-2019
CELEBRATING 25 YEARS OF GLORY

Department of Computer Science

Journal Certificate

Class: M.Sc. Computer Science – Part II (Semester: III)

Roll No: 4716

Academic Year: 2023-2024

This is to certify that the work entered in this journal is the work of Mr Amal Sathyan, who has worked for academic year 2023-2024 in the computer laboratory. He / She has completed prescribed practical of following course satisfactorily.

Course Title: Cyber and Information Security- II (Cyber Forensics)

Teacher Incharge **Head of Department**

Date: **Examiner**

INDEX

1.Aim: Write a program to take backup of mysql database

Step 1: `@echo off`
This command turns off the default behavior of echoing each command to the console before executing it. It makes the script output cleaner by not displaying each command.

Step 2: `cd /d "C:\Program Files\MySQL\MySQL Server 8.0\bin"`
This command changes the working directory to the MySQL server's bin folder using the `cd` (Change Directory) command. The `/d` option ensures that it changes both the drive and directory.

Step 3: Setting MySQL Credentials:
Two environment variables are set to store the MySQL server credentials:
`mysql_user=root`: This variable stores the MySQL username (root in this case).
`mysql_password=Bunty@123`: This variable stores the MySQL password.

Step 4: Setting Backup File Parameters:
Two environment variables are set to define the backup file's parameters:
`backup_path`: This variable stores the path where the backup file will be saved (`C:\Users\Ashutosh Mishra\OneDrive\Desktop\backup_cis`).
`backup_name`: This variable stores the name of the backup file (`myalldatabases`).

Step 5: Backup Creation:
The `mysqldump` command is used to create a MySQL database backup.
`user=%mysql_user%`: Specifies the MySQL username.
`password=%mysql_password%`: Specifies the MySQL password.
`alldatabases`: Dumps all databases.
`routines`: Includes stored routines (e.g., functions and procedures).
`events`: Includes events (scheduled tasks).
`resultfile="%backup_path%\%backup_name%.sql"`: Specifies the path and name of the backup file.

Step 6: Checking for Errors:

The script checks the value of the `%ERRORLEVEL%` environment variable, which stores the exit code of the last executed command.

If `%ERRORLEVEL%` is not equal (`neq`) to `0`, it means there was an error during the backup creation.

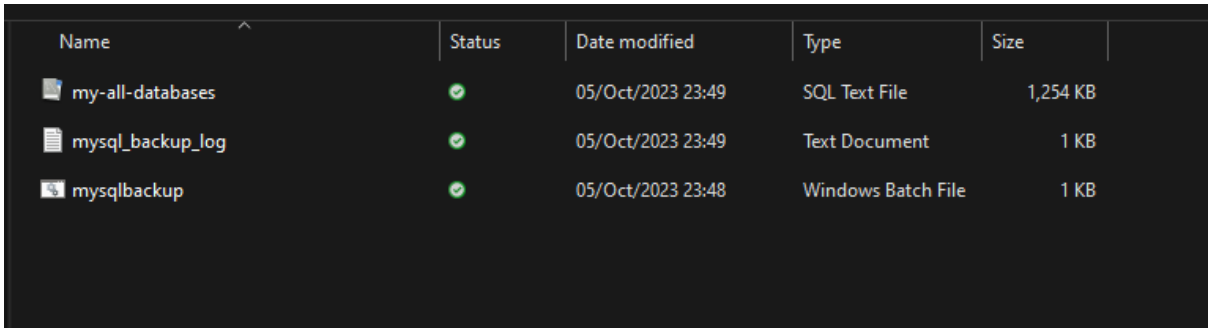
If there's an error, it appends an error message to the `mysql_backup_log.txt` file in the specified backup path.

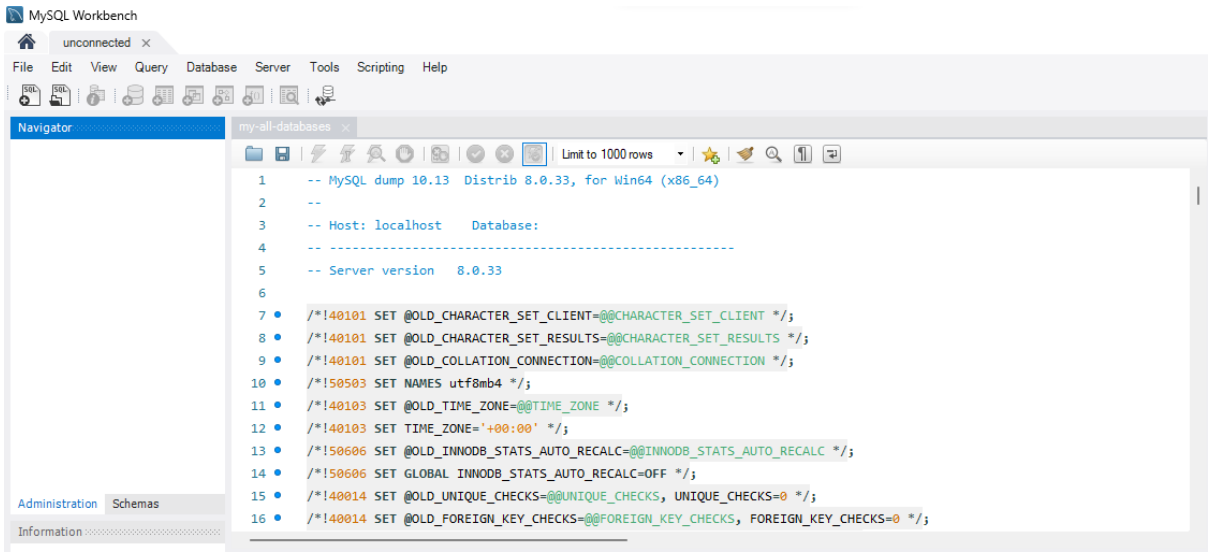
If there's no error (`else`), it appends a success message to the same log file.

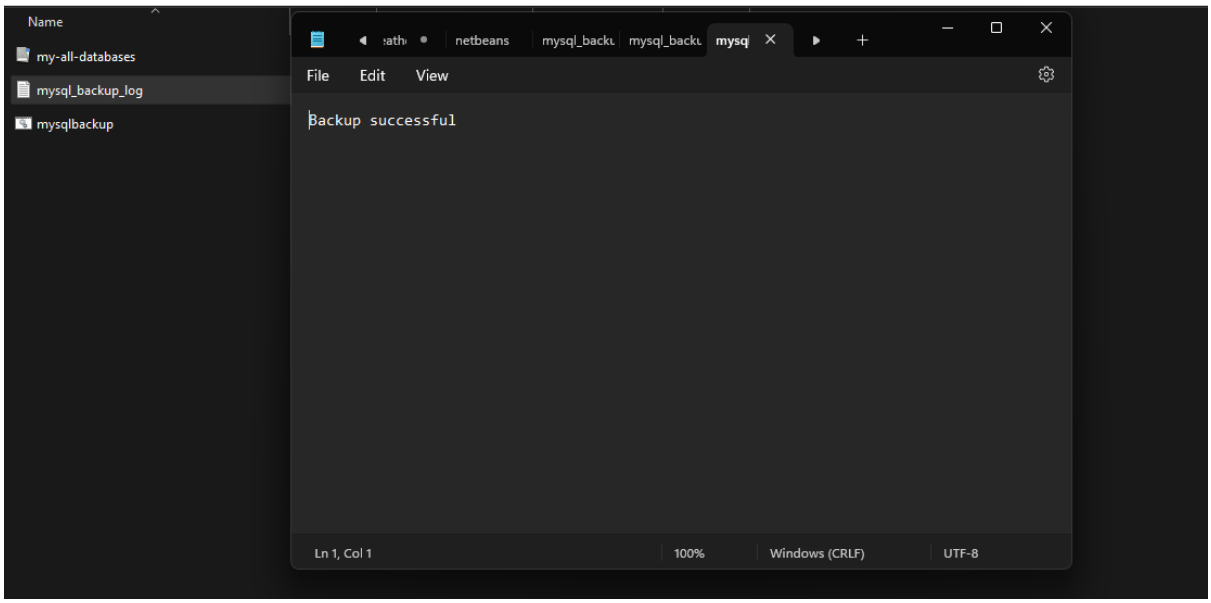
Code:

```
@echo off

rem Path to MySQL server bin folder
cd /d "C:\Program Files\MySQL\MySQL Server 8.0\bin"

rem Credentials to connect to MySQL server
set mysql_user=root
set mysql_password=Bunty@123

rem Backup file name generation
set "backup_path=C:\Users\Ashutosh Mishra\OneDrive\Desktop\backup_cis"
set "backup_name=my-all-databases"

rem Backup creation
mysqldump --user=%mysql_user% --password=%mysql_password% --all-databases --
routines --events --result-file="%backup_path%\%backup_name%.sql"
if %ERRORLEVEL% neq 0 (
    (echo Backup failed: error during dump creation) >> "%backup_path%\mysql_b
ackup_log.txt"
) else (
    (echo Backup successful) >> "%backup_path%\mysql_backup_log.txt"
)
```

Output:







Conclusion:

This batch script is designed to automate the process of creating a MySQL database backup with certain options and storing it in a specified directory.

It sets MySQL credentials and backup file parameters as environment variables.

It uses the `mysqldump` command to create the backup, including all databases, routines, and events.

It checks for errors during the backup creation and logs the result in a text file.

To troubleshoot any issues with this script, ensure that the MySQL server is running, the specified paths are correct, and the MySQL user has the necessary privileges. Review the `mysql_backup_log.txt` file for any error messages or success confirmation after running the script.

2.Aim: Write a program to restore mysql database

Softwares required to perform practical
Mysql cmd client 8.0

Theory:-

It is important to restore a database from backup to recover from data loss or corruption. This can happen for a variety of reasons, including:

Human error: Accidental deletion or modification of data is a common occurrence.
Hardware failure: Hard drives, SSDs, and other storage devices can fail, causing data loss.
Software failure: Software bugs or errors can also cause data corruption.
Security breaches: Malicious actors can encrypt or delete data as part of a ransomware attack or other attack.
Natural disasters: Floods, fires, and other natural disasters can damage or destroy data storage devices.
If you do not have a database backup, you will not be able to recover any lost or corrupted data. This can be catastrophic for businesses, which rely on their databases to operate.

Restoring a database from backup allows you to recover your data to a specific point in time before the data loss or corruption occurred. This means that you can minimize the impact on your business and get back up and running quickly.

Steps:

Restoring a MySQL database from a backup is a common task in database management. Here are the general steps to restore a MySQL database from a backup:

Note: Before you begin, make sure you have a recent backup of your MySQL database.

1. Login into mysql cmd client

.

2. Create a New Database (Optional): If you want to restore the backup to a new database, create an empty database first. Replace `new_database` with your desired database name.

3. Select the Database: If you created a new database, switch to it.

4. Restore the Backup:

```
mysql -u username -p new_database < backup_file.sql
```

Replace `username` with your MySQL username, `new_database` with the database name, and `backup_file.sql` with the path to your backup file.

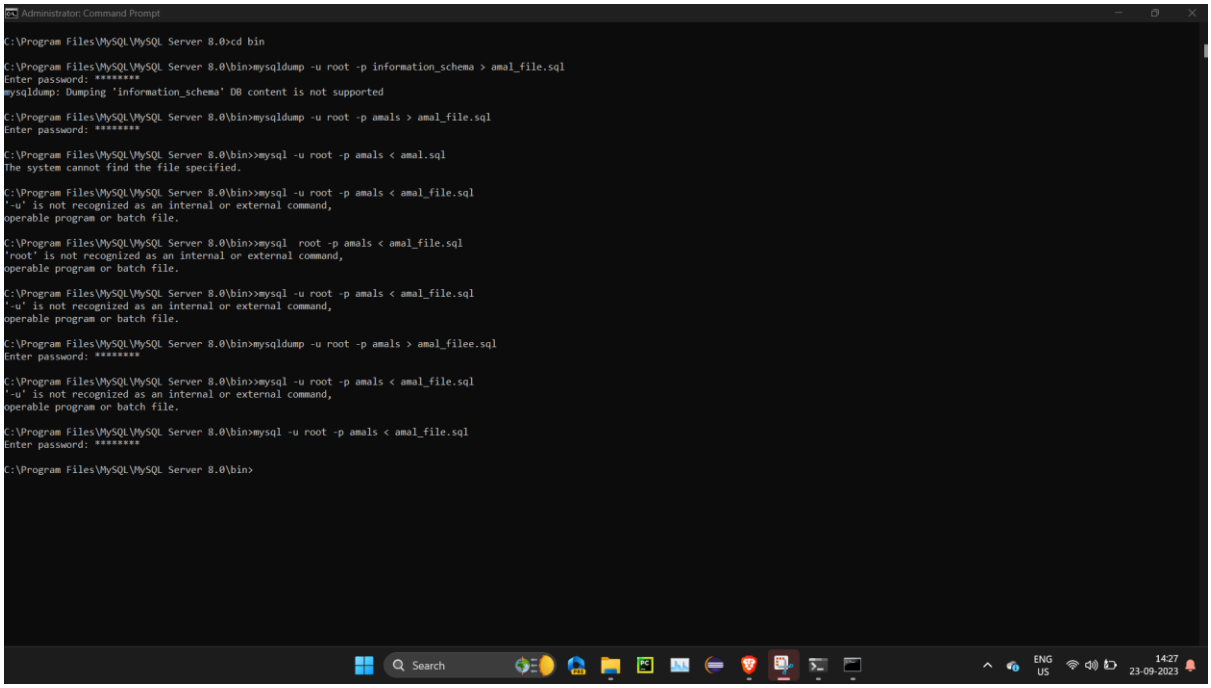
5. Verify the Restore: After the restore process is complete, you can verify that your data has been restored by querying the database:

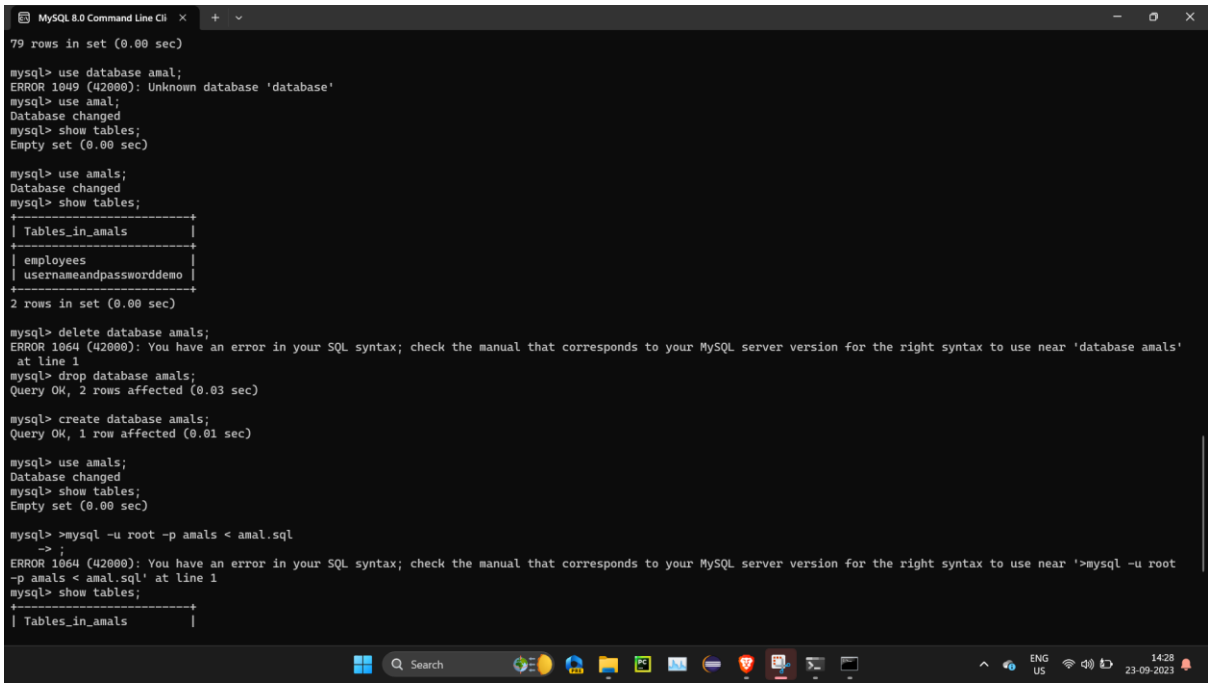
```
SELECT  FROM your_table;
```

Replace `your_table` with the name of a table in your database.

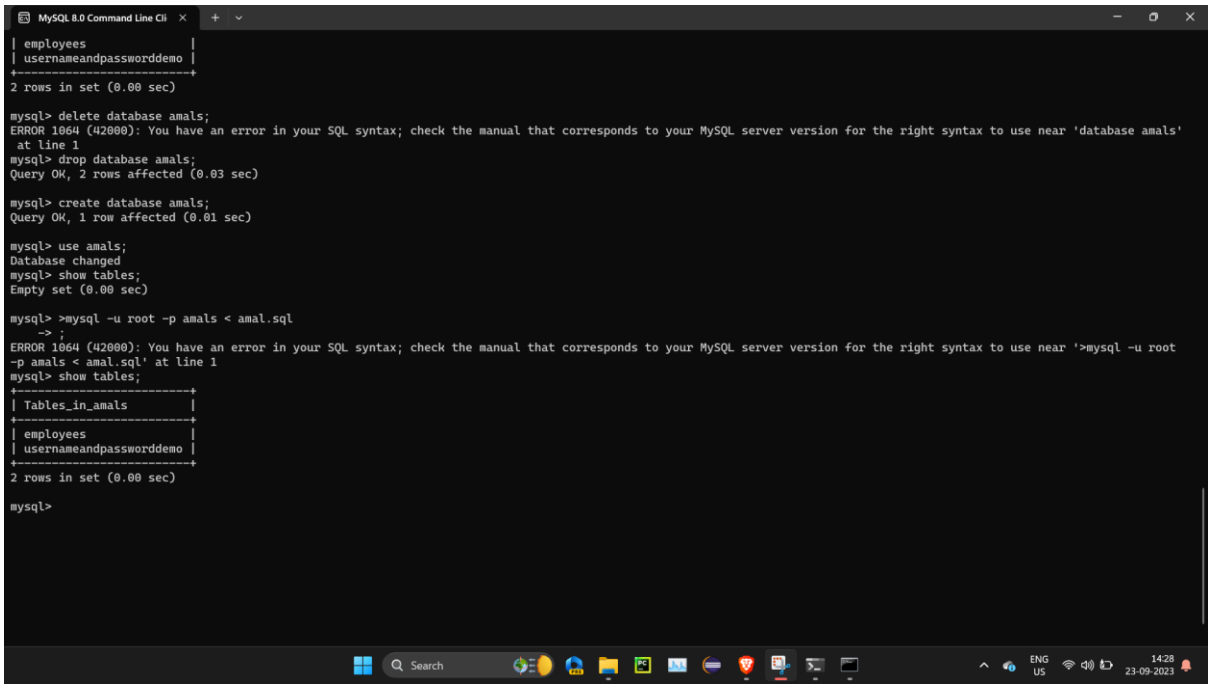
Output:

Creating backup





Verifying restore



Conclusion:

Restoring a MySQL database from backup is an important part of any data protection strategy. It allows you to recover from data loss or corruption and minimize the impact on your business.

To restore a MySQL database from backup, you can use the mysql command-line tool or a GUI tool such as phpMyAdmin. The process is simple and straightforward, but it is important to make sure that the database backup file is compatible with the version of MySQL you are using and that the new server has the same MySQL user accounts and permissions as the old server.

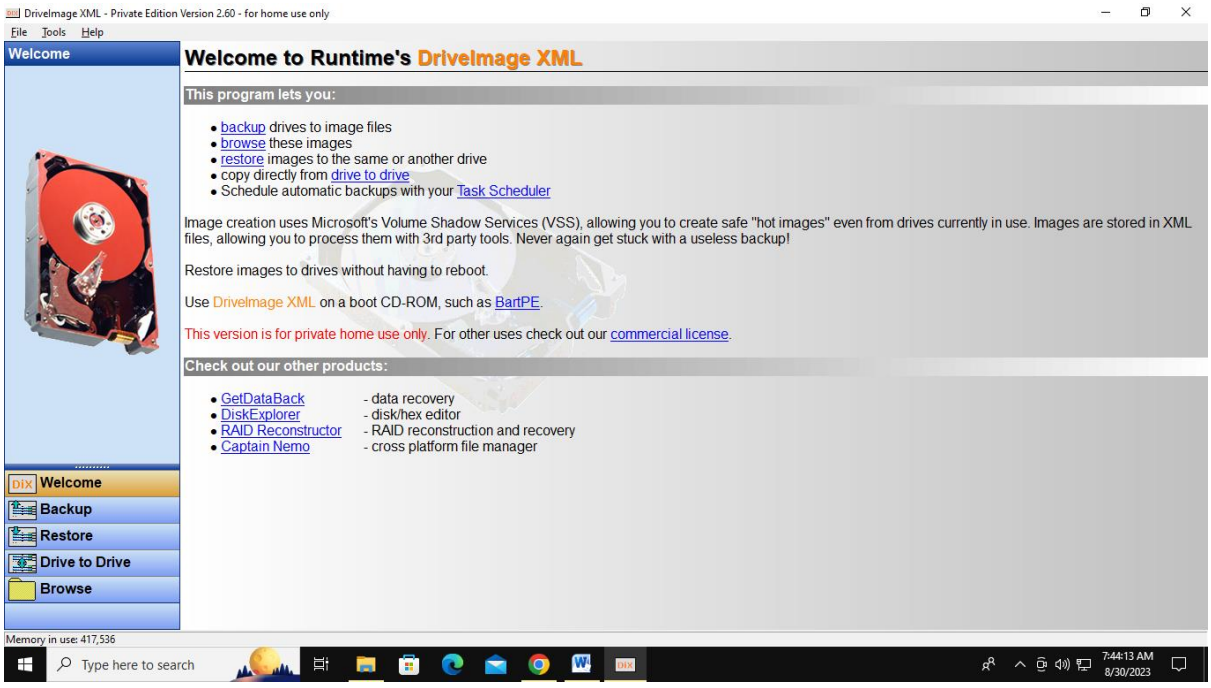
If you are restoring a large database, you may want to consider restoring it to a test environment first to verify that the restore was successful.

3.Aim: Use DriveImage xml to image a hard drive.

Software: DriveImage xml

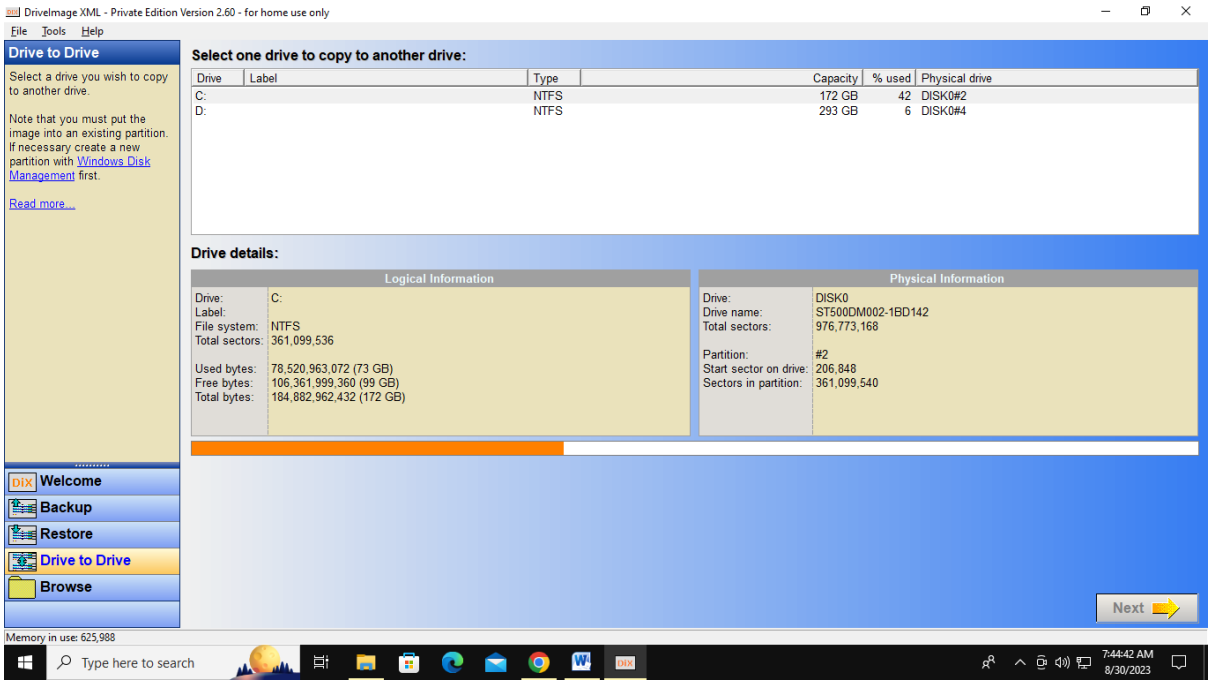
Theory: DriveImage XML is an easy to use and reliable program for imaging and backing up partitions and logical drives. It includes all the basic functions you expect from disk cloning software. This tool allows you to **copy directly from one drive to another**. This is helpful if you need to clone HDD to SSD or clone computer hard drive to external drive as a backup.

What's more, you can **back up any drive** on your computer to an XML image file or **restore an existing image to any drive** connected to the computer. Even if your computer is running, you can make a backup. And there is no need to restart when restoring the backup.



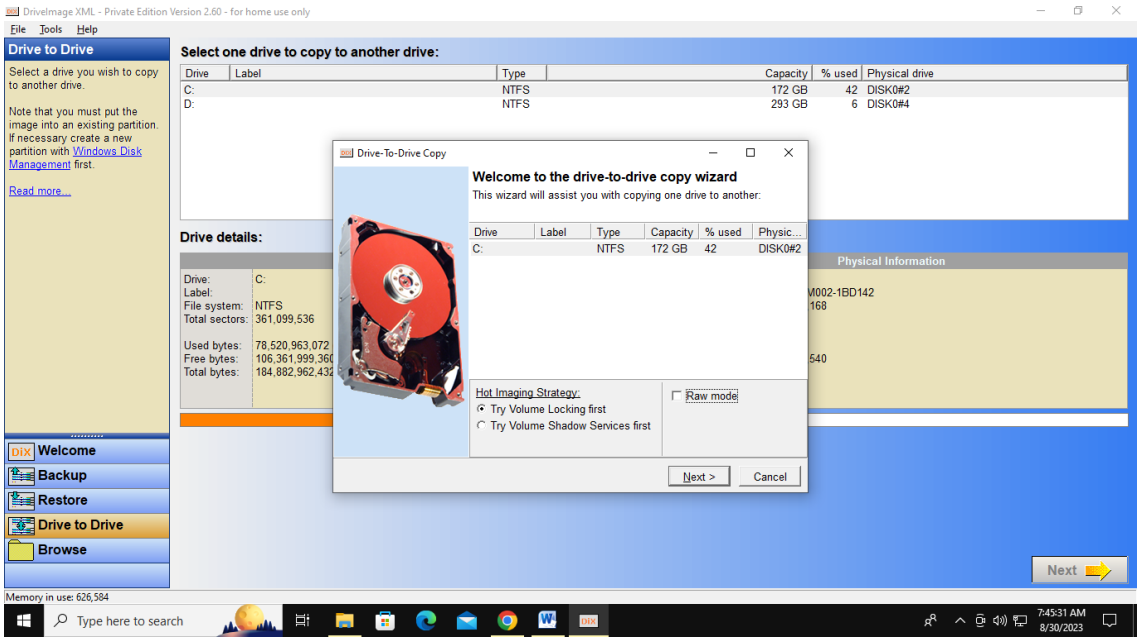
1. First, download **DriveImage XML** for free and install it as usual.

2. When you launch **DriveImage XML (DiX)**, click on the "**Drive to Drive**" button on the lower left of the screen. DriveImage XML will display all available source partitions and show information about their geometry below.

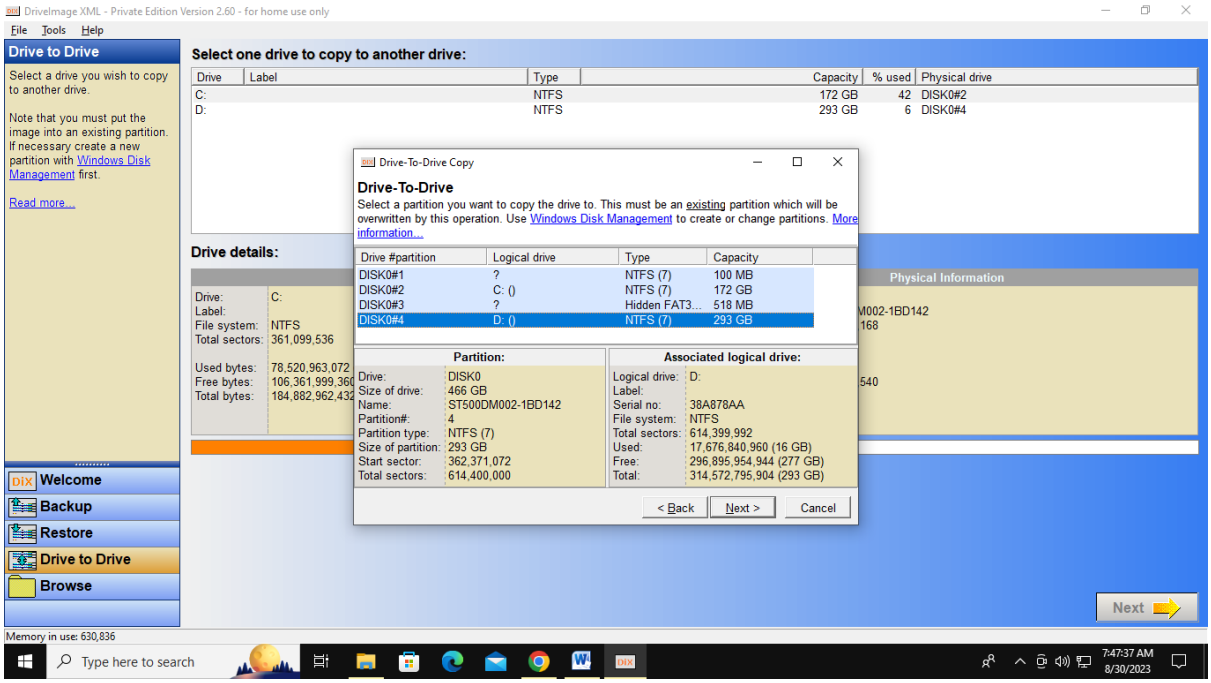


3.Select the boot drive you wish to copy to another drive and click **Next** to continue.

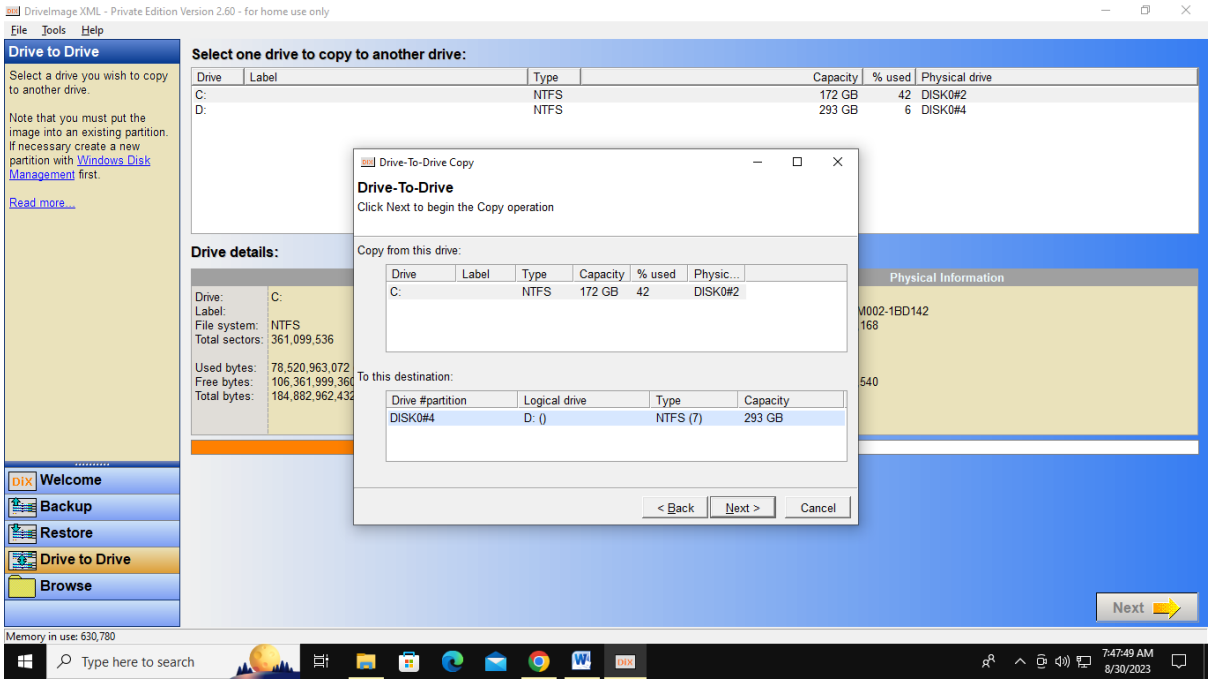
4.The drive you selected as the source drive in the previous step will be displayed again. And some options are provided. According to your needs, check the options and click **Next**.

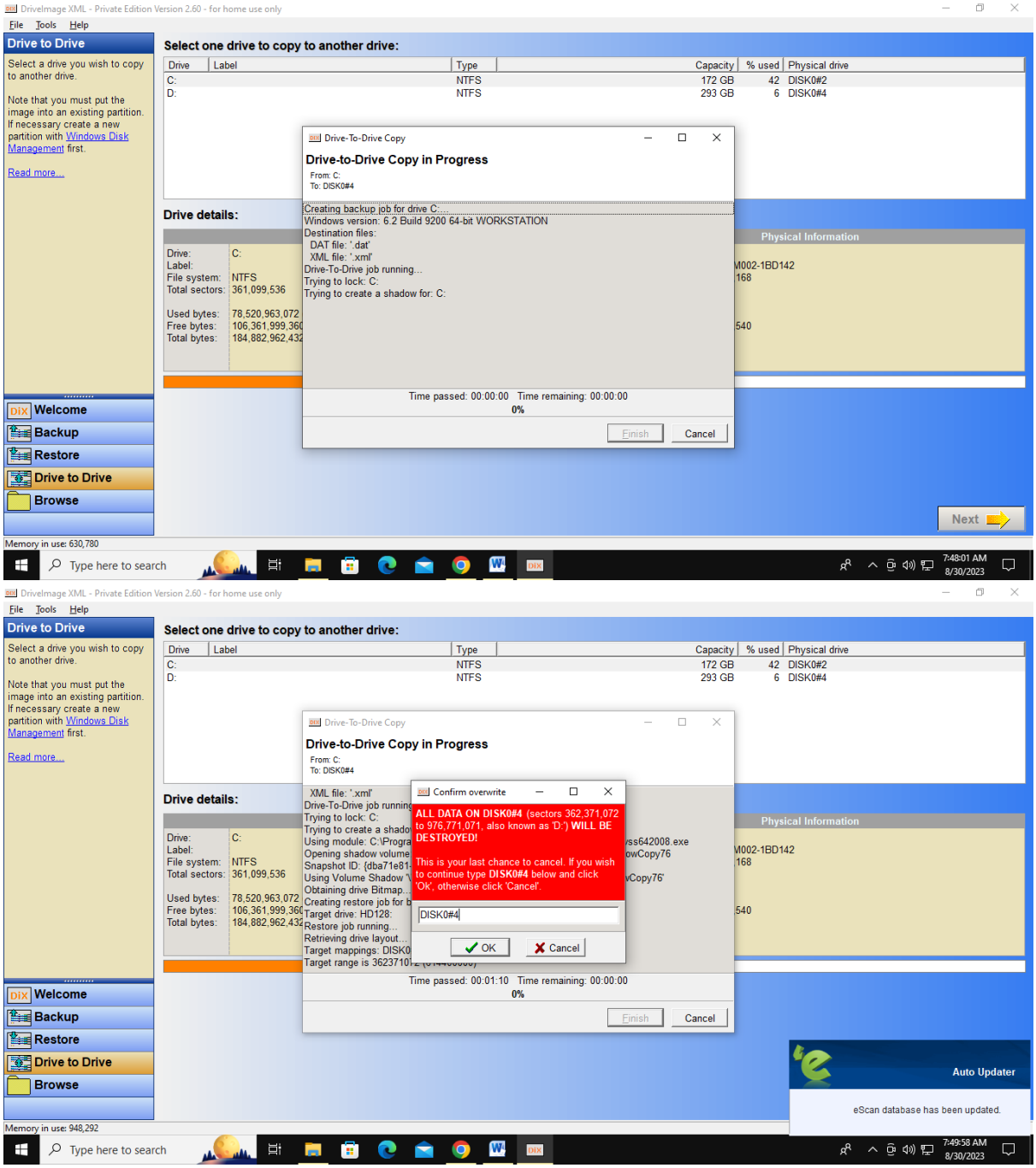


4.Then select the destination drive and click **Next**.

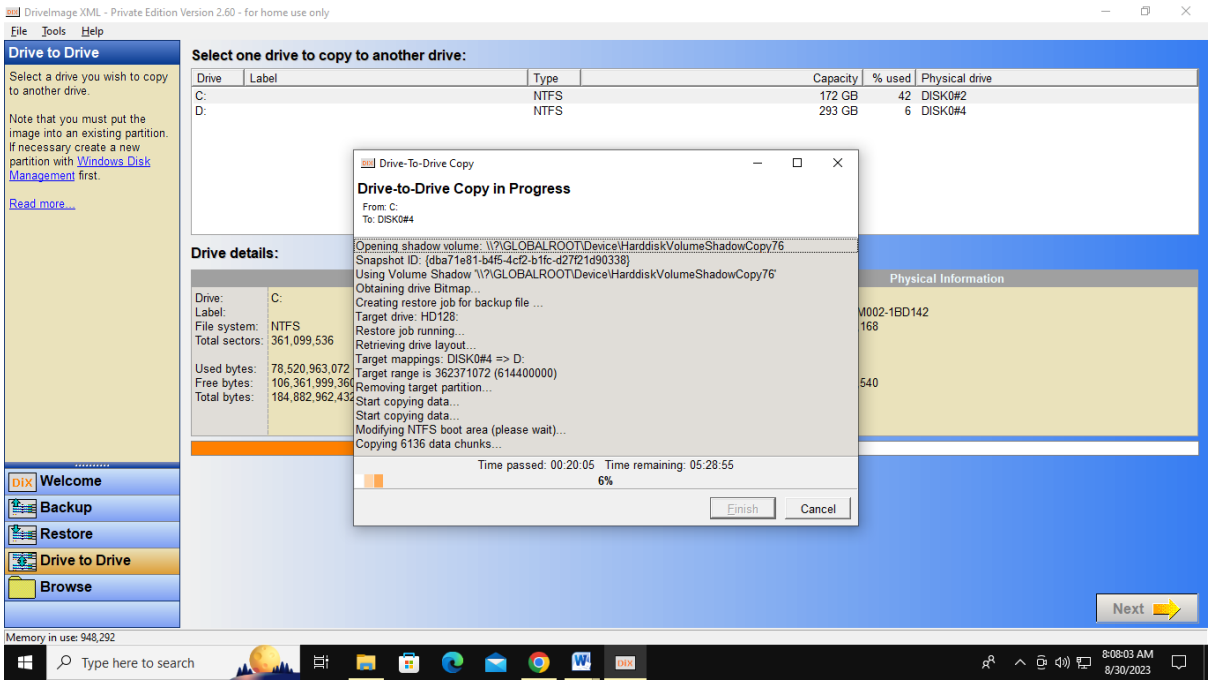


5.Then DriveImage XML will show you again which drives you selected as the source drive and as the destination drive. Make sure you selected the correct drives and click **Next** to begin the operation.




DriveImage XML - Private Edition Version 2.60 - for home use only
File Tools Help
Drive to Drive
Select a drive you wish to copy to another drive.
Note that you must put the image into an existing partition. If necessary create a new partition with Windows Disk Management first.
Read more...
Select one drive to copy to another drive:
Drive Label Type Capacity % used Physical drive
C: NTFS 172 GB 42 DISK0#2
D: NTFS 293 GB 6 DISK0#4
Drive-To-Drive Copy
Drive-to-Drive Copy in Progress
From: C:
To: DISK0#4
Creating backup job for drive C:...
Windows version: 6.2 Build 9200 64-bit WORKSTATION
Destination files:
DAT file: '.dat'
XML file: '.xml'
Drive-To-Drive job running...
Trying to lock: C:
Trying to create a shadow for: C:
Time passed: 00:00:00 Time remaining: 00:00:00
0%
Finish
Cancel
Drive details:
Physical Information
Drive: C:
Label:
File system: NTFS
Total sectors: 361,099,536
Used bytes: 78,520,963,072
Free bytes: 106,361,999,360
Total bytes: 184,882,962,432
Welcome
Backup
Restore
Drive to Drive
Browse
Next
Memory in use: 630,780
Type here to search
7:48:01 AM
8/30/2023

DriveImage XML - Private Edition Version 2.60 - for home use only
File Tools Help
Drive to Drive
Select a drive you wish to copy to another drive.
Note that you must put the image into an existing partition. If necessary create a new partition with Windows Disk Management first.
Read more...
Select one drive to copy to another drive:
Drive Label Type Capacity % used Physical drive
C: NTFS 172 GB 42 DISK0#2
D: NTFS 293 GB 6 DISK0#4
Drive-To-Drive Copy
Drive-to-Drive Copy in Progress
From: C:
To: DISK0#4
Confirm overwrite
ALL DATA ON DISK0#4 (sectors 362,371,072 to 976,771,071, also known as 'D:') WILL BE DESTROYED!
This is your last chance to cancel. If you wish to continue type DISK0#4 below and click 'Ok', otherwise click 'Cancel'.
DISK0#4
OK
Cancel
Time passed: 00:01:10 Time remaining: 00:00:00
0%
Finish
Cancel
Drive details:
Physical Information
Drive: C:
Label:
File system: NTFS
Total sectors: 361,099,536
Used bytes: 78,520,963,072
Free bytes: 106,361,999,360
Total bytes: 184,882,962,432
Welcome
Backup
Restore
Drive to Drive
Browse
Auto Updater
eScan database has been updated.
Memory in use: 948,292
Type here to search
7:49:58 AM
8/30/2023



6.Then a DriveImage XML bootable clone will be performed. Once completed you will be notified whether or not the copy process was successful. Press **Finish**

Conclusion:

In conclusion, using DriveImage XML to image a hard drive is a reliable and efficient method for creating backups and cloning drives

4.AIM: Write a program to create a log file

Software: JAVA IDE

Windows Event Logs:

The Windows operating system can generate event logs in response to activities on its hardware or software components. Network security and operations analysts often use specialized software tools to aggregate and analyze these logs, detect patterns and trends, and respond to incidents or potential issues. Windows classifies events into six categories:

1. Application Logs: These logs are generated when an event occurs within an application. They assist developers in understanding application behavior during development and prior to release.

2. Directory Service Logs: Computers configured to respond to security authentication requests within a Windows Server domain, known as domain controllers, generate these logs. They record user privilege changes, authentication operations, and requests in Windows Active Directory.

3. DNS Server Logs: DNS servers match website hostnames with their corresponding IP addresses. DNS server logs record activity related to DNS server operations, such as website navigation.

4. File Replication Service Log: This log is available only for domain controllers and records information about file replications.

5. Security Log: Security logs are created in response to security events, including failed log-ins, password changes, authentication requests, and file deletions. Network administrators configure which events are logged in the security log.

6. System Log: System logs record events within the operating system itself, such as driver errors during startup, sign-in/sign-out events, and other system activity.

Java Program for Creating a Log File:

```java
package logfile;
import java.io.FileWriter;
import java.io.IOException;
import java.io.PrintWriter;
public class LogFile {
   public static void main(String[] args) {
      try {
         // Command to retrieve process information and save it to a file
         String cmd = "WMIC /OUTPUT:C:\\LIST.TXT PROCESS WHERE \"Name='adminservices.exe' OR Name='applicationFramehost.exe' OR Name='audiodg.exe' OR Name='bdagent.exe' OR Name='bdntwrk.exe'\" get name,processid,creationdate/format:table";
         // Create a batch file to execute the command
         FileWriter fos = new FileWriter("C:\\mycmd.bat");
         fos.write(cmd);
         fos.close();
         // Execute the batch file using cmd.exe
         Runtime rt = Runtime.getRuntime();
         Process process = rt.exec("cmd.exe /c start C:\\mycmd.bat");
         // Wait for the process to complete
         int exitCode = process.waitFor();
          // Check if the command was executed successfully
         if (exitCode == 0) {
            System.out.println("Command executed successfully.");
         } else {
            System.out.println("Command execution failed with exit code: " + exitCode);
         }
        // Write your name to the console
         System.out.println("Logged by: Ashutosh A Mishra");
         // Write your name to the log file
PrintWriter writer = new PrintWriter(new FileWriter("C:\\LIST.TXT", true)); // Open file in append mode
         writer.println("Logged by: Ashutosh A Mishra");
```

```
            writer.close();
        } catch (IOException | InterruptedException e) {
            e.printStackTrace();
        }
    }
}
```

Instructions for Running the Program in NetBeans IDE:

To create and run the provided Java program in NetBeans IDE to generate a log file, follow these steps:

Step 1: Open NetBeans IDE as Administrator and create a new Java project.

1. Launch NetBeans IDE.
2. Go to "File" > "New Project..."
3. In the "New Project" dialog, select "Java" under "Categories" and "Java Application" under "Projects." Click "Next >."
4. Enter a project name (e.g., "LogFile") and choose a location for your project. Click "Finish."

Step 2: Create a new Java class.

1. In the "Projects" window on the left, right-click on the "Source Packages" folder within your project.
2. Select "New" > "Java Class..."
3. Enter a class name (e.g., "LogFile") and click "Finish."

Step 3: Replace the contents of the Java class with the provided code.

Copy and paste the Java code you provided into the "LogFile.java" class file, replacing any existing code.

Step 4: Save the changes.

Make sure to save the Java class file by clicking "File" > "Save" or pressing Ctrl + S (Cmd + S on macOS).

Step 5: Run the Java program.

1. Right-click on the "LogFile.java" file in the "Projects" window.
2. Select "Run File" from the context menu.

Step 6: Verify the results.

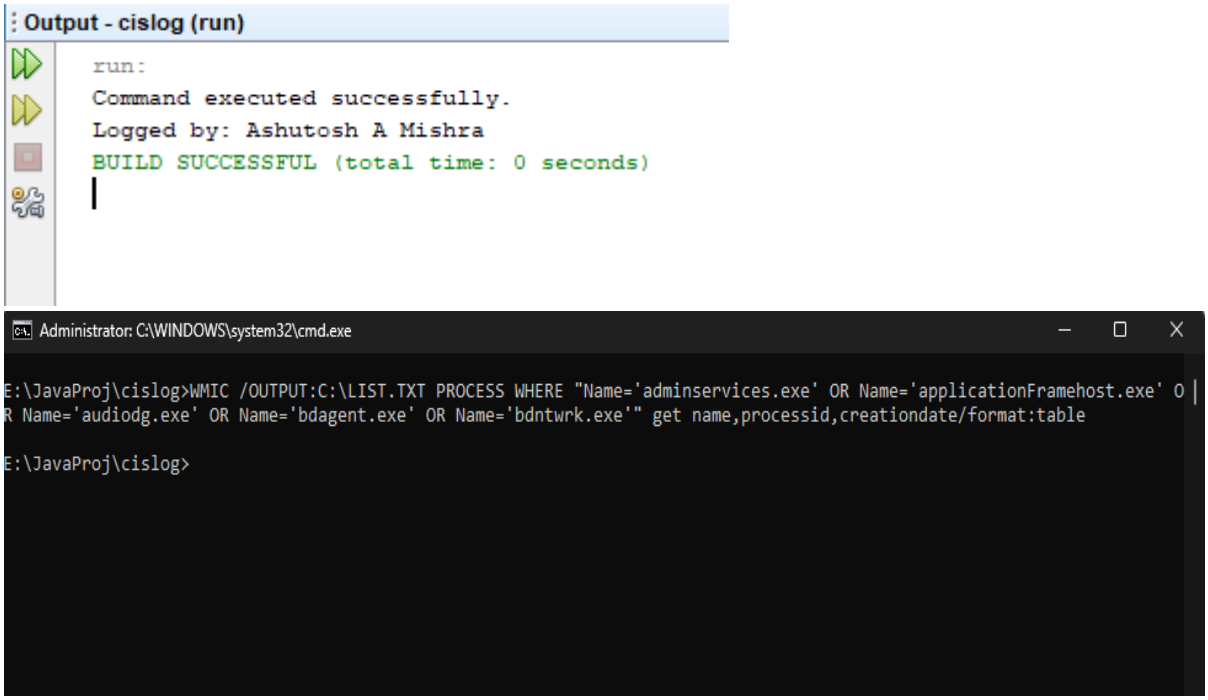
After running the program, check the following:

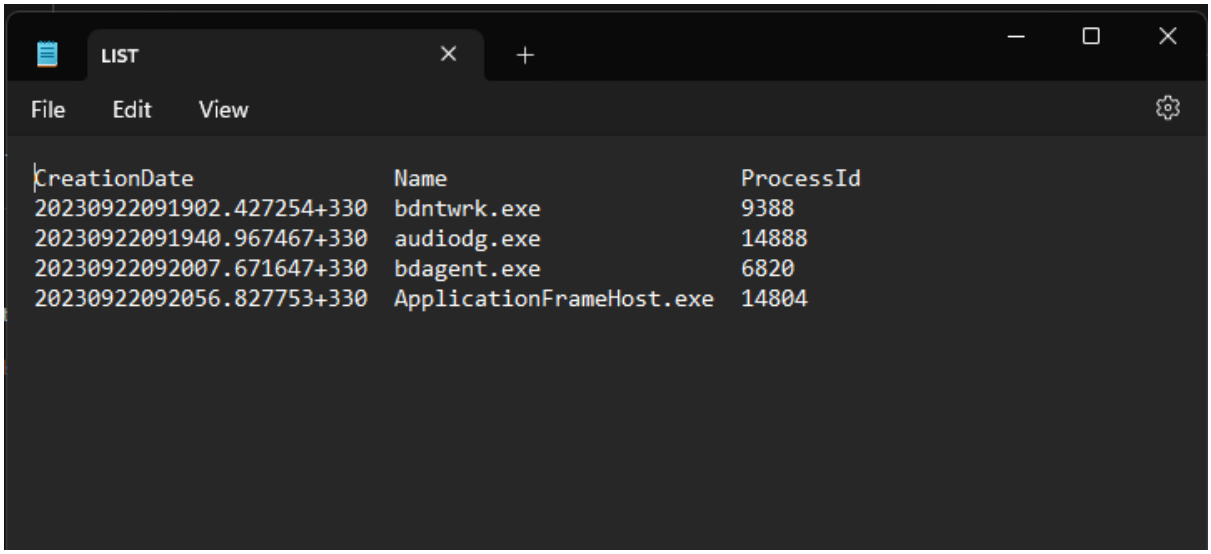
- A batch file named "mycmd.bat" should be created in the root of your C: drive (C:\).

- The batch file should execute the provided command using cmd.exe, which will generate an output file named "LIST.TXT" in the root of your C: drive.
- The program will print "Command executed successfully." or "Command execution failed with exit code: X" in the Output window of NetBeans, depending on whether the command was successful or not.
- Your name ("Logged by: Ashutosh A Mishra") should be appended to the "LIST.TXT" file.

Note: Make sure you have the necessary permissions to create files in the C: drive and execute commands using cmd.exe. Additionally, ensure that the specified Windows processes exist on your system or modify the list of processes in the `cmd` variable as needed.

OUTPUT:





Conclusion:

This Java program automates the generation of a log file on a Windows system. It runs a predefined command using WMIC, captures the output in a file named LIST.TXT, The program demonstrates a practical approach to event log generation, highlighting the importance of event logs for system analysis, security, and troubleshooting. It serves as a foundation for leveraging event logs to enhance system management and analysis on Windows-based systems.

5.Aim: - Write a program to find a file in a directory

Software: JAVA IDE

Theory:

Pattern matching is a fundamental concept in shell scripting and file manipulation, commonly used to locate files based on specific criteria. In Unix-like systems, the asterisk () is a widely used wildcard character that matches any number of characters. For instance, the command `ls .html` lists all files in the current directory that end with `.html`. The `java.nio.file` package in Java provides programmatic support for pattern matching, allowing developers to perform similar tasks programmatically.

In the `java.nio.file` package, each file system implementation provides a `PathMatcher` to handle pattern matching operations. You can obtain a file system's `PathMatcher` by using the `getPathMatcher(String)` method in the `FileSystem` class. For example, the code snippet below retrieves the path matcher for the default file system and uses glob syntax:

```java
String pattern = ...;
PathMatcher matcher = FileSystems.getDefault().getPathMatcher("glob:" + pattern);
```

Java's `java.nio.file` package supports glob syntax, which is simple and flexible. However, you can also use regular expressions (regex) if needed. The `PathMatcher` interface defines a `matches` method that takes a `Path` argument and returns a boolean value, indicating whether the path matches the specified pattern.

If you prefer a different form of string-based pattern matching, you can create a custom `PathMatcher` class. In the examples provided, we use glob syntax.

CODE:

```
package cis5;

import java.awt.;
import java.awt.event.;
import javax.swing.;
import javax.swing.filechooser.;
import java.io.File;

public class FileChooser extends JPanel implements ActionListener {
   static private final String newline = "\n";
   JButton openButton, saveButton;
   JTextArea log;
   JFileChooser fc;

   public FileChooser() {
      super(new BorderLayout());

      log = new JTextArea(5, 20);
      log.setMargin(new Insets(5, 5, 5, 5));
      log.setEditable(false);
      JScrollPane logScrollPane = new JScrollPane(log);

      fc = new JFileChooser();

      openButton = new JButton("Open a File...");
      openButton.addActionListener(this);

      saveButton = new JButton("Save a File...");
      saveButton.addActionListener(this);

      JPanel buttonPanel = new JPanel();
      buttonPanel.add(openButton);
      buttonPanel.add(saveButton);

      add(buttonPanel, BorderLayout.PAGE_START);
```

```
        add(logScrollPane, BorderLayout.CENTER);
    }

    public void actionPerformed(ActionEvent e) {
        if (e.getSource() == openButton) {
            int returnVal = fc.showOpenDialog(FileChooser.this);
            if (returnVal == JFileChooser.APPROVE_OPTION) {
                File file = fc.getSelectedFile();
                log.append("Opening: " + file.getName() + "." + newline);
            } else {
                log.append("Open command cancelled by user." + newline);
            }
            log.setCaretPosition(log.getDocument().getLength());
        } else if (e.getSource() == saveButton) {
            int returnVal = fc.showSaveDialog(FileChooser.this);
            if (returnVal == JFileChooser.APPROVE_OPTION) {
                File file = fc.getSelectedFile();
                log.append("Saving: " + file.getName() + "." + newline);
            } else {
                log.append("Save command cancelled by user." + newline);
            }
            log.setCaretPosition(log.getDocument().getLength());
        }
    }

    private static void createAndShowGUI() {
        JFrame frame = new JFrame("FileChooser");
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.add(new FileChooser());
        frame.pack();
        frame.setVisible(true);
    }

    public static void main(String[] args) {
        SwingUtilities.invokeLater(new Runnable() {
            public void run() {
                UIManager.put("swing.boldMetal", Boolean.FALSE);
```

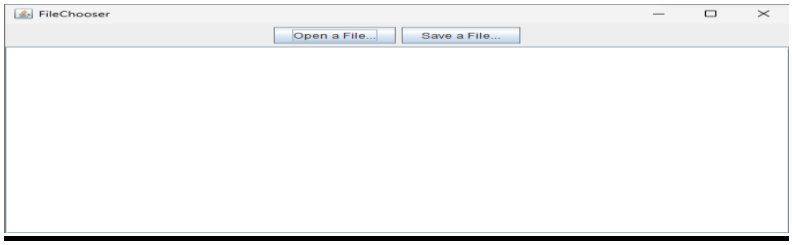
```
            createAndShowGUI();
        }
    });
  }
}
```

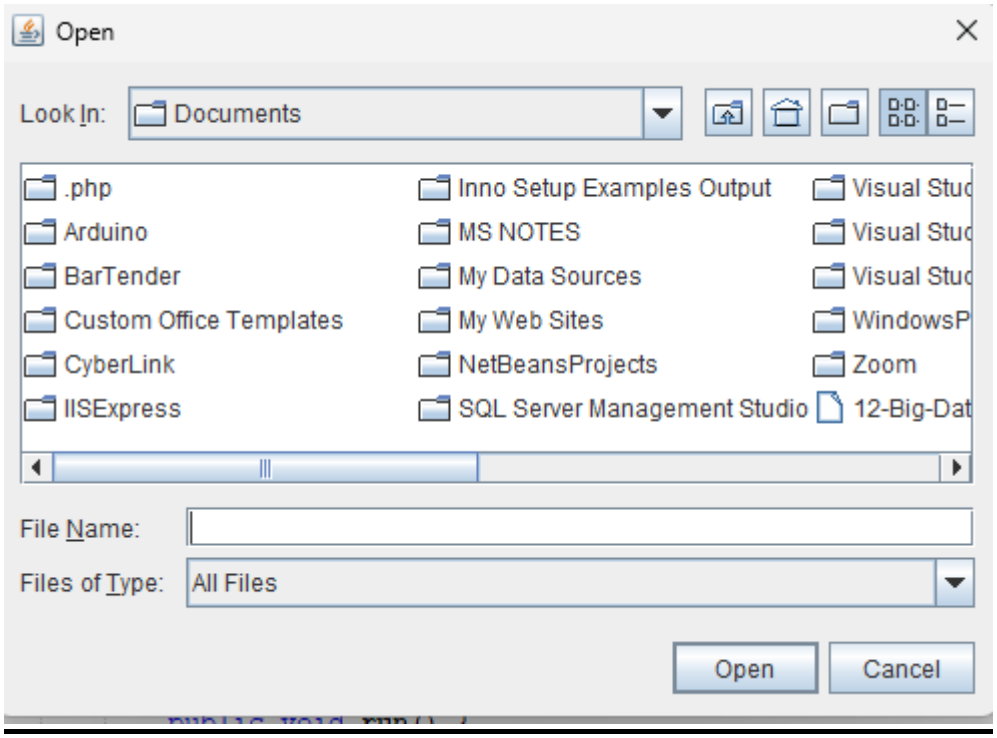
Steps to Run the Program in NetBeans IDE:

1. Open NetBeans IDE.
2. Create a new Java project:
   Go to "File" > "New Project..."
   Choose "Java" under "Categories" and "Java Application" under "Projects." Click "Next >."
   Enter a project name (e.g., "FileChooserApp") and choose a location for your project. Click "Finish."
3. In the newly created project, create a new Java class:
   Rightclick on the "Source Packages" folder in the "Projects" window.
   Select "New" > "Java Class..."
   Enter the class name (e.g., "FileChooser") and click "Finish."
4. Replace the contents of the `FileChooser` class with the provided code.
5. Save the changes (Ctrl + S or Cmd + S on macOS).
6. Run the Java program:
   Rightclick on the "FileChooser.java" file in the "Projects" window.
   Select "Run File" from the context menu.

7. The program's user interface will open, allowing you to interact with it. Click the "Open a File..." button to select a file or the "Save a File..." button to save a file. The actions you take will be displayed in the program's text area.

OUTPUT:





Conclusion:

This Java program demonstrates the use of the `java.nio.file` package to handle pattern matching and file selection through a graphical user interface (GUI). Pattern matching, a common technique in shell scripting, is essential for locating files based on specified patterns. The program showcases how to create a basic file chooser applicatio

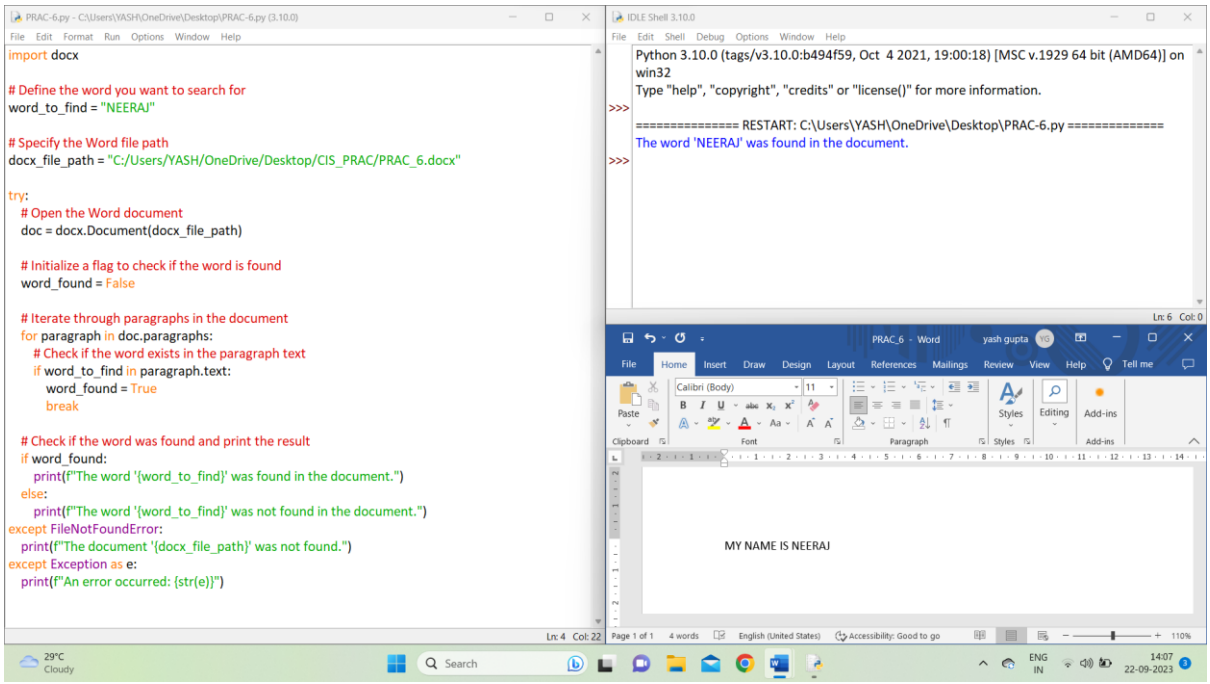
n using Swing components, allowing users to open and save files interactively. By integrating Java's `java.nio.file` package and Swing, developers can build versatile file manipulation tools with user-friendly interfaces. Pattern matching and file selection are essential functionalities in various applications, making this program a practical example of their implementation in Java

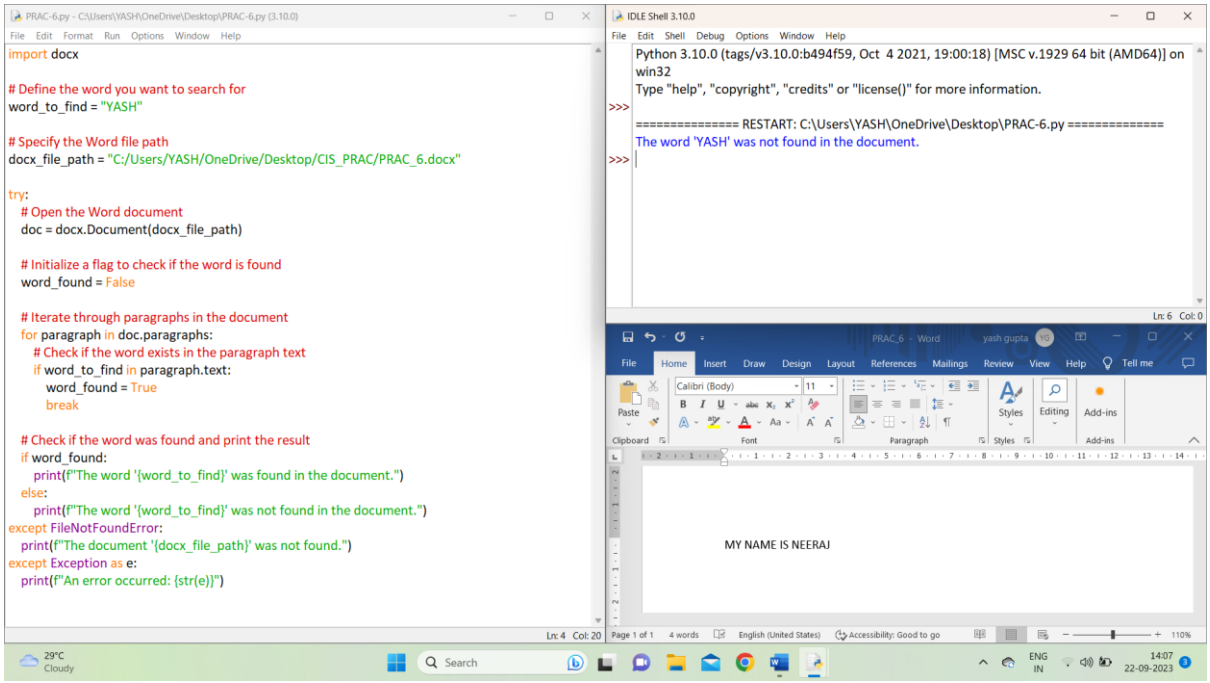
6.Aim:-Write a program to find a word in a file.

Code:-

```
import docx

# Define the word you want to search for
word_to_find = "NEERAJ"

# Specify the Word file path
docx_file_path =
"C:/Users/YASH/OneDrive/Desktop/CIS_PRAC/PRAC_6.docx"

try:
    # Open the Word document
    doc = docx.Document(docx_file_path)

    # Initialize a flag to check if the word is found
    word_found = False

    # Iterate through paragraphs in the document
    for paragraph in doc.paragraphs:
        # Check if the word exists in the paragraph text
        if word_to_find in paragraph.text:
            word_found = True
            break

    # Check if the word was found and print the result
    if word_found:
        print(f"The word '{word_to_find}' was found in the document.")
    else:
        print(f"The word '{word_to_find}' was not found in the document.")
except FileNotFoundError:
    print(f"The document '{docx_file_path}' was not found.")
except Exception as e:
    print(f"An error occurred: {str(e)}")
```

Output:-


PRAC-6.py - C:\Users\YASH\OneDrive\Desktop\PRAC-6.py (3.10.0)
File Edit Format Run Options Window Help
import docx
# Define the word you want to search for
word_to_find = "NEERAJ"
# Specify the Word file path
docx_file_path = "C:/Users/YASH/OneDrive/Desktop/CIS_PRAC/PRAC_6.docx"
try:
# Open the Word document
doc = docx.Document(docx_file_path)
# Initialize a flag to check if the word is found
word_found = False
# Iterate through paragraphs in the document
for paragraph in doc.paragraphs:
# Check if the word exists in the paragraph text
if word_to_find in paragraph.text:
word_found = True
break
# Check if the word was found and print the result
if word_found:
print(f"The word '{word_to_find}' was found in the document.")
else:
print(f"The word '{word_to_find}' was not found in the document.")
except FileNotFoundError:
print(f"The document '{docx_file_path}' was not found.")
except Exception as e:
print(f"An error occurred: {str(e)}")
Ln: 4 Col: 22
IDLE Shell 3.10.0
File Edit Shell Debug Options Window Help
Python 3.10.0 (tags/v3.10.0:b494f59, Oct 4 2021, 19:00:18) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
========== RESTART: C:\Users\YASH\OneDrive\Desktop\PRAC-6.py ==========
The word 'NEERAJ' was found in the document.
>>>
Ln: 6 Col: 0
PRAC_6 - Word
yash gupta
MY NAME IS NEERAJ
Page 1 of 1 4 words English (United States) Accessibility: Good to go
29°C Cloudy
Search
ENG IN
14:07 22-09-2023

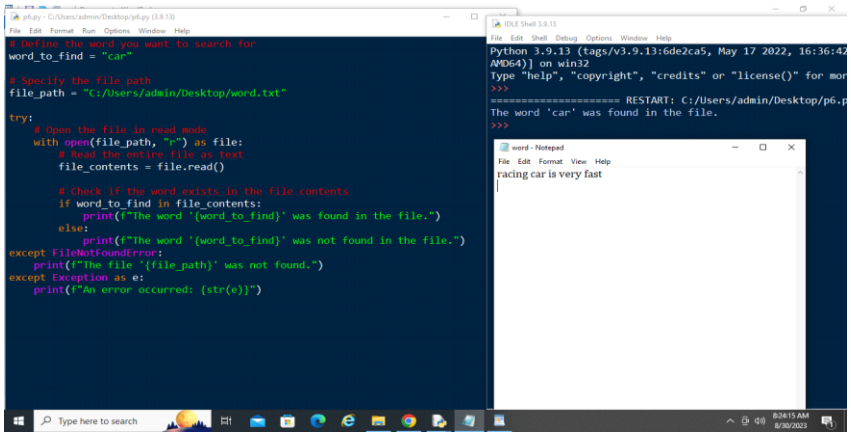

PRAC-6.py - C:\Users\YASH\OneDrive\Desktop\PRAC-6.py (3.10.0)
File Edit Format Run Options Window Help
import docx
# Define the word you want to search for
word_to_find = "YASH"
# Specify the Word file path
docx_file_path = "C:/Users/YASH/OneDrive/Desktop/CIS_PRAC/PRAC_6.docx"
try:
# Open the Word document
doc = docx.Document(docx_file_path)
# Initialize a flag to check if the word is found
word_found = False
# Iterate through paragraphs in the document
for paragraph in doc.paragraphs:
# Check if the word exists in the paragraph text
if word_to_find in paragraph.text:
word_found = True
break
# Check if the word was found and print the result
if word_found:
print(f"The word '{word_to_find}' was found in the document.")
else:
print(f"The word '{word_to_find}' was not found in the document.")
except FileNotFoundError:
print(f"The document '{docx_file_path}' was not found.")
except Exception as e:
print(f"An error occurred: {str(e)}")
Ln: 4 Col: 20
IDLE Shell 3.10.0
File Edit Shell Debug Options Window Help
Python 3.10.0 (tags/v3.10.0:b494f59, Oct 4 2021, 19:00:18) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
========== RESTART: C:\Users\YASH\OneDrive\Desktop\PRAC-6.py ==========
The word 'YASH' was not found in the document.
>>>
Ln: 6 Col: 0
PRAC_6 - Word
yash gupta
MY NAME IS NEERAJ
Page 1 of 1 4 words English (United States) Accessibility: Good to go
29°C Cloudy
Search
ENG IN
14:07 22-09-2023

(Text File)

Code:

```
# Define the word you want to search for
word_to_find = "car"
# Specify the file path
file_path = "C:/Users/admin/Desktop/word.txt"
try:
    # Open the file in read mode
    with open(file_path, "r") as file:
        # Read the entire file as text
        file_contents = file.read()
        # Check if the word exists in the file contents
        if word_to_find in file_contents:
            print(f"The word '{word_to_find}' was found in the file.")
        else:
            print(f"The word '{word_to_find}' was not found in the file.")
except FileNotFoundError:
    print(f"The file '{file_path}' was not found.")
except Exception as e:
    print(f"An error occurred: {str(e)}")
```

Output:



Aim: Create forensic images of digital devices from volatile data such as memory using Imager for: (i) Computer System, (ii) Server, (iii)Mobile Device

Software requirement:

(i) For Computer Systems:
- Imager Software
- Operating System Compatibility

(ii) For Servers:
- Imager Software
- Network Forensic Tools
- Server-Specific Forensic Tools

(iii) For Mobile Devices:
- Imager Software for Mobile
- Mobile Device Drivers
- Mobile Forensic Suite
- OS Compatibility

- General Software Requirements for All Devices:
    - Chain of Custody Software
    - Documentation and Reporting Tools
    - Forensic Analysis Software

Theory:

Forensic analysis of volatile data from digital devices, such as memory, is a critical aspect of digital forensics. The choice of tools and techniques may vary depending on the type of device being analysed. Here's a brief overview of using Imager for each category:

(i) Computer System:

1.Memory Acquisition: Imager can be used to create a forensic image of a computer system's volatile memory (RAM). This involves capturing the current state of the RAM, which can contain valuable information like running processes, open files, and network connections.
2.Analysis: Once the memory image is acquired, it can be analyzed using various tools and techniques. Volatile data may include cached passwords, encryption keys, and evidence of recent activities.

(ii) Server:

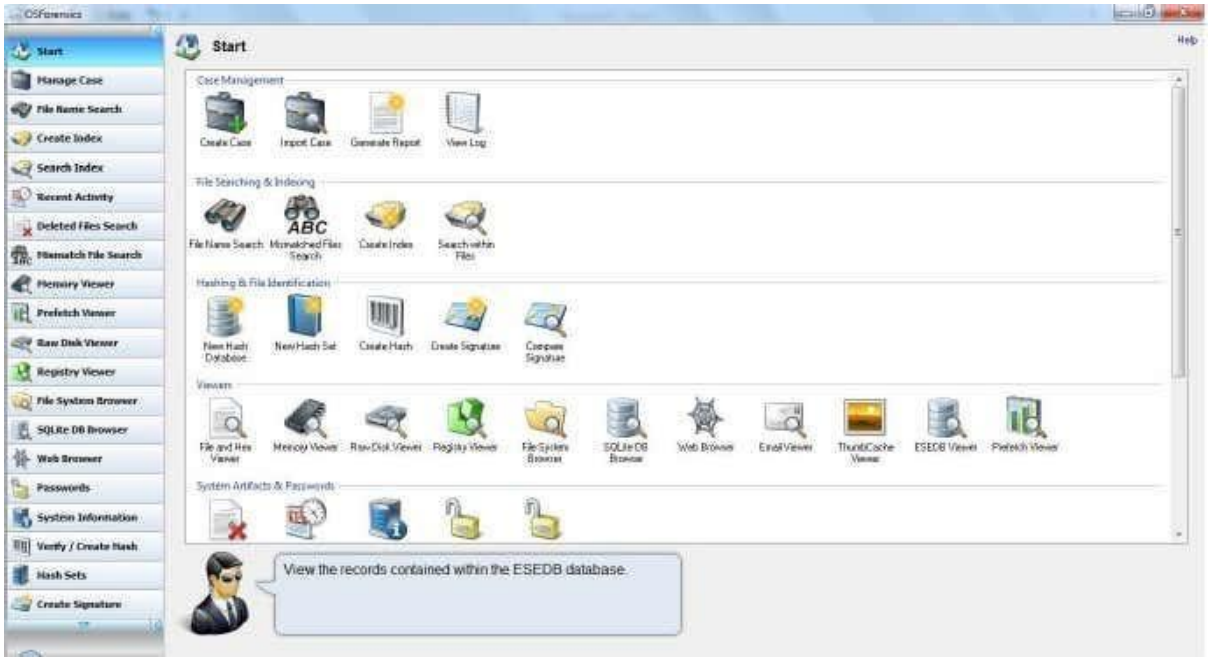
1.Remote Acquisition: Servers often run critical services, so remote memory acquisition is preferred to avoid service interruption. Imager can be used in conjunction with network forensic tools to remotely capture memory.
2.Server-Specific Analysis: Server memory may contain logs, database information, and records of user interactions. Imager can create images that are then analyzed with server-specific forensic tools.

(iii) Mobile Device:

1.Memory Dump: Imager can be used to create a memory dump of a mobile device's RAM. This is typically done through a physical or logical acquisition process.
2.Mobile OS Analysis: The analysis of mobile device volatile memory may involve extracting information about running applications, recent communications, location data, and more. Specialized mobile forensic tools may be used in conjunction with Imager.

Regardless of the device type, it's important to follow established forensic procedures, maintain chain of custody, and ensure data integrity throughout the process. Additionally, it's crucial to stay updated with the latest forensic techniques and tools, as digital technology evolves rapidly, and new challenges constantly emerge in the field of digital forensics.

Step 1: Start OSForenscis as Start->All Programs -> OSForensics ->OSForensics

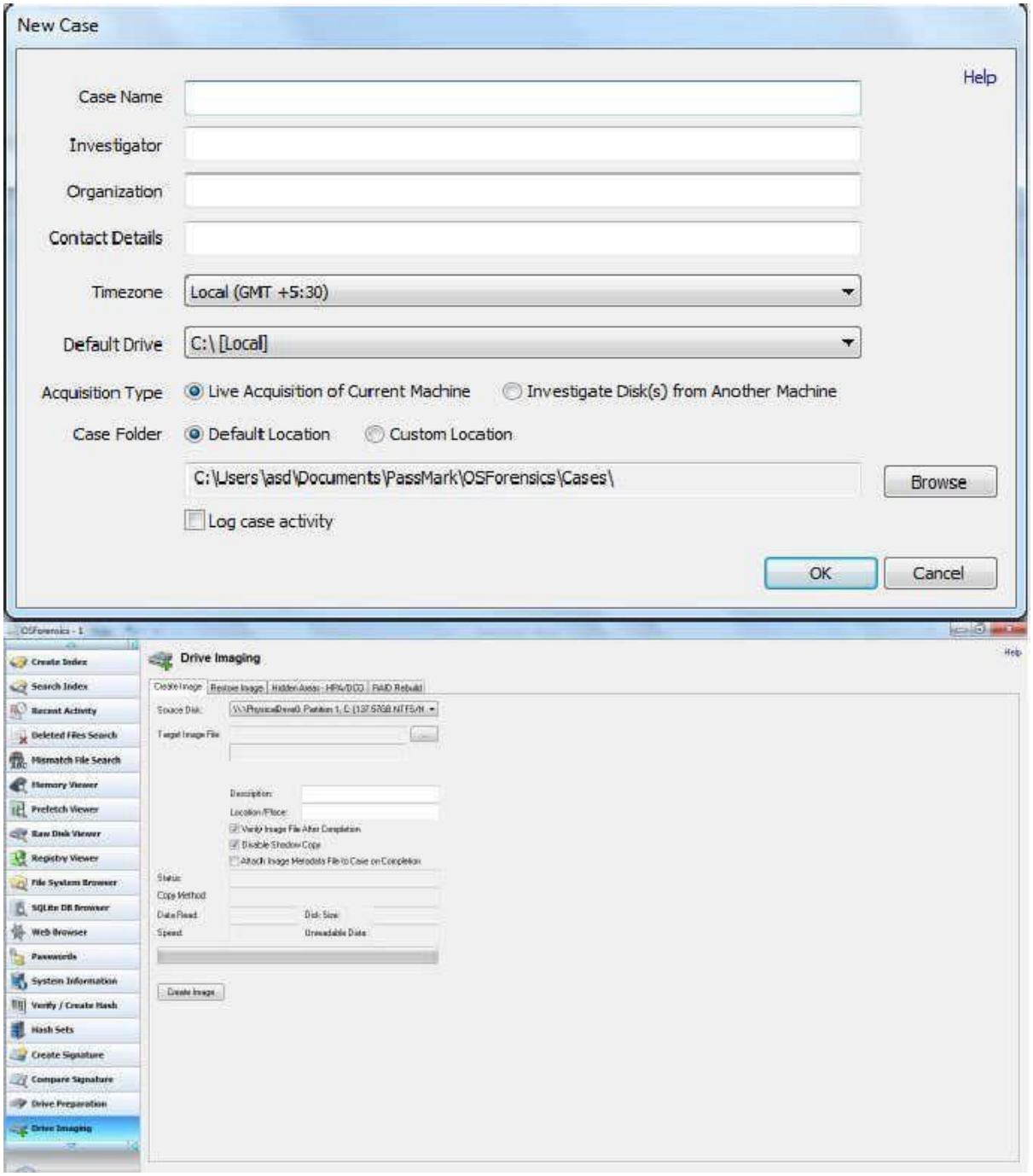


Step 2: Create a new Case by selecting Create Case. The following window will be popped:

Enter Appropriate Details and Select OK

Step 3 : Select Drive Imaging under Housekeeping

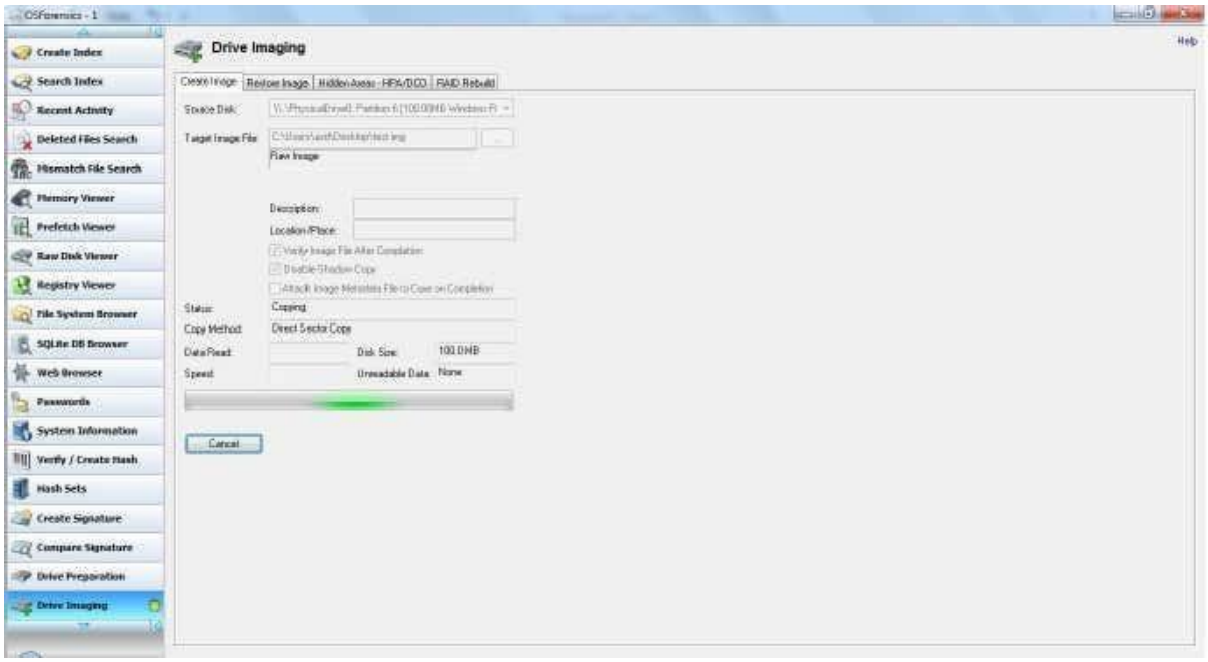
Step 4 : Select the appropriate disk for which the image has to be procured under Source Disk. Select the Traget Image Location and Click Create Image



Step 5: The Drive Image (.img File) will be created at the target location.

The same process should be repeated for Mobile as follows:
Attach the mobile device and select the appropriate mobile drive as the Source Disk.

The same process should be repeated for Server as follows: Attach the mobile device and select the appropriate mobile drive as the Source Disk.



Conclusion:

In conclusion, the forensic imaging of volatile data from digital devices, including computer systems, servers, and mobile devices, using tools like Imager, is a critical process in digital forensics. In all cases, digital forensics professionals must adhere to established forensic procedures, ensuring the preservation of evidence, maintaining the chain of custody, and complying with legal requirements. Staying up-to-date with the latest forensic techniques and tools is essential due to the ever-evolving nature of digital technology and the challenges it presents in the field of digital forensics.

7.Aim: Access and extract relevant information from Windows Registry for investigation process using Registry View, perform data analysis and bookmark the findings with respect to: (i) Computer System; (ii) Computer Network; (iii) Mobile Device;

What Is the Registry?

The registry is a database of stored configuration information about the users, hardware, and software on a Windows system. Although the registry was designed to configure the system, to do so, it tracks such a plethora of information about the user's activities, the devices connected to system, what software was used and when, etc. All of this can be useful for the forensic investigator in tracking the who, what, where, and when of a forensic investigation. The key is just knowing where to look.

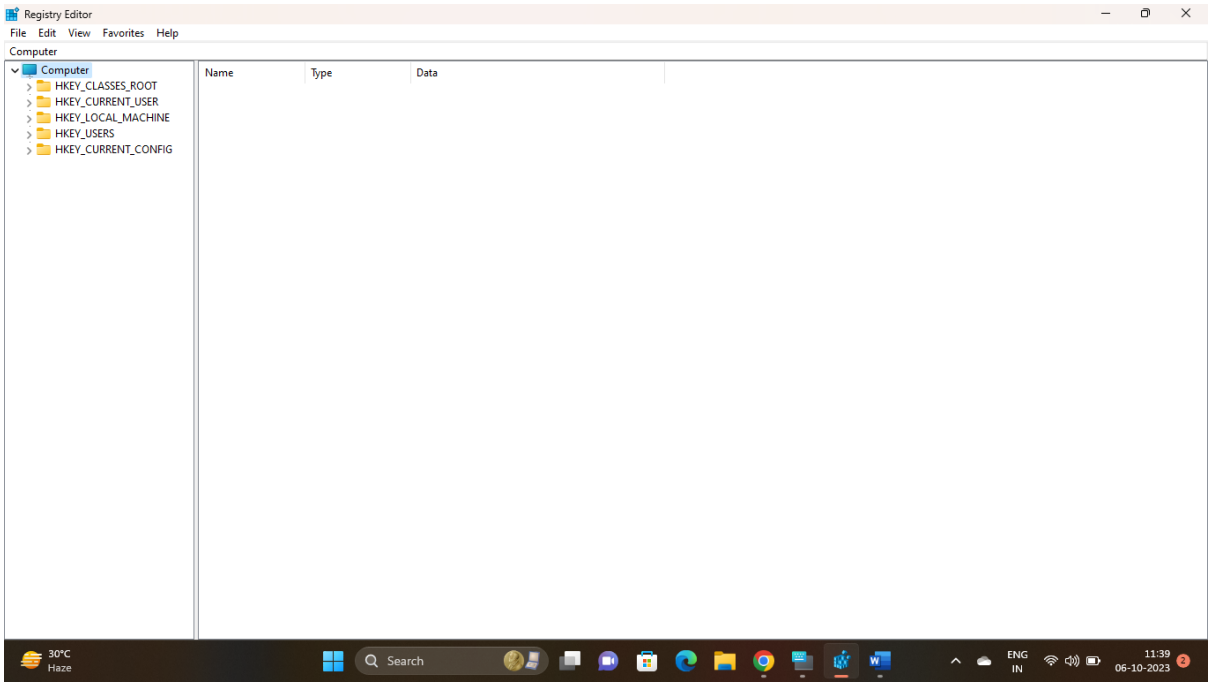
Hives

Inside the registry, there are root folders. These root folders are referred to as hives. There are five (5) registry hives.

- **HKEY_USERS**: contains all the loaded user profiles
- **HKEYCURRENT_USER**: profile of the currently logged-on user
- **HKEYCLASSES_ROOT**: configuration information on the application used to open files
- **HKEYCURRENT_CONFIG**: hardware profile of the system at startup
- **HKEYLOCAL_MACHINE**: configuration information including hardware and software settings

Accessing the Registry

On our own system—not in a forensic mode—we can access the registry by using the regedit utility built into Windows. Simply type regedit in the search window and then click on it to open the registry editor like that below



- **Wireless Evidence in the Registry**

Many hackers crack a local wireless access point and use it for their intrusions. In this way, if the IP address is traced, it will lead back to the neighbor's or other wireless AP and not them.

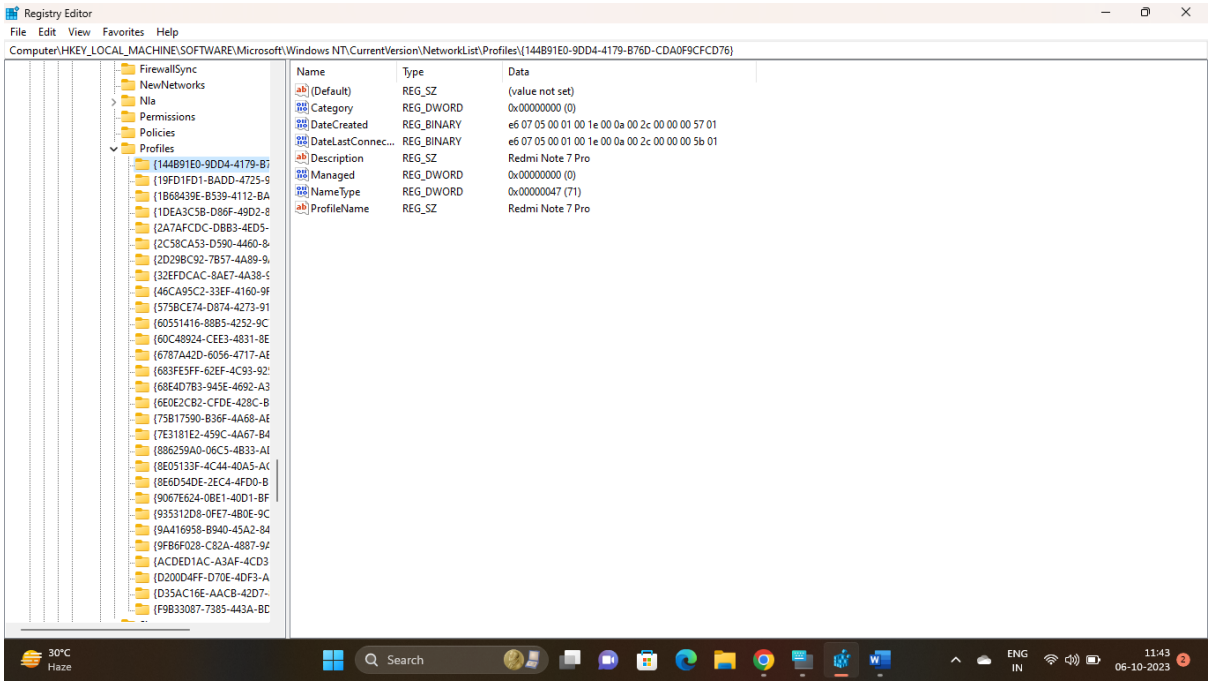
For example, back in January 2012, an Anonymous member, John Borrell III, hacked into the computer systems of the Salt Lake City police department and the Utah Chiefs of Police. The FBI was called in to investigate and they traced the hacker back to the IP address of Blessed Sacrament Church's Wi-Fi AP in Toledo, Ohio. The hacker had apparently cracked the password of the church's wireless AP and was using it to hack "anonymously" on the Internet.

Eventually, the FBI was able to find the suspect through various investigation techniques, mostly low-tech, exhaustive, detective work. It helped that John Borrell had bragged on Twitter of his success as a hacker. Eventually, Mr. Borrell was convicted and sentenced to two years in Federal prison.

When the FBI tracked down Mr. Borrell and seized his computer, they were able to prove he had been connected to the church AP by examining his registry. The forensic investigator simply had to look in the registry at this location:

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\NetworkList\Profiles

There, you will find a list of GUIDs of wireless access points the machine has been connected to. When you click on one, it reveals information including the SSID name and the date last connected in hexadecimal. So, although Mr. Borrell initially denied his involvement with this hack, this evidence was conclusive and he eventually plead guilty.



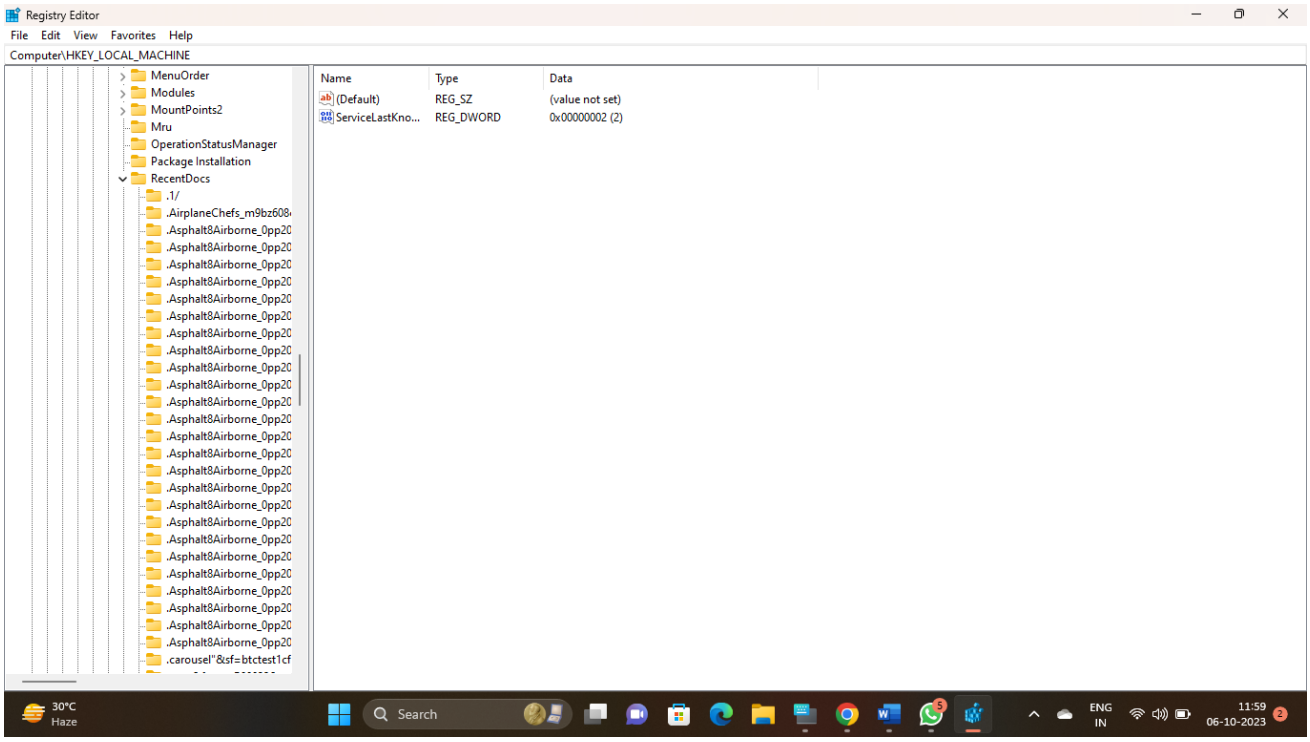
- **The RecentDocs Key**

The Windows registry tracks so much information about the user's activities. In most cases, these registry keys are designed to make Windows run more efficiently and smoothly. As a forensic investigator, these keys are like a road map of the activities of the user or attacker.

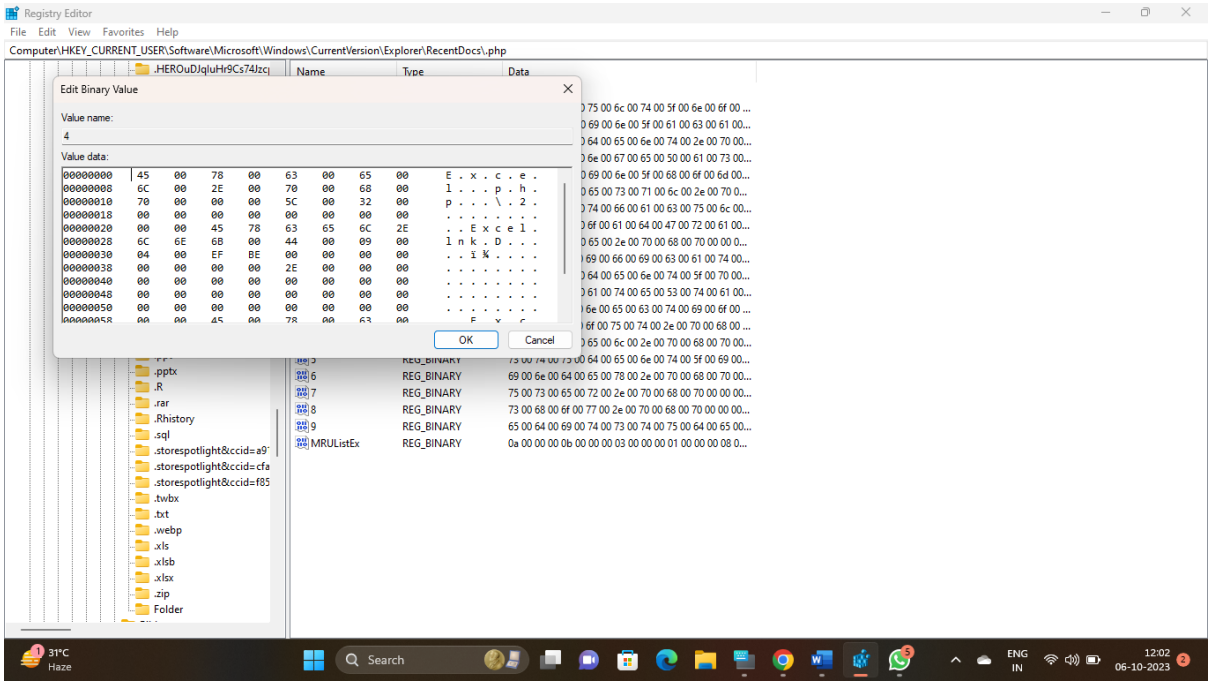
One of those keys is the "RecentDocs" key. It tracks the most recent documents used or opened on the system by file extension. It can be found at:

HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs

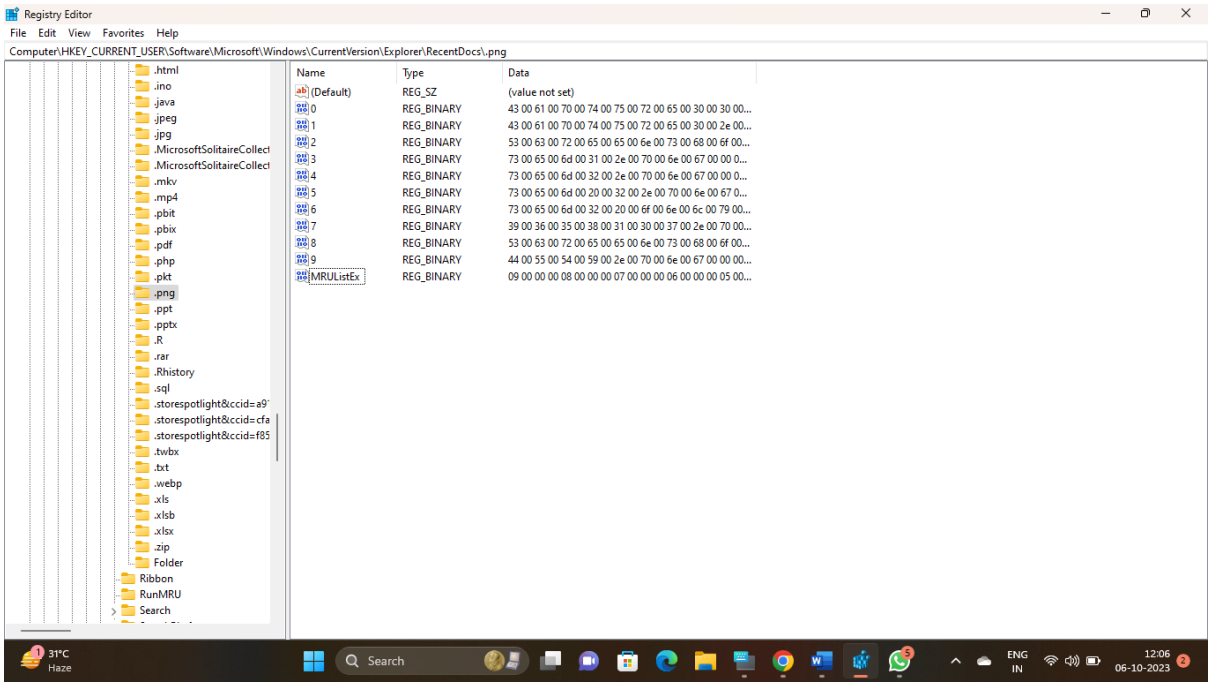
So, for instance, the most recently used Word documents would be found under .doc or the .docx extension depending upon the version of Word they were created in (each key can hold up to the last 10 documents). If we go to the .docx extension, we see the last 10 Word documents listed under this key.



When we click on one of those keys, it reveals information about the document as seen below. We can view the document data in both hex, to the left, and ASCII, to the right. In this case, it show that this document was a Metasploit course outline.



In civil or policy violation investigations, evidence might be found in the various graphic file extensions such as .jpg, .gif, or .png.

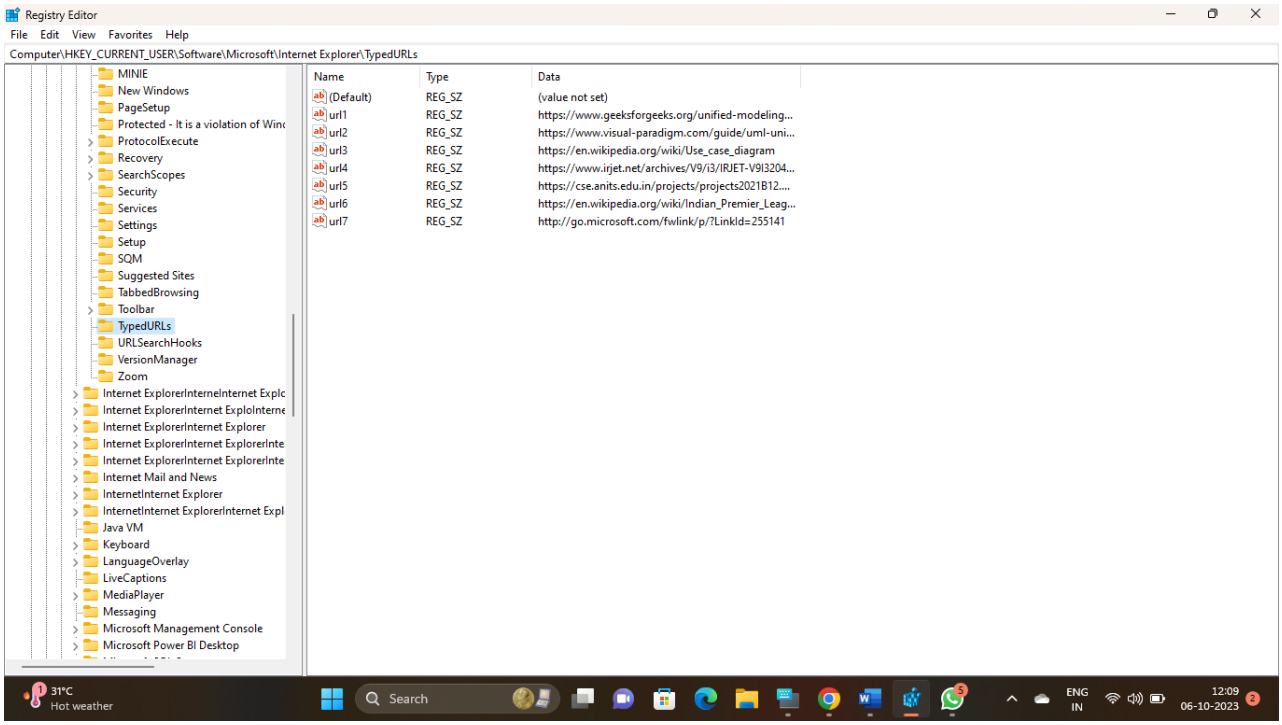


- **TypedURLs Key**

When the user types a URL in Internet Explorer, this value is stored in the registry at:

HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\TypedURLs

When we open that key in the registry, it lists the last URLs that the user visited with IE. This could reveal the source of malicious malware that was used in the breach, or in civil or policy violation types of investigations, may reveal what the user was looking for/at.



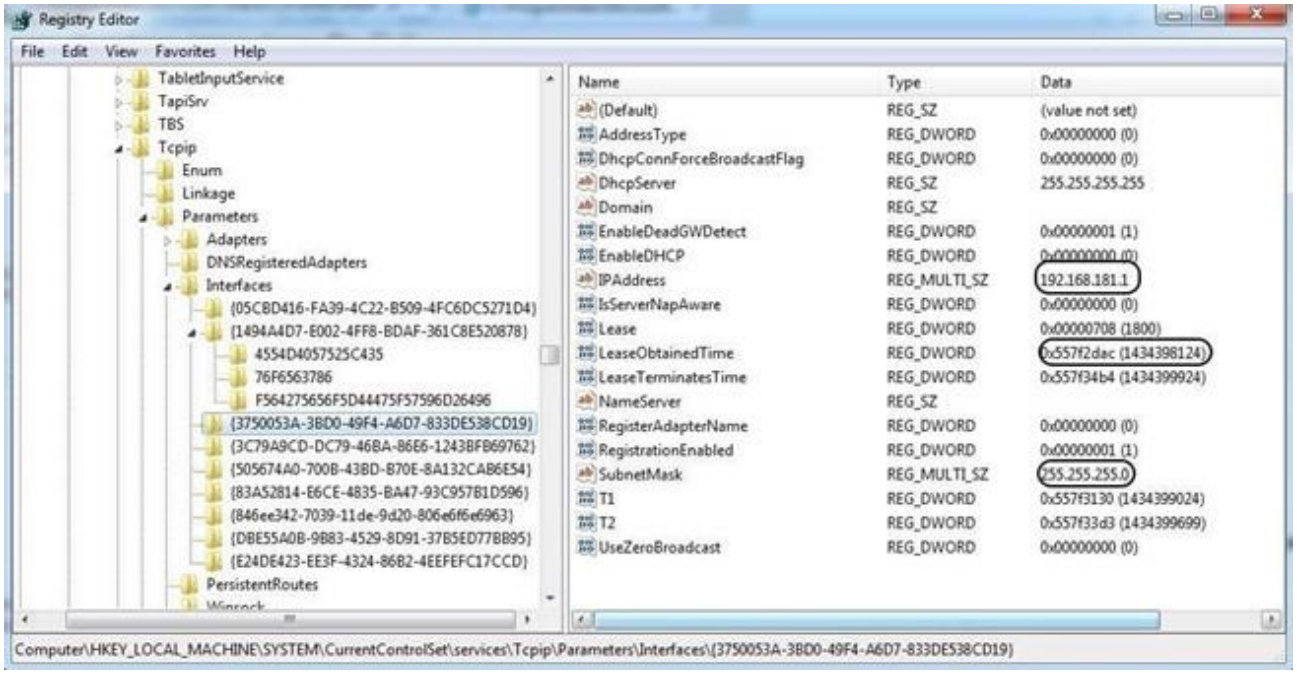
The values will run from urI1 (the most recent) to urI7 (the oldest).

➢ IP Addresses

The registry also tracks the IP addresses of the user interfaces. Note that there may be numerous interfaces and this registry key tracks each interface's IP address and related information.

HKEY_LOCAL_MACHINE\System\Services\CurrentControlSet\services\Tcpip\Parameters\Interfaces

As we can see below, we can find the IP address assigned to the interface, the subnet mask, and the time when the DHCP server leased the IP. In this way, we can tell whether the suspect was using that particular IP at the time of the intrusion or crime.

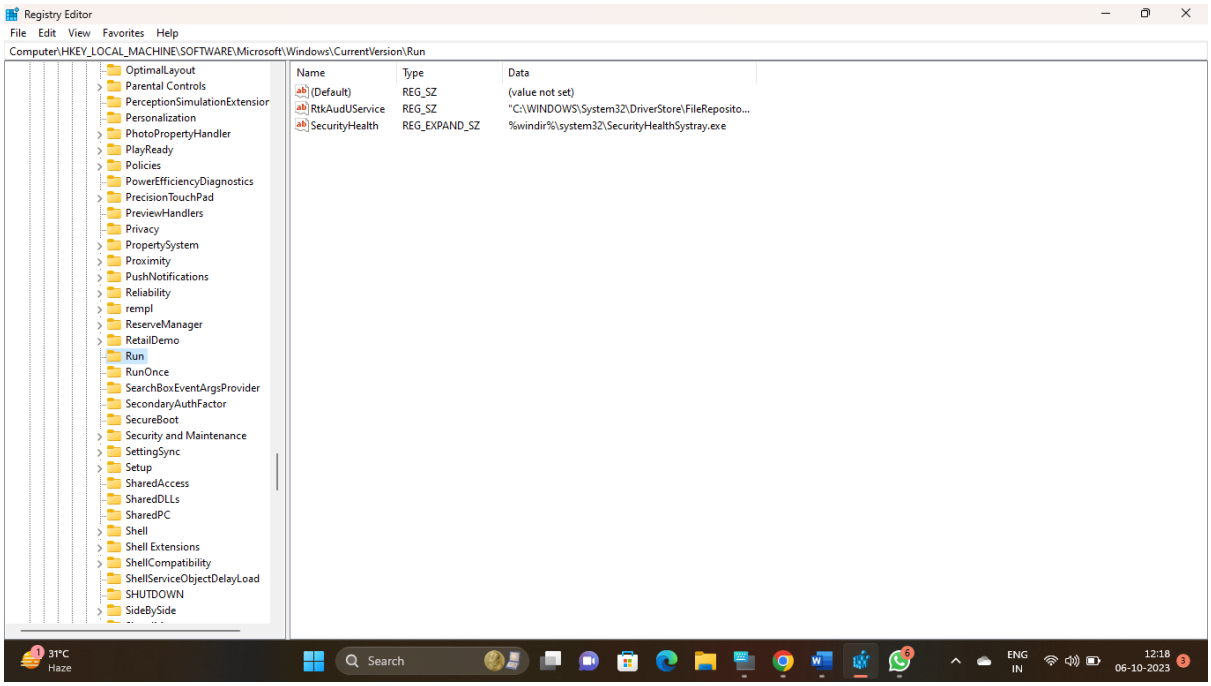


- **Start Up Locations in the Registry**

As a forensic investigator, we often need to find what applications or services were set to start when the system starts. Malware is often set to start each time the system restarts to keep the attacker connected. This information can be located in the registry in literally tens of locations. We will look at a just a few of the most commonly set keys.

Probably the most used location is:

HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run



Any software/locations designated in these subkeys will start every time the system starts. Rootkits and other malicious software can often be found here and they will start each time the system starts.

- **RunOnce Startup**

If the hacker just wanted the software to run once at start up, the subkey may be set here.

HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunOnce

- **Start Up Services**

The key below lists all the services that set to start at system startup. If the key is set to 2, the service starts automatically; if it is set to 3, the service must be started manually; and if the key is set to 4, the service is disabled.

HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services

- **Start Legacy Applications**

When legacy 16-bit applications are run, the program listed is run at:
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\WOW

- **Start When a Particular User Logs On**

In the following key, the values are run when the specific user logs in.
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run

Storage Artifacts in the Registry

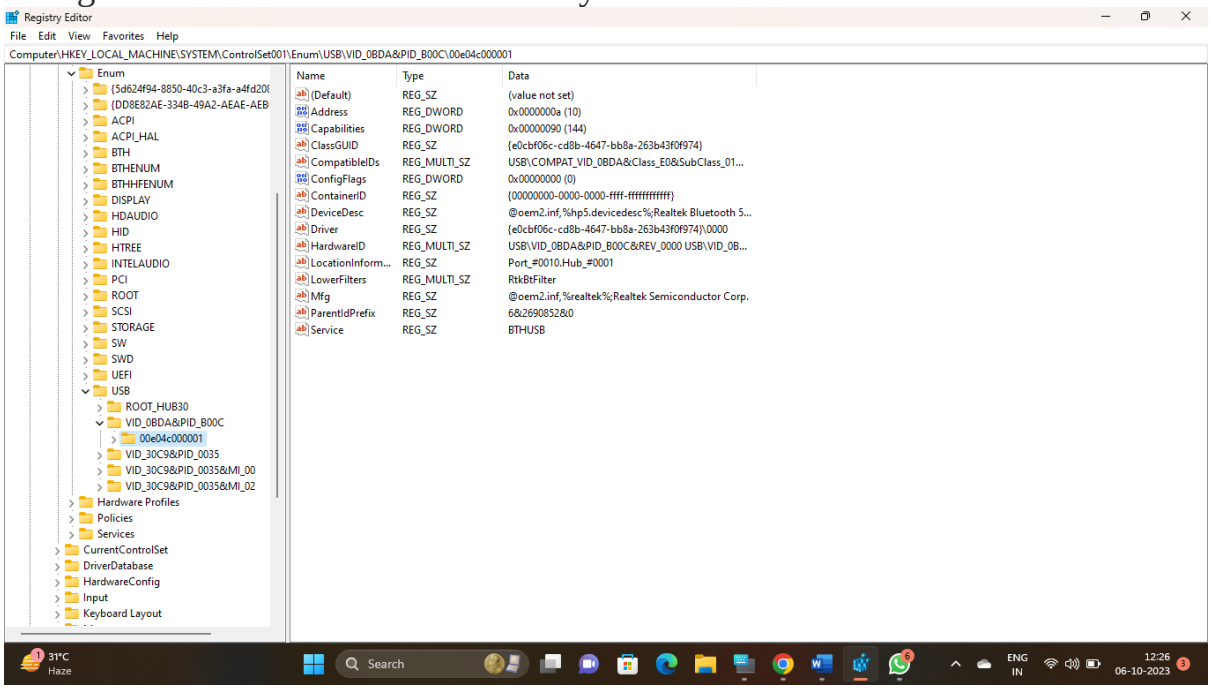
Often, the suspect will use a Flash drive or hard drive for their malicious activities and then remove them so as not to leave any evidence. The skilled forensic investigator, though, can still find traces of evidence of those storage devices within the registry, if they know where to look.
The registry on a Windows system varies a bit from version to version. A skilled, professional digital forensic investigator needs to be able to work with nearly all versions of Windows and other operating systems. Since Windows 7 is still the most widely used operating system, by far, I will be demonstrating on it. Keep in mind, though, that this will vary slightly between versions.

- USB Storage Devices

Imagine a case where we suspect that someone installed a keylogger or removed confidential information with a USB drive. How would we find evidence that a

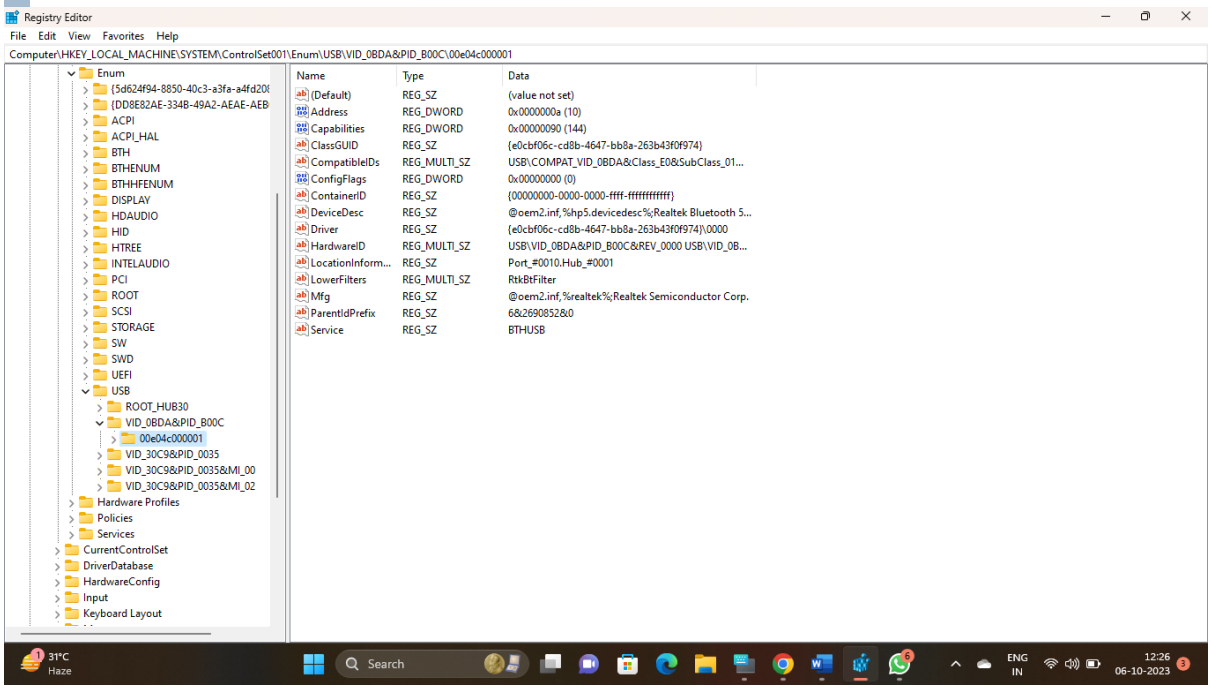
USB storage device was inserted and used? To find evidence of USB storage devices, we want to look at the following key.
HK_Local_Machine\System\ControlSet00x\Enum\USBSTOR

In this key, we will find evidence of any USB storage device that has ever been connected to this system. Expand USBSTOR to see a listing of every USB storage device ever connected to this system.



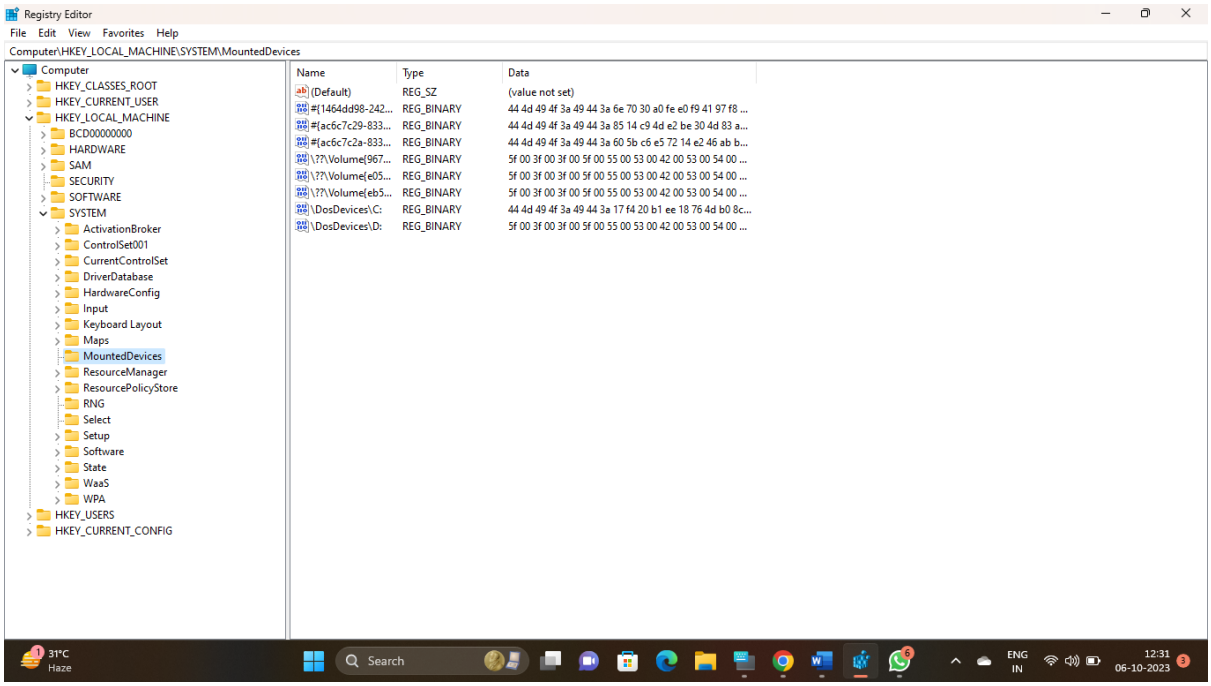
When we expand it, it reveals a unique identifier for that device. By clicking on this identifier, we can find much more information about the device

e.

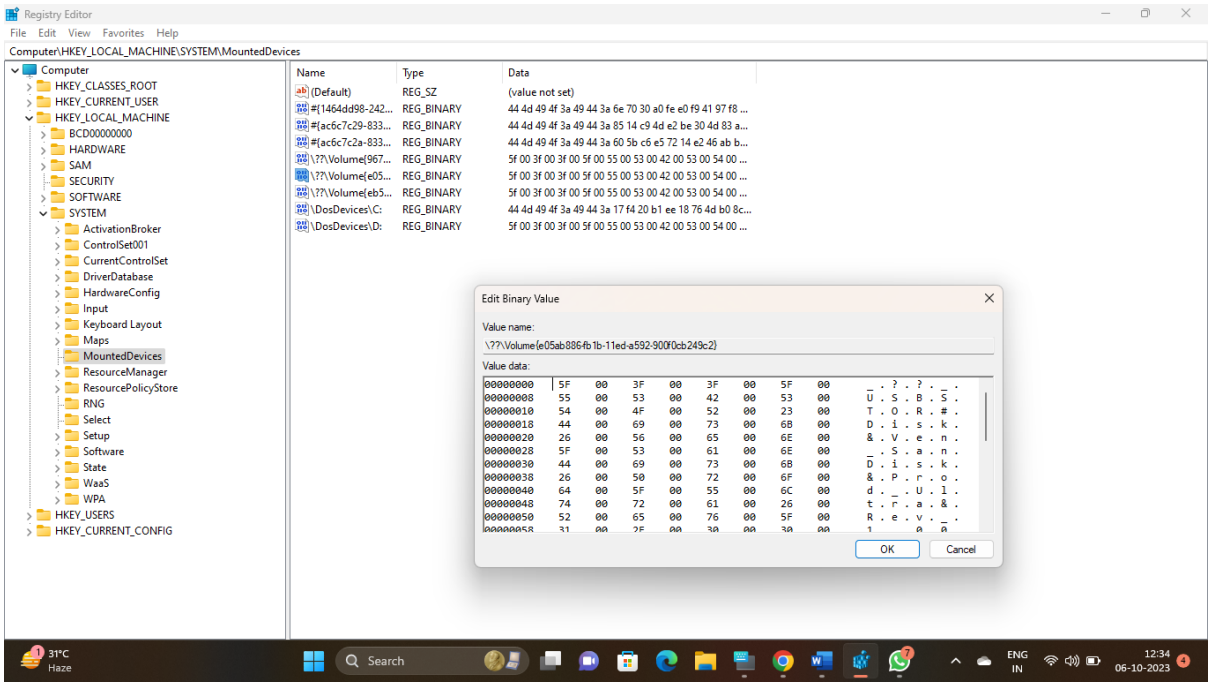


➢ Mounted Devices

If the suspect used any hardware device that must be mounted to either read or write data (CD-ROM, DVD, hard drive, flash drive, etc.), the registry will record the mounted device. This information is stored at:
HKEY_LOCAL_MACHINE\System\MountedDevices
As you can see below, when we click on this key, it provides us a list of every device ever mounted on that machine.



If we need further information on any of those mounted devices, we can simply click on it, and it will open a small app that will enable us to read the data in ASCII.

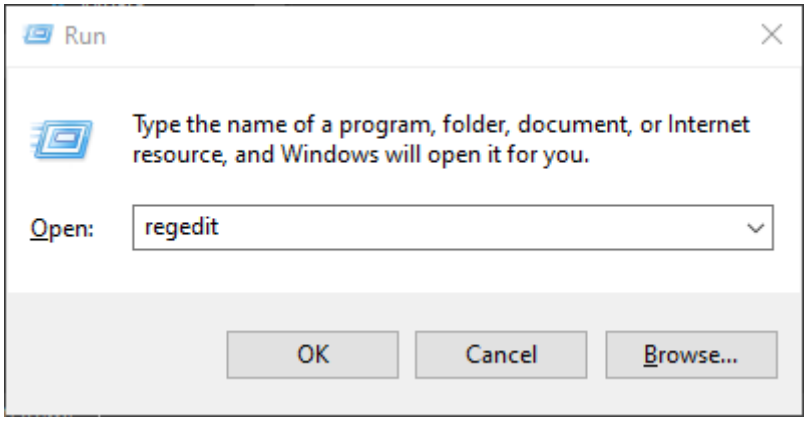


CONCLUSION: The registry is a depository of volumes of information on what happened on a Windows system, and by learning our way around it, we can reconstruct the elements of a crime that it was used for.

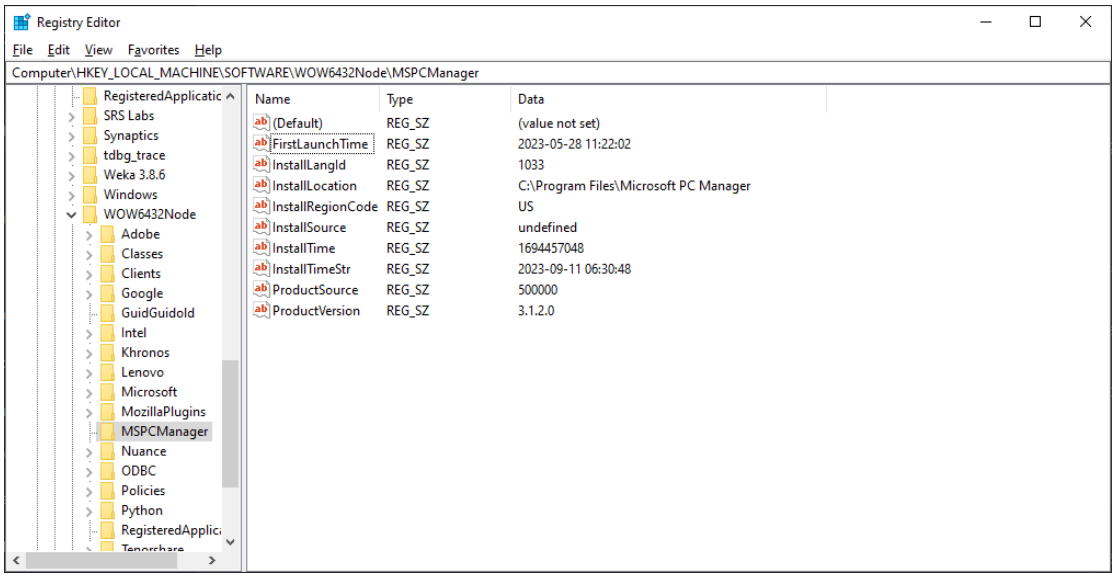
8.Aim: Generate a report based on the analysis done using Registry View for different case scenario of the following: (i) Computer System; (ii) Computer Network (iii) Mobile Device (iv) Wireless Network

1. Computer System Analysis:

Tool: Regedit
Demo: Use Regedit to access the Windows Registry. Analyze key registry hives like HKEY_LOCAL_MACHINE and
HKEY_CURRENT_USER
for suspicious entries. Look for unauthorized changes, malware signatures, or unusual startup items.

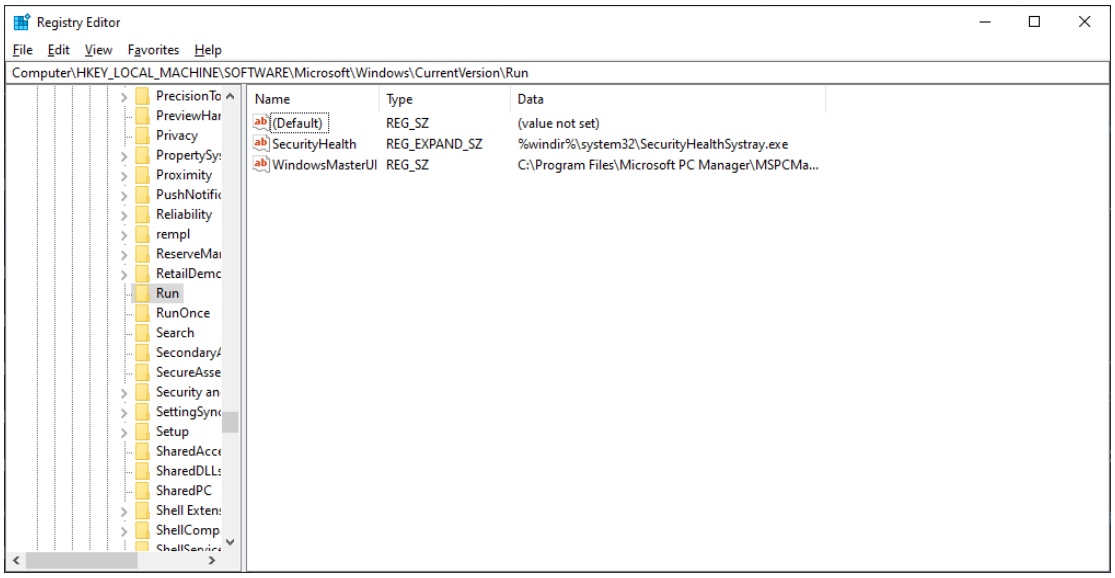


Path:
Computer\HKEY_LOCAL_MACHINE\SOFTWARE\WOW6432Node\MSPCManager



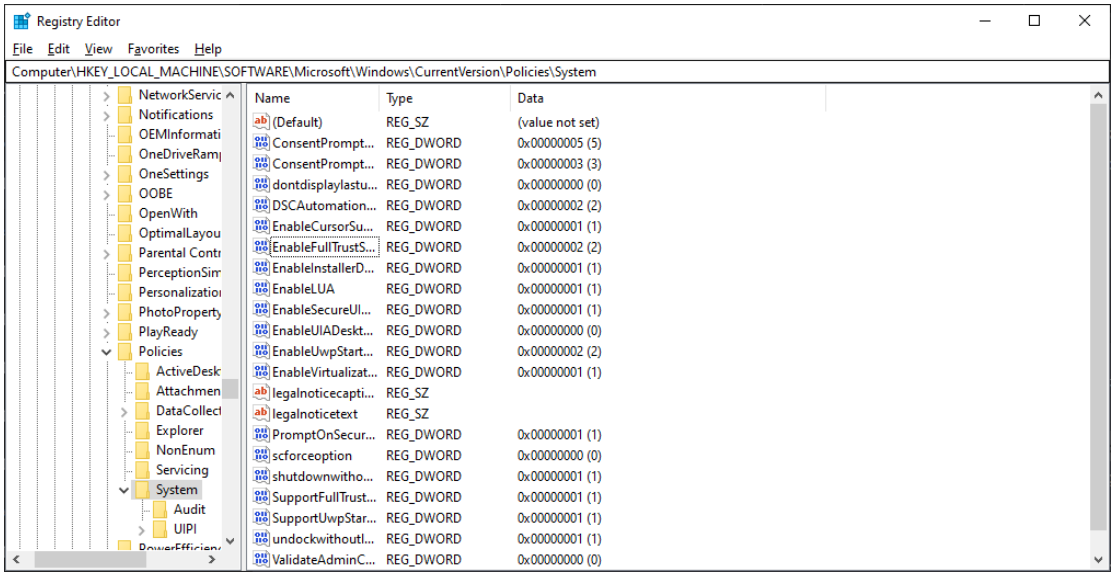
- Unusual Autostart Entries: Registry entries that enable unknown or unauthorized programs to start automatically with the system.

  Path:
  Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run



Changes to User Account Control (UAC): Unauthorized modifications to UAC settings that could allow applications to run with elevated privileges without user consent.

Path:
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System



Registry Value: EnableLUA

The EnableLUA value controls whether UAC is enabled or disabled. Here's how to interpret it:

EnableLUA = 1: UAC is enabled.

EnableLUA = 0: UAC is disabled.

When analyzing UAC settings as part of a cybersecurity assessment, you would want to check if this value has been modified or tampered with. Any unauthorized changes to this value could potentially indicate an attempt to bypass UAC or escalate privileges, which is a suspicious activity.

- Unfamiliar DLL Files: Presence of unfamiliar or suspicious dynamic link library (DLL) files registered in system registry keys.

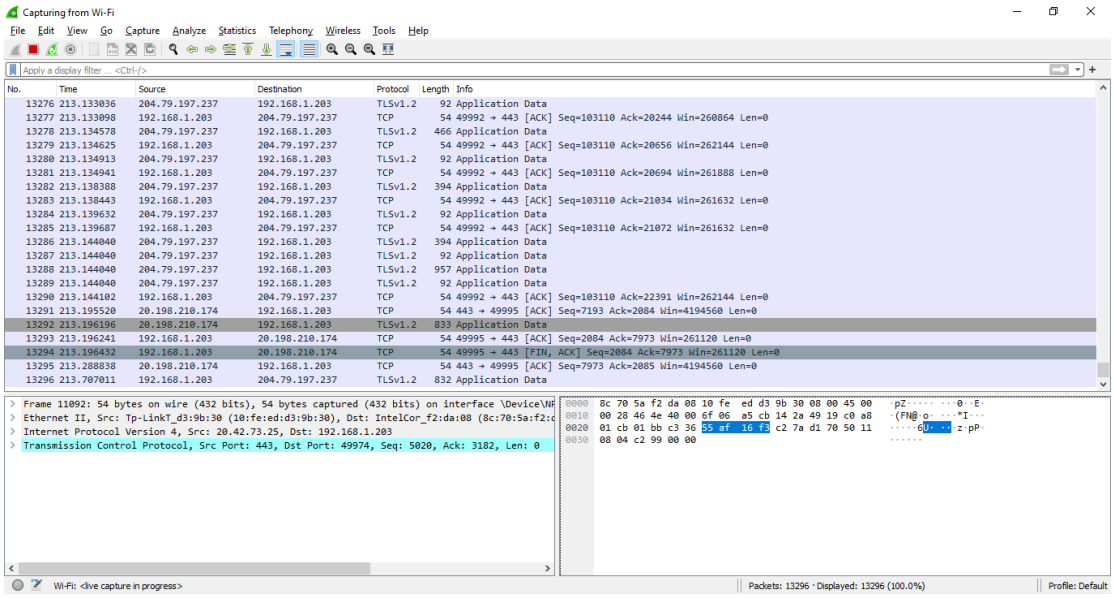
- New or Altered Services: The creation or modification of system services without a clear, legitimate reason.

- Unexplained Network Configuration Changes: Suspicious changes to network-related registry settings, such as firewall rules or proxy configurations, that might indicate an intrusion or malware activity.

2. **Computer Network Analysis:**

Tool: Wireshark

Demo: Capture network traffic with Wireshark. Analyze packets to identify unusual or suspicious network behavior. Look for anomalies, unauthorized connections, or signs of network intrusion.



Unusual Traffic Patterns: Irregular spikes in network traffic or unexpected surges in data transfer rates may indicate a possible DDoS attack or network scanning.

Unauthorized Port Scanning: Repeated attempts to access various ports on a network device can be a sign of a hacker attempting to find open services or vulnerabilities.

Unexplained Outbound Connections: Outbound connections to known malicious IP addresses or domains may suggest a compromised system or a malware infection.

- Large Amounts of Unencrypted Data: The transmission of sensitive information, such as login credentials or financial data, in unencrypted form can pose a significant security risk.

- Abnormal Protocol Usage: Instances where unusual or non-standard protocols are being used can signal malicious activity, as attackers often employ custom protocols to evade detection.

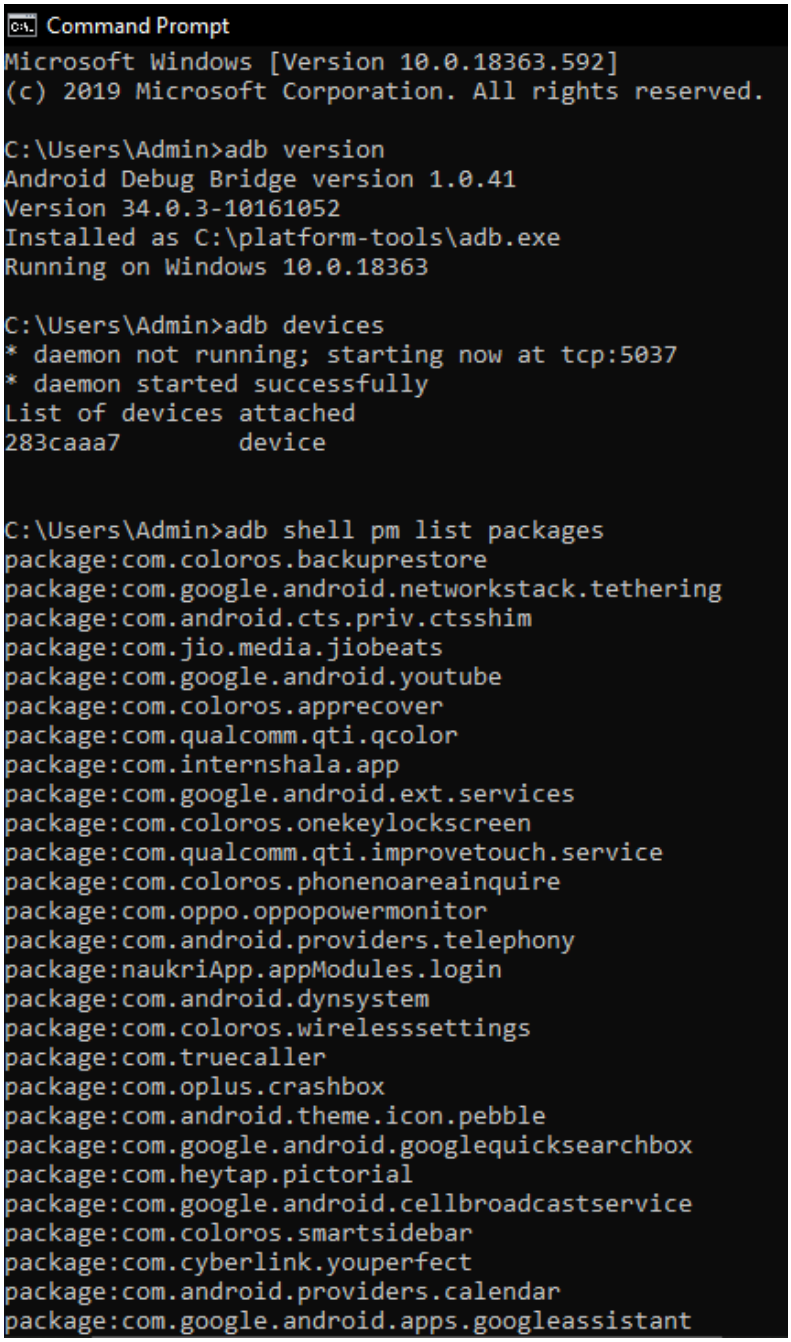
3. Mobile Device Analysis:

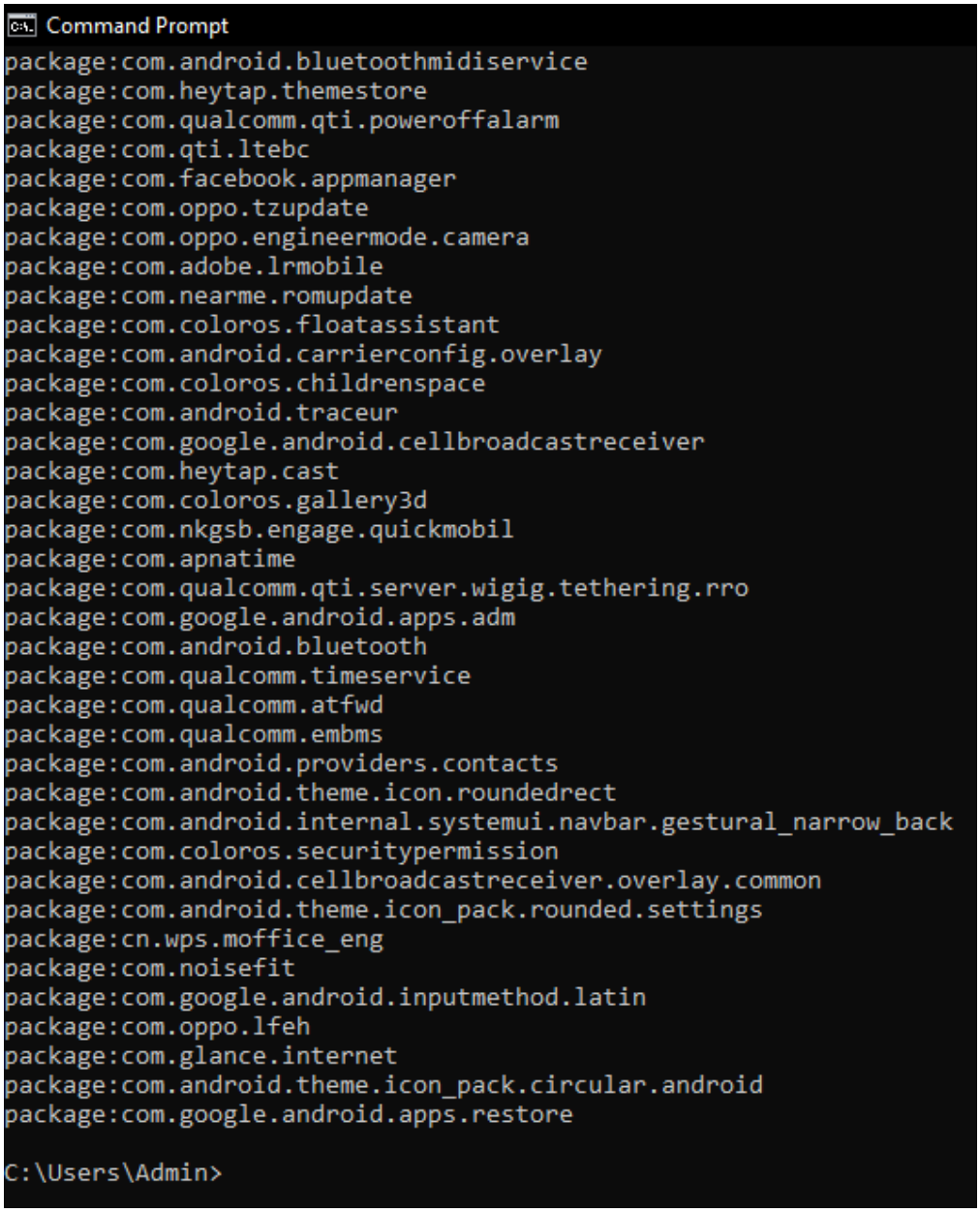
Tool: Android Debug Bridge (ADB)
Demo: Connect the mobile device to a computer and use ADB to access the Android device's registry-like data. Analyze app permissions, installed apps, and system settings for any irregularities or potential security issues.

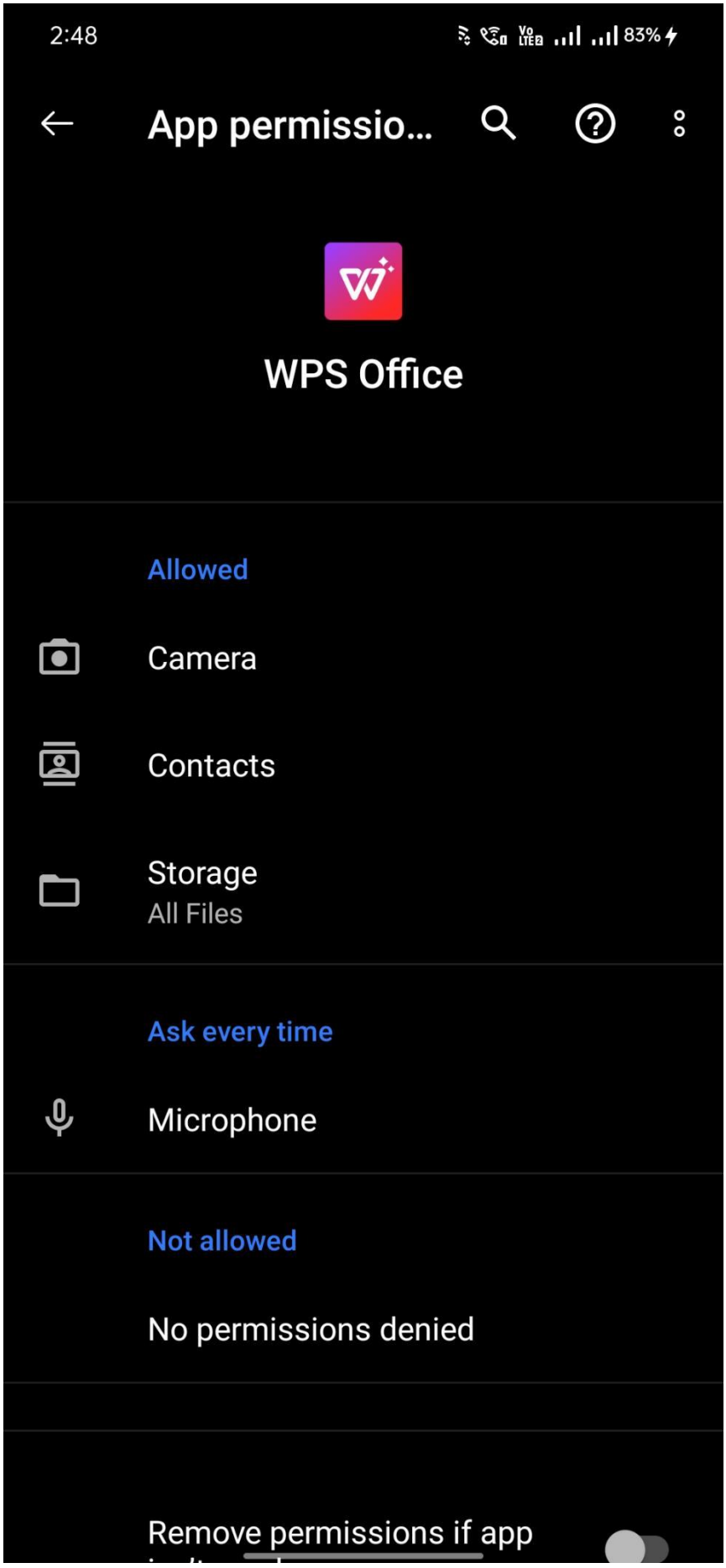
Look for

- Unfamiliar or potentially malicious app names and package IDs in the list of installed applications.
- Recent installation timestamps for apps that the user does not recall installing.
- Suspicious permissions granted to newly installed apps, such as access to sensitive data or device functions.

Use adb from platform tools and connect a physical device through usb and perform the following commands:


Command Prompt
Microsoft Windows [Version 10.0.18363.592]
(c) 2019 Microsoft Corporation. All rights reserved.
C:\Users\Admin>adb version
Android Debug Bridge version 1.0.41
Version 34.0.3-10161052
Installed as C:\platform-tools\adb.exe
Running on Windows 10.0.18363
C:\Users\Admin>adb devices
* daemon not running; starting now at tcp:5037
* daemon started successfully
List of devices attached
283caaa7 device
C:\Users\Admin>adb shell pm list packages
package:com.coloros.backuprestore
package:com.google.android.networkstack.tethering
package:com.android.cts.priv.ctsshim
package:com.jio.media.jiobeats
package:com.google.android.youtube
package:com.coloros.apprecover
package:com.qualcomm.qti.qcolor
package:com.internshala.app
package:com.google.android.ext.services
package:com.coloros.onekeylockscreen
package:com.qualcomm.qti.improvetouch.service
package:com.coloros.phonenoareainquire
package:com.oppo.oppopowermonitor
package:com.android.providers.telephony
package:naukriApp.appModules.login
package:com.android.dynsystem
package:com.coloros.wirelesssettings
package:com.truecaller
package:com.oplus.crashbox
package:com.android.theme.icon.pebble
package:com.google.android.googlequicksearchbox
package:com.heytap.pictorial
package:com.google.android.cellbroadcastservice
package:com.coloros.smartsidebar
package:com.cyberlink.youperfect
package:com.android.providers.calendar
package:com.google.android.apps.googleassistant


Command Prompt
package:com.android.bluetoothmidiservice
package:com.heytap.themestore
package:com.qualcomm.qti.poweroffalarm
package:com.qti.ltebc
package:com.facebook.appmanager
package:com.oppo.tzupdate
package:com.oppo.engineermode.camera
package:com.adobe.lrmobile
package:com.nearme.romupdate
package:com.coloros.floatassistant
package:com.android.carrierconfig.overlay
package:com.coloros.childrenspace
package:com.android.traceur
package:com.google.android.cellbroadcastreceiver
package:com.heytap.cast
package:com.coloros.gallery3d
package:com.nkgsb.engage.quickmobil
package:com.apnatime
package:com.qualcomm.qti.server.wigig.tethering.rro
package:com.google.android.apps.adm
package:com.android.bluetooth
package:com.qualcomm.timeservice
package:com.qualcomm.atfwd
package:com.qualcomm.embms
package:com.android.providers.contacts
package:com.android.theme.icon.roundedrect
package:com.android.internal.systemui.navbar.gestural_narrow_back
package:com.coloros.securitypermission
package:com.android.cellbroadcastreceiver.overlay.common
package:com.android.theme.icon_pack.rounded.settings
package:cn.wps.moffice_eng
package:com.noisefit
package:com.google.android.inputmethod.latin
package:com.oppo.lfeh
package:com.glance.internet
package:com.android.theme.icon_pack.circular.android
package:com.google.android.apps.restore
C:\Users\Admin>


2:48
83%
App permissio...
WPS Office
Allowed
Camera
Contacts
Storage
All Files
Ask every time
Microphone
Not allowed
No permissions denied
Remove permissions if app

10.Aim: Create a new investigation case using Forensic Tool: (i) Computer System; (ii) Computer Network; (iii) Mobile Device; (iv) Wireless Network.

Theory:

(i) Computer System:

The investigation will utilize specialized computer forensic tools such as EnCase or Autopsy to analyze the compromised computer systems, recover deleted files, examine access logs, and establish a timeline of events leading to the breach.

(ii) Computer Network:

Network forensic tools like Wireshark and Snort will be employed to capture and analyze network traffic, including packet captures and intrusion detection logs, to trace the path of the attacker and pinpoint vulnerabilities that were exploited.

(iii) Mobile Device:

Mobile device forensic tools like Cellebrite or Oxygen Forensic Suite will be used to extract and analyze data from the missing person's smartphone, including text messages, call records, photos, and geolocation data.

(iv) Wireless Network:

Wireless network forensic tools like Aircrack-ng or Kismet will be used to scan the Wi-Fi environment, locate unauthorized access points, and analyze their characteristics to determine if they are a security threat.

1. Computer System:

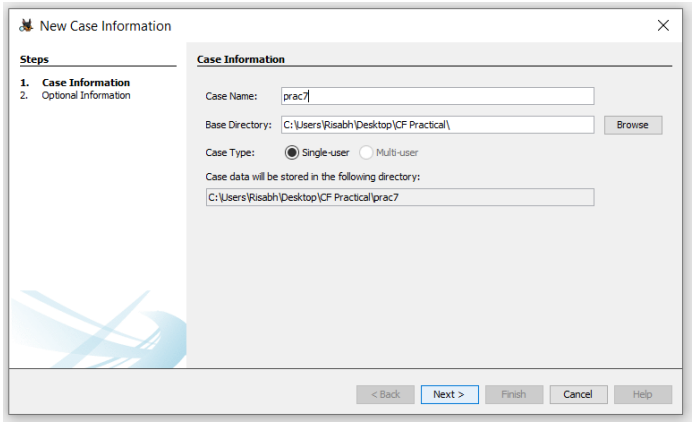
Software used: Autopsy 4.21.0

Steps:

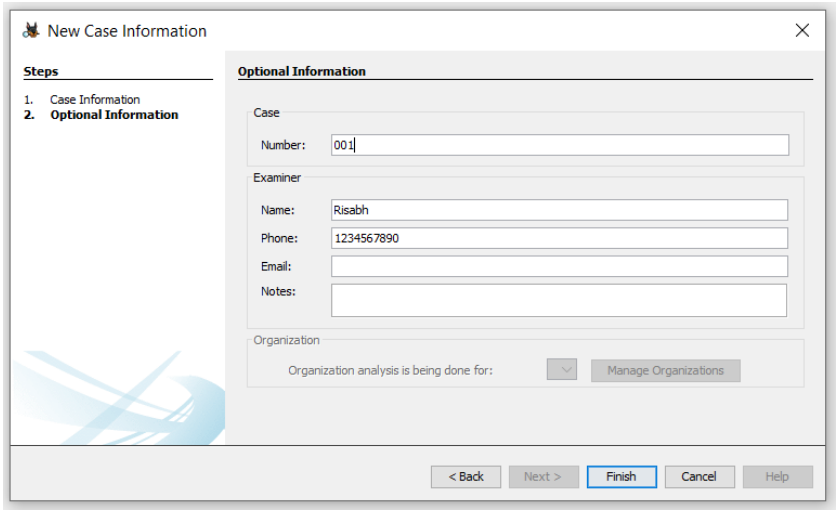
1. Download 8-jpeg-search.zip from the link and extract it. https://dftt.sourceforge.net/test8/index.html

2. Open Autopsy -> Click New Case

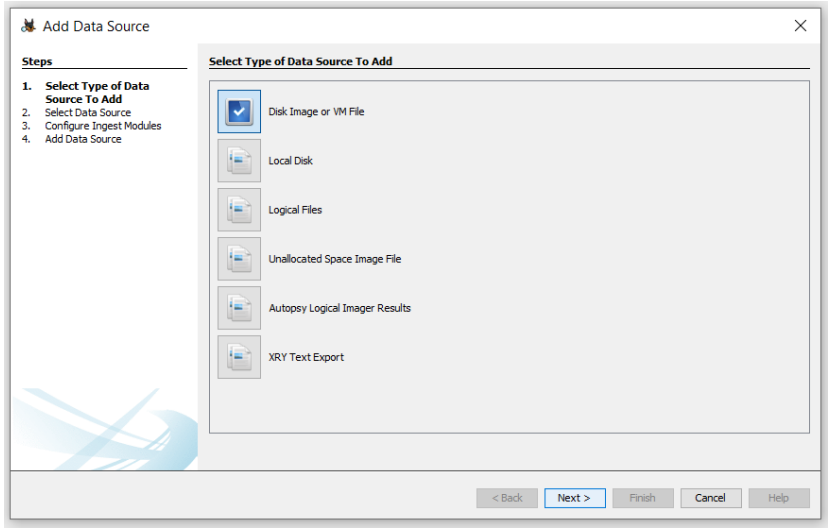
3. Give Case Name and Base Directory (where the case folder will be saved).



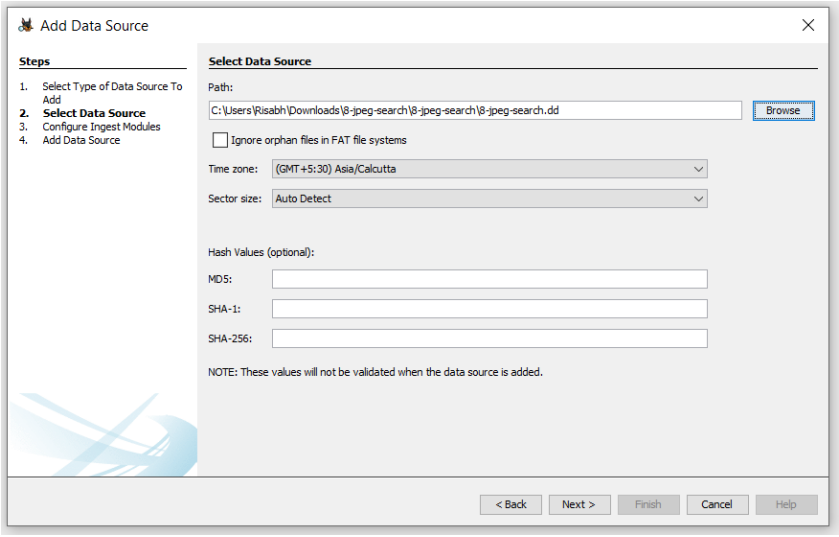
4. Give Optional Information -> Click Finish.



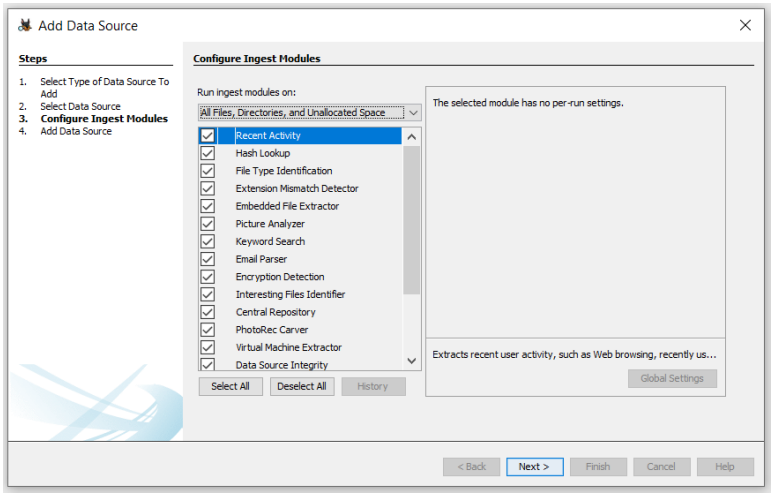
5. Select Disk Image or VM File



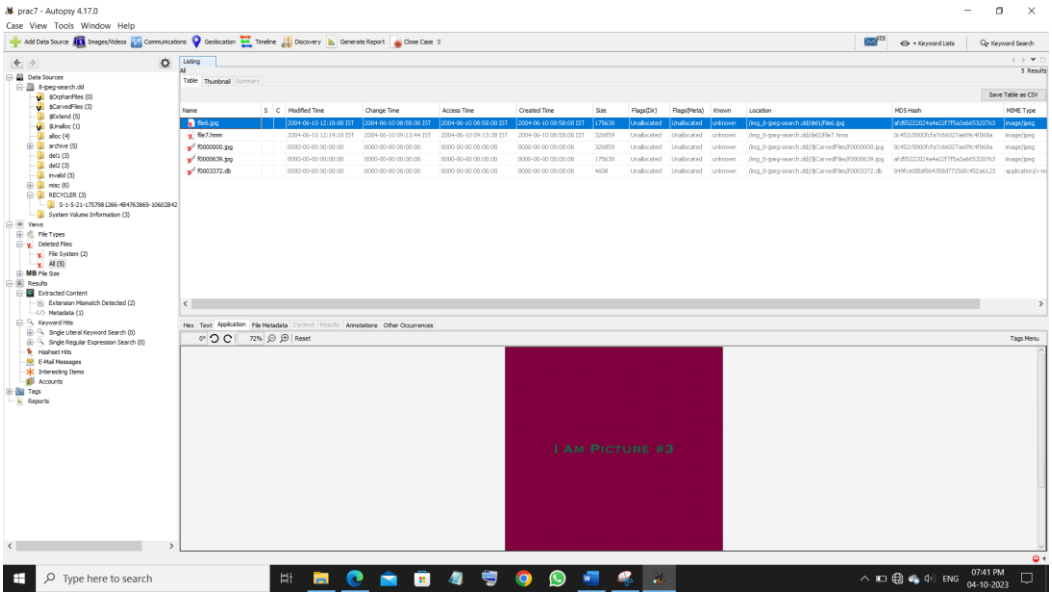
6. Give Path of the downloaded 8-jpeg-search.dd file.



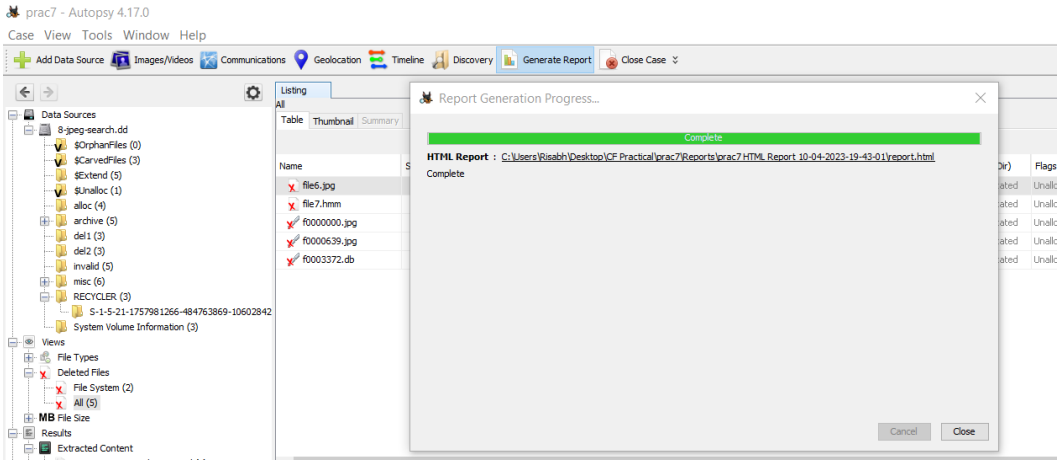
7. Click Next -> Click Finish



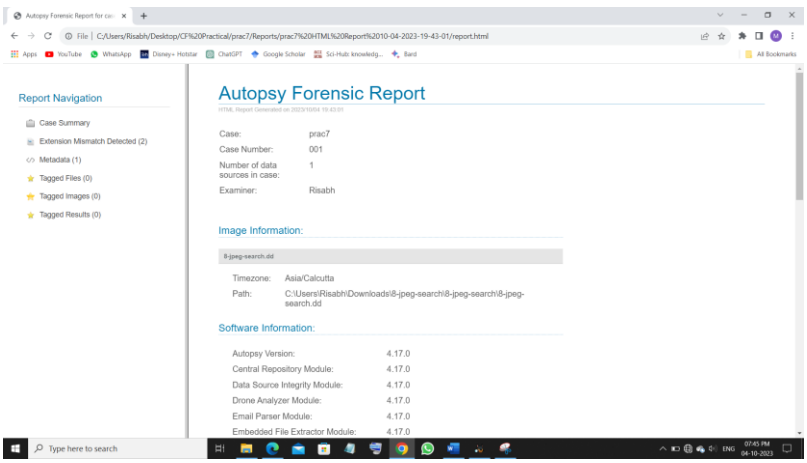
8. Click on Deleted Files and browse through files.



9. To generate case summary click Generate Report.



10. Browse through the directory to get HTML Report.



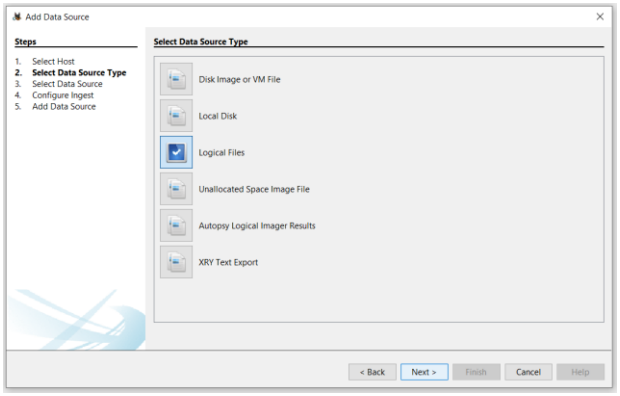
2. Mobile Device:

Software used: Autopsy 4.21.0

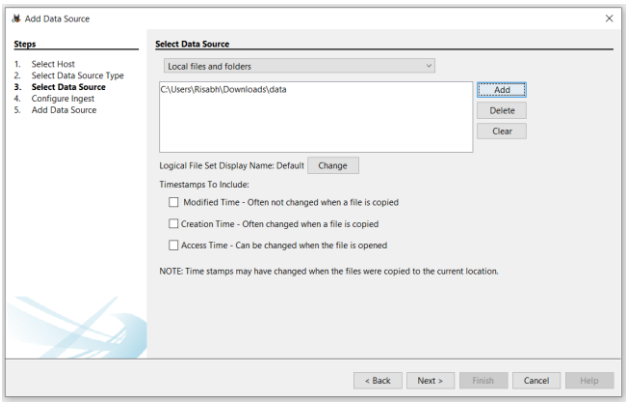
Steps:

1. Download android_image2.tar.gz from the link and extract it using 7-zip. https://samsclass.info/128/proj/M144.htm

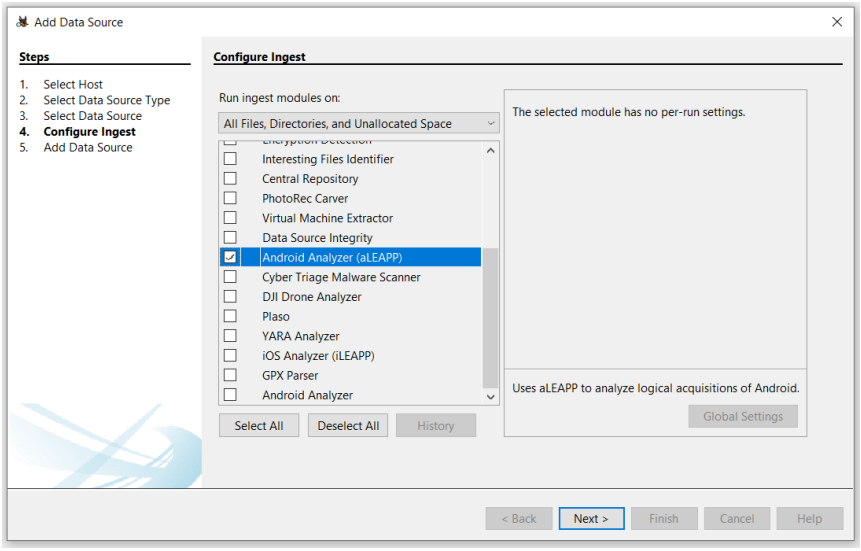
2. Follow steps as above and select Logical Files as Data Source Type.



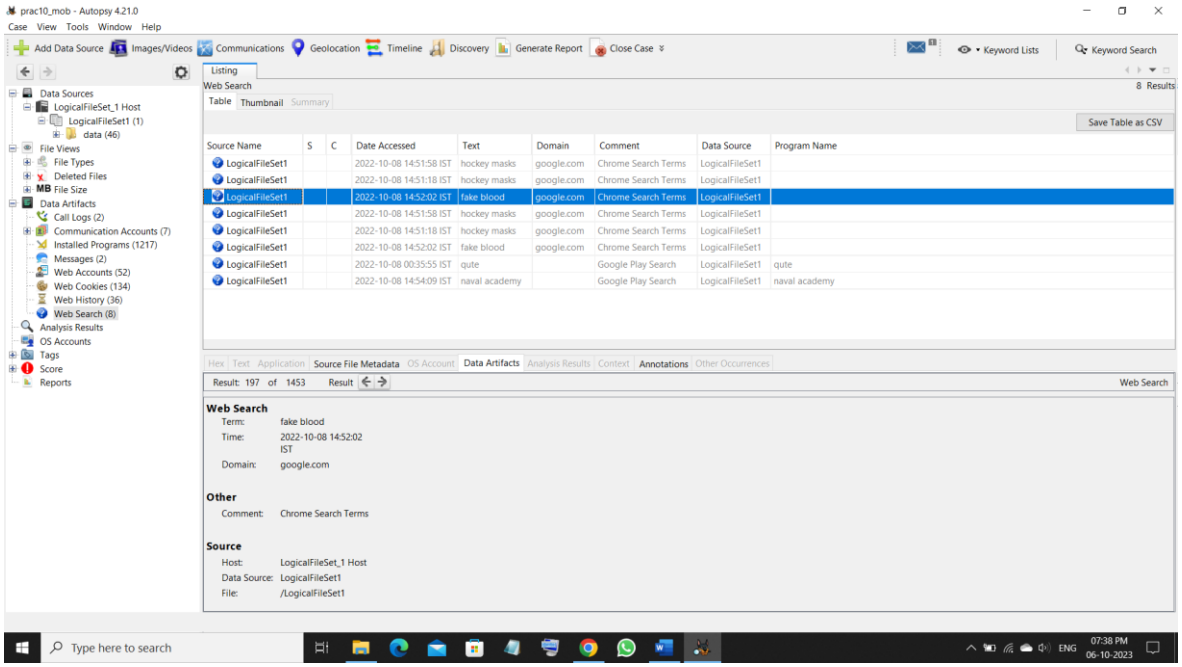
3. Add path of the extracted folder as Data Source -> Click Next.



4. Select Android Analyzer (aLEAPP) -> Click Next -> Click Finish



5. Browse through the call logs, messages,etc.

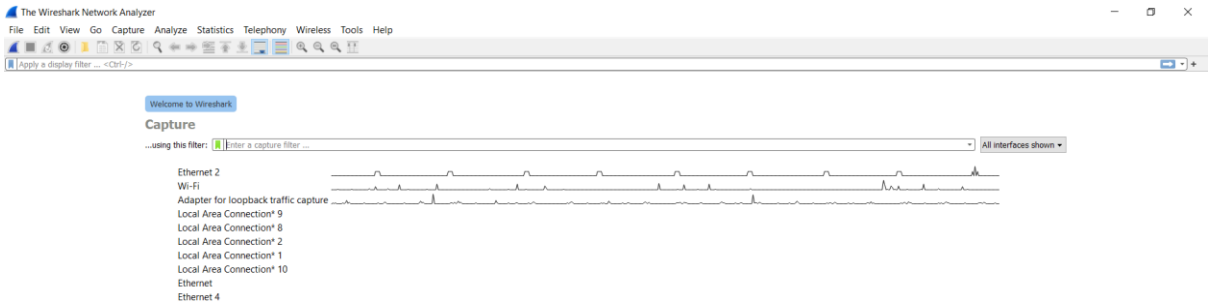


3. Computer Network:

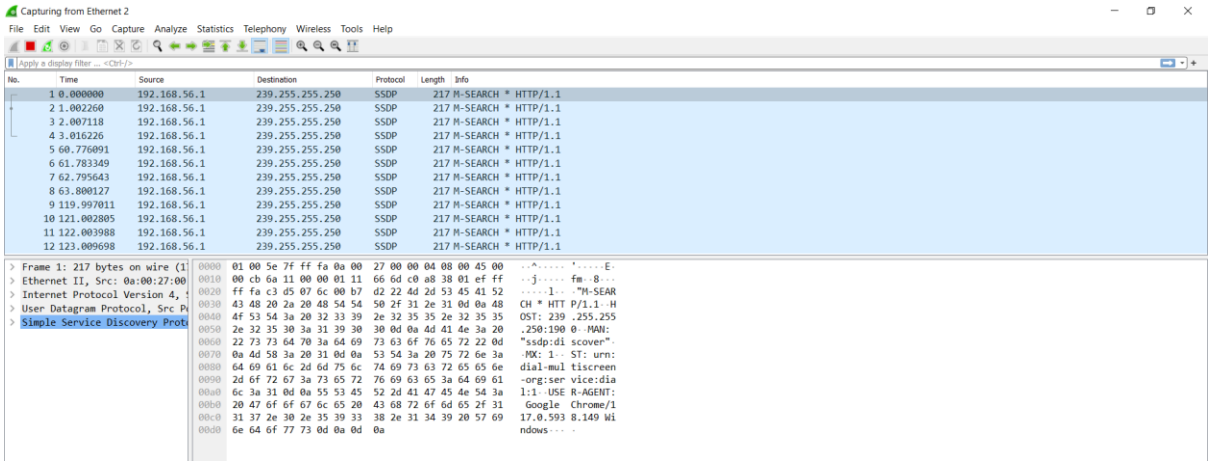
Software used: Wireshark

Steps:

1. Open Wireshark -> Click on Ethernet

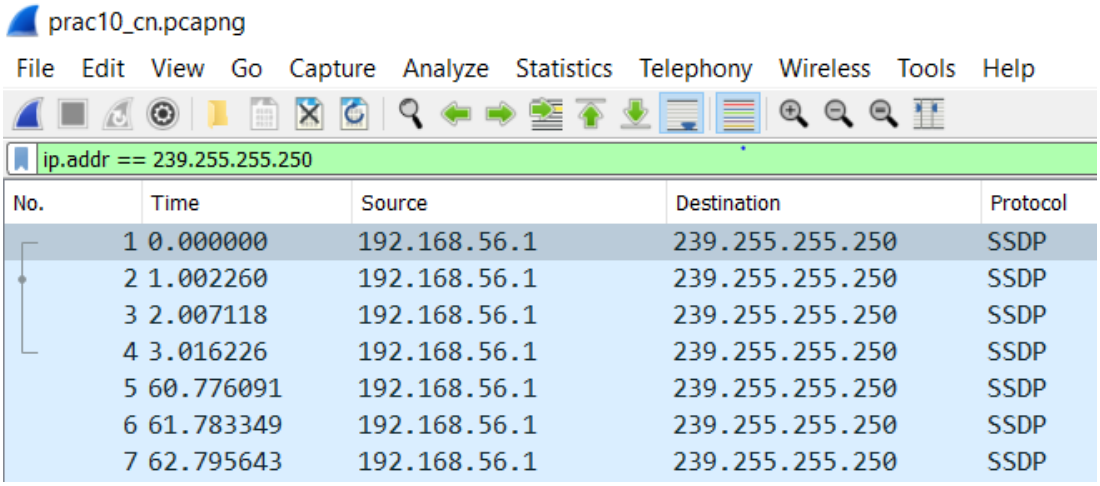


2. It will start capturing packets.



3. Click on Stop to stop capturing packets and save the file. It can be used to analysis the captured network.

4. To search any particular IP address in a network, apply filter -> Type ip.addr == 'IP address'

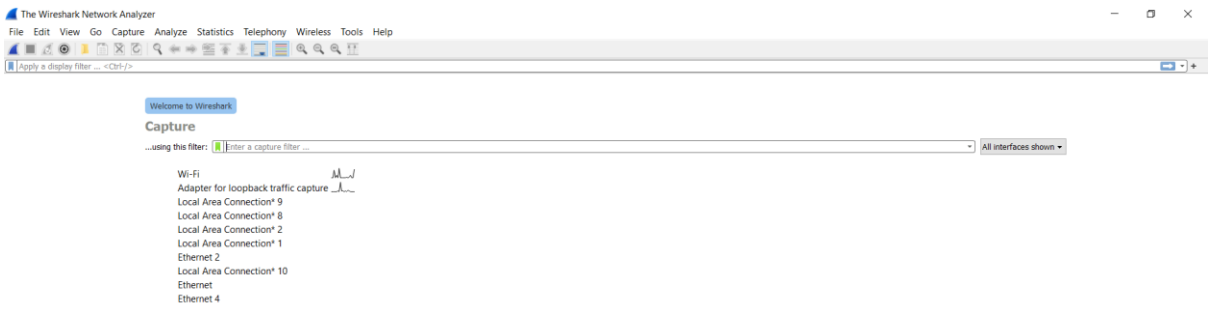


4. Wireless Network:

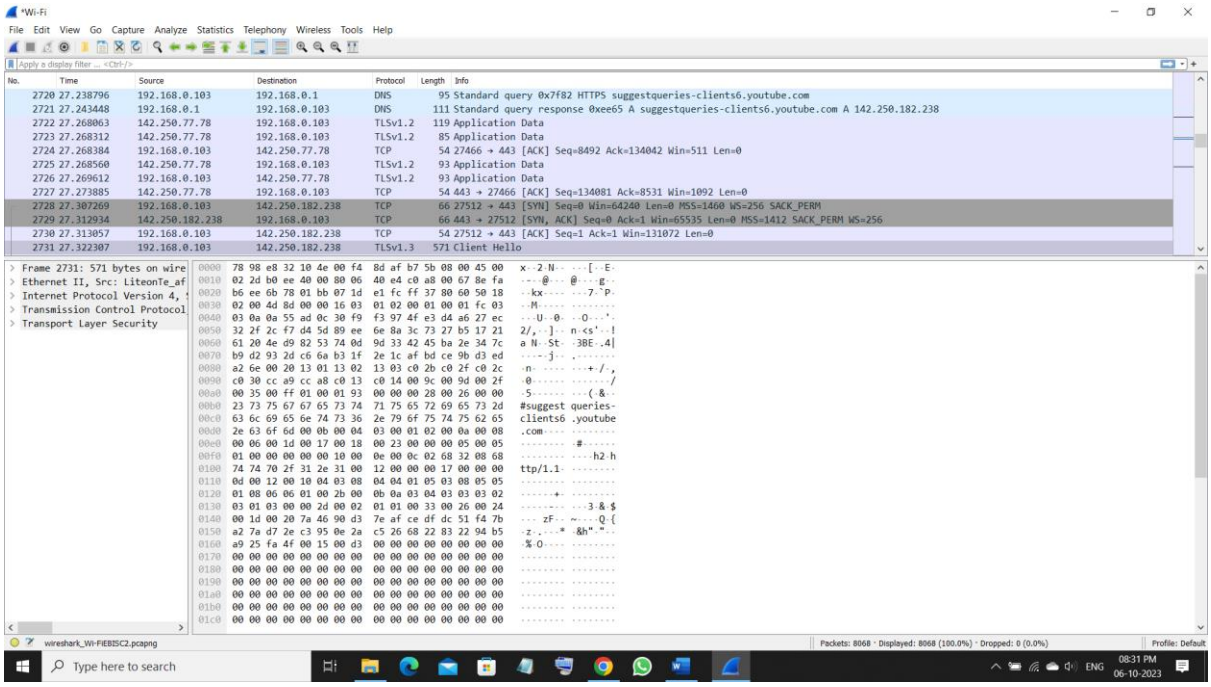
Software used: Wireshark

Steps:

1. Open Wireshark -> Click Wi-Fi

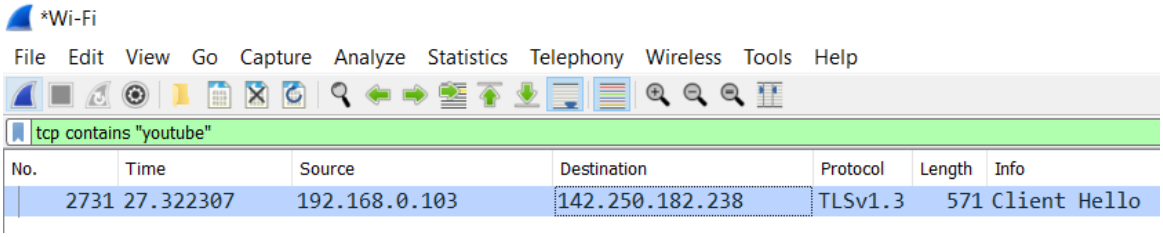


2. Go to browser and go to any website.



3. Click on Stop to stop capturing packets. Save the file.

4. To search visited websites, apply filter -> Type tcp contains “websitename”



Conclusion:

Created a new investigation case using Autopsy. It helped to access and analyze the deleted files, access logs, etc. It also generated case summary or report. Autopsy is also used in creating investigation cases for mobile devices. Wireshark tool helped in monitoring network activity.