Maria Valencia

CSC 153

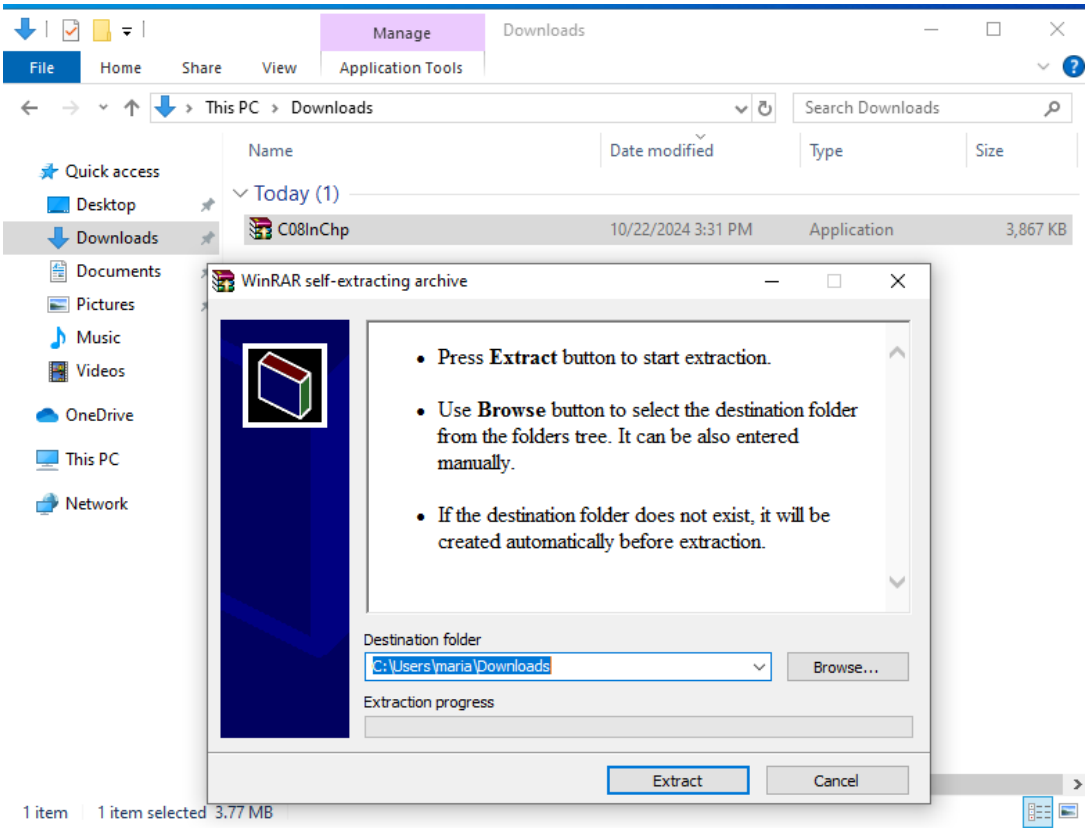
Lab 8

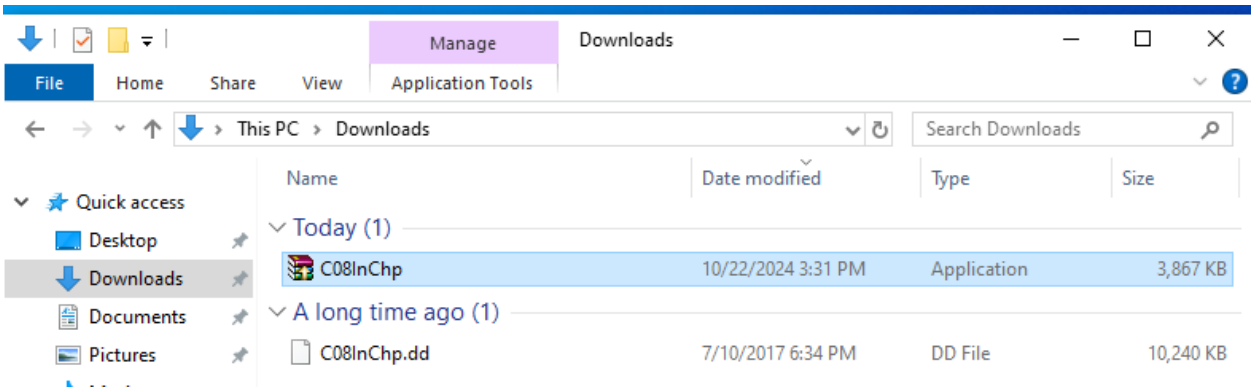
Recovering Graphics Files

**Task 1: Recover Digital Photo Evidence**

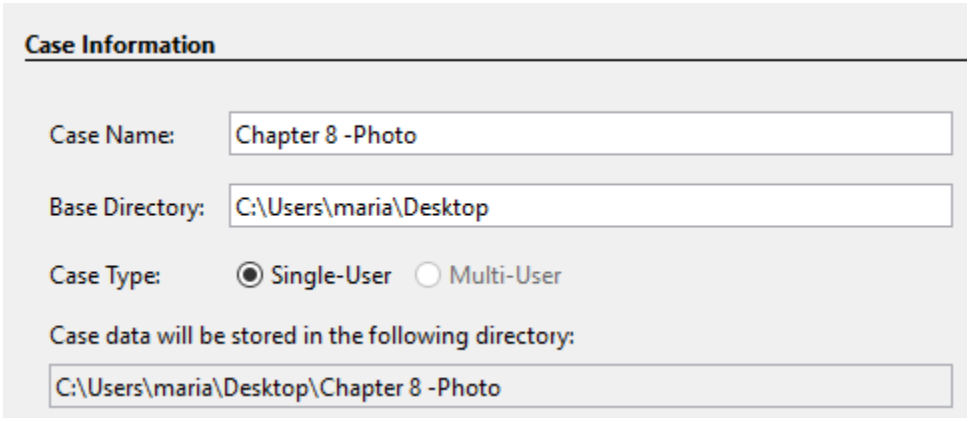
In this task, I discovered obfuscated image files and repaired a manipulated image file to open in my Windows forensics VM.

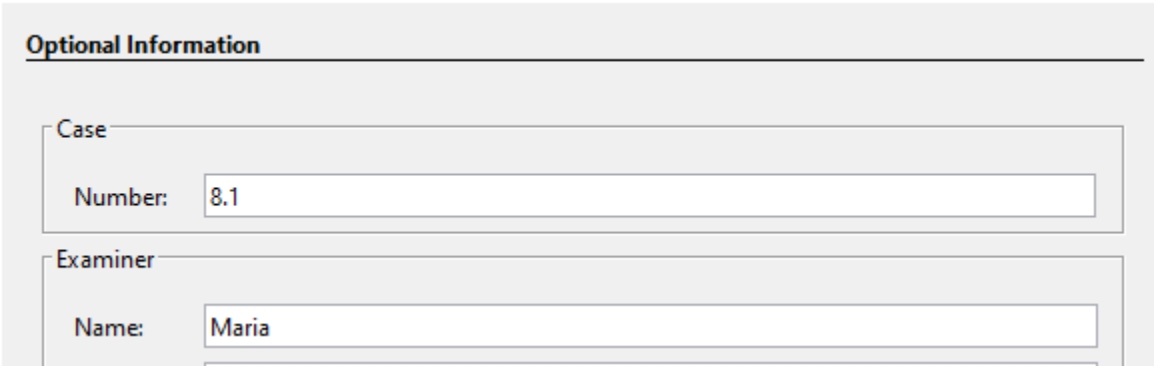
Step 1: I copied the “c08InChp.dd” file onto the Windows VM forensics workstation. I double clicked the exe to decompress the “c08InChp.dd” image file.



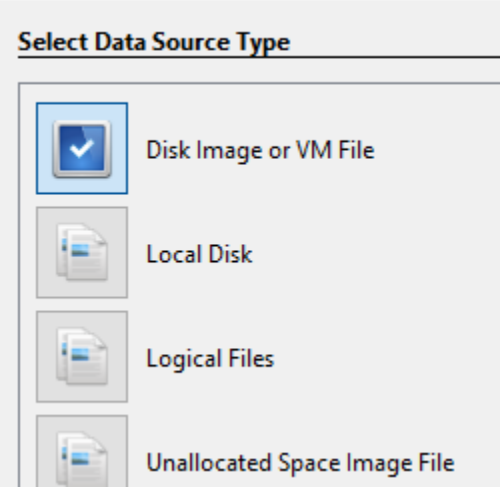


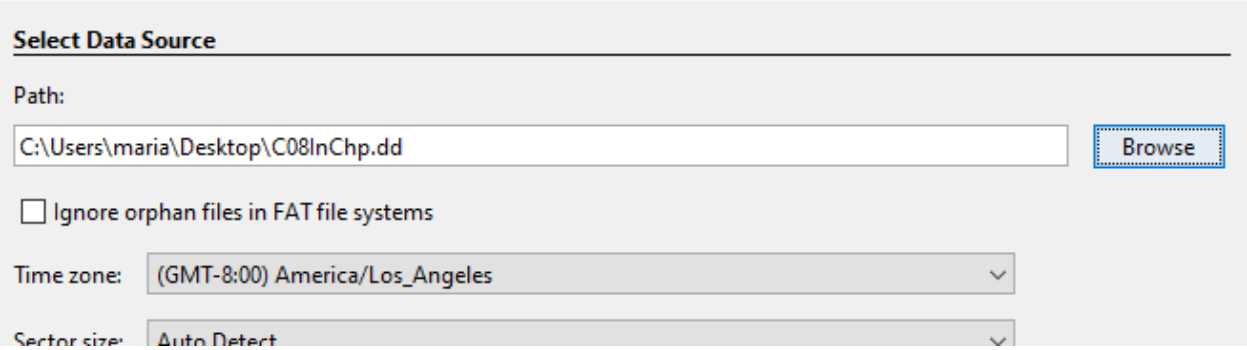

I started autopsy in my Windows VM and created a new case named “Chapter 8 – Photo", name a case ID and entered my name as the examiner.



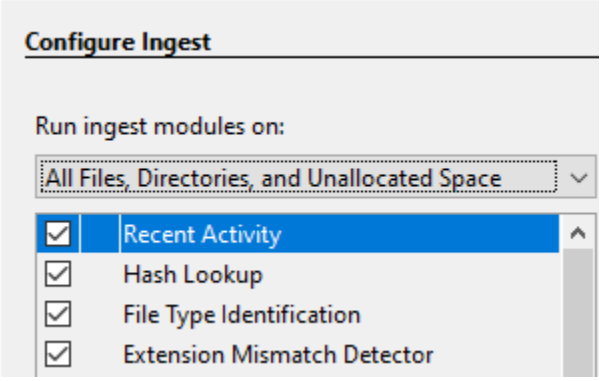



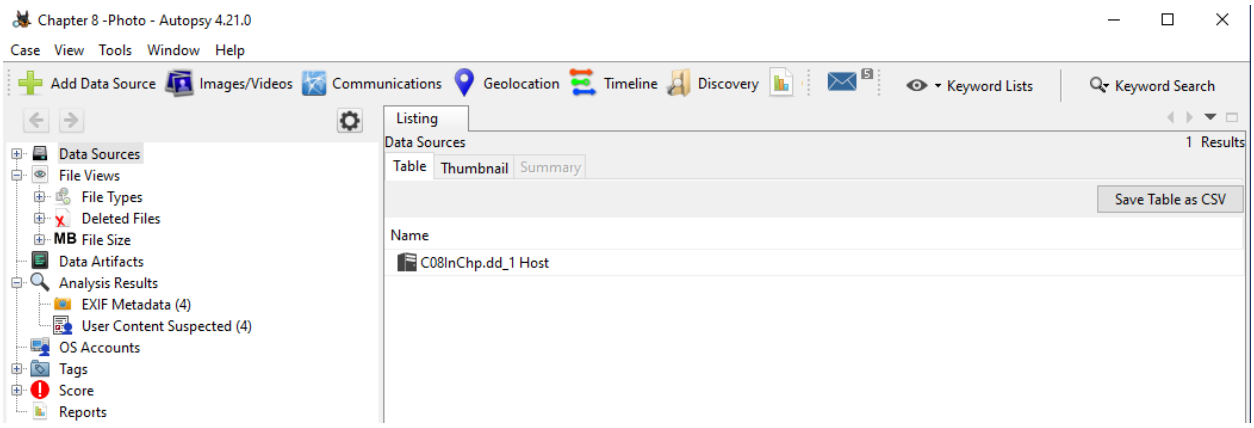
Then, I added a “Disk Image or VM File” as the data source using the “C08InChp.dd” image file.



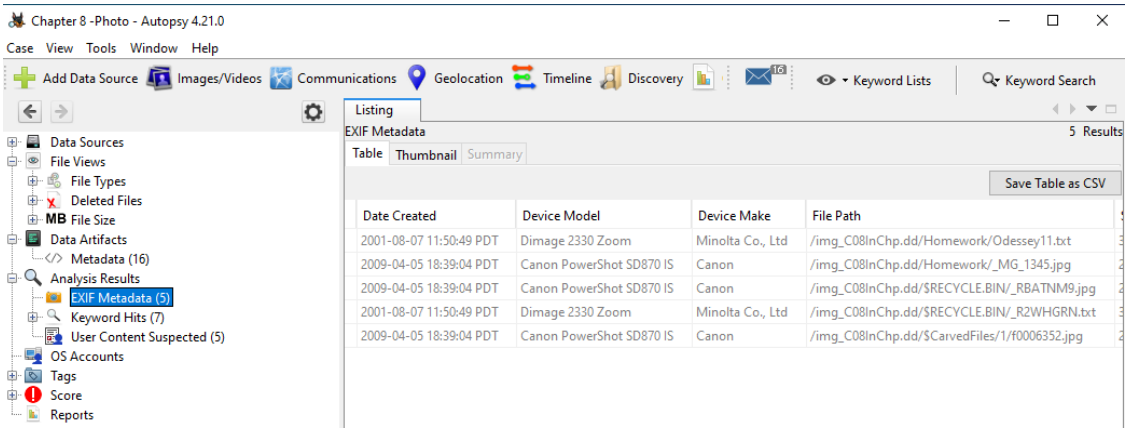

Then I selected to ingest all modules, selected next and then finish.



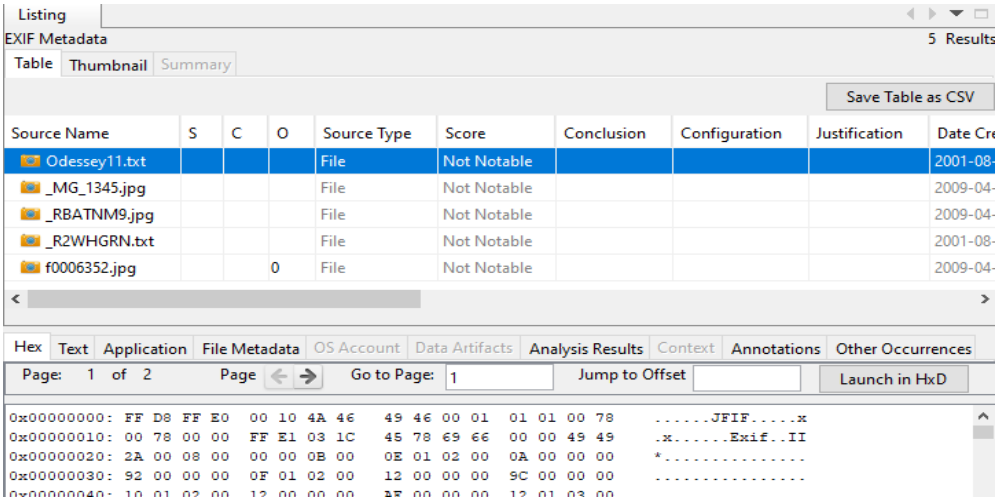



Step 2: Identify Image Files

With the case created and the image loaded, I expanded the Analysis results menu and selected EXIF Metadata in the left navigation pane tree.

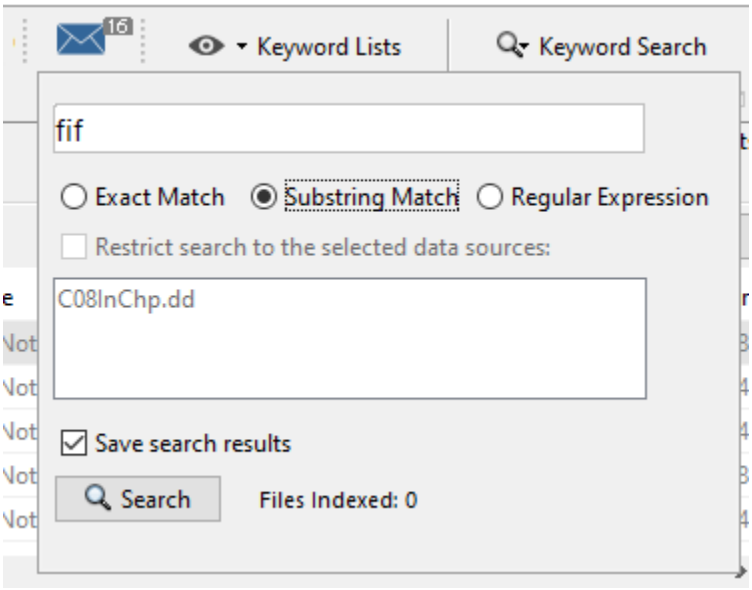



I selected the file “Odessy11.txt” and the hex tab in the viewer pane. Observe that the file name etxension does not match the hex file type code JFIF.

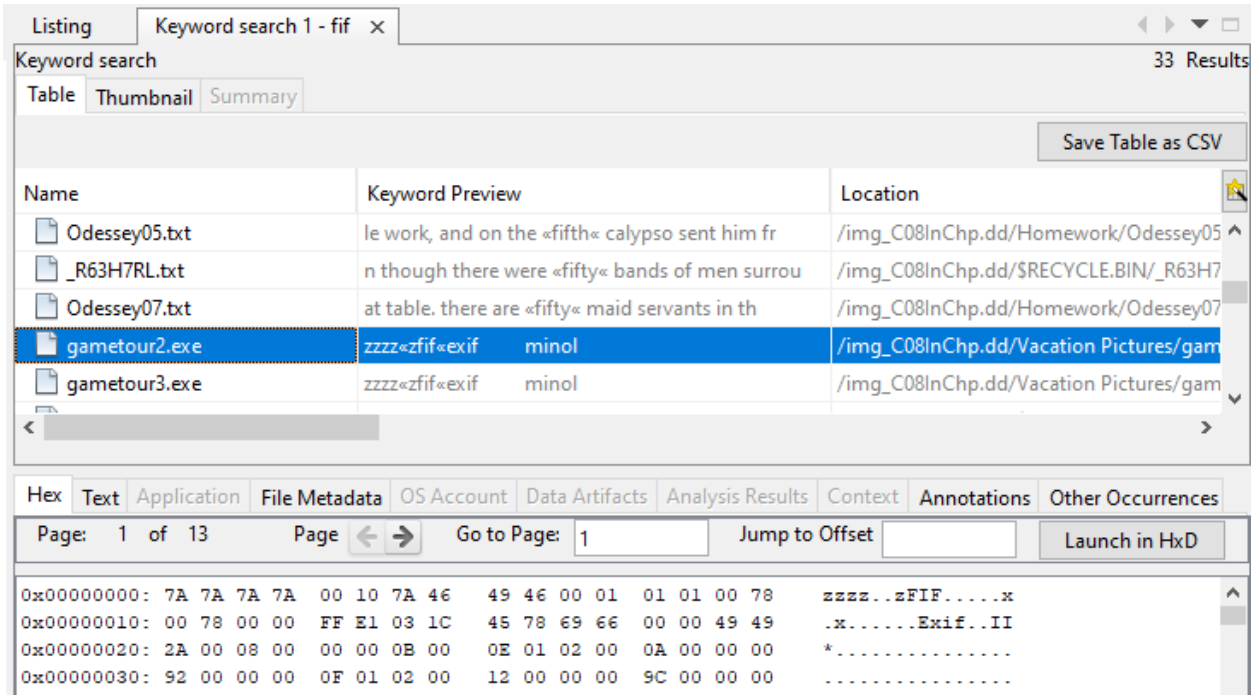


Step 3: Find Manipulated File Types

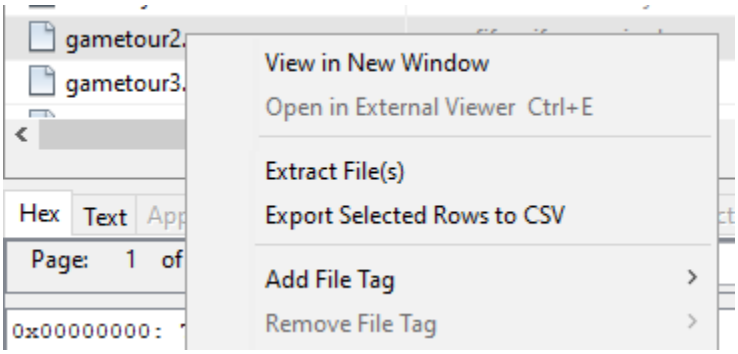
I selected the keyword search in the upper right corner of autopsy. I entered “fif” for the search term and Substring match and rhen pressed the search button.

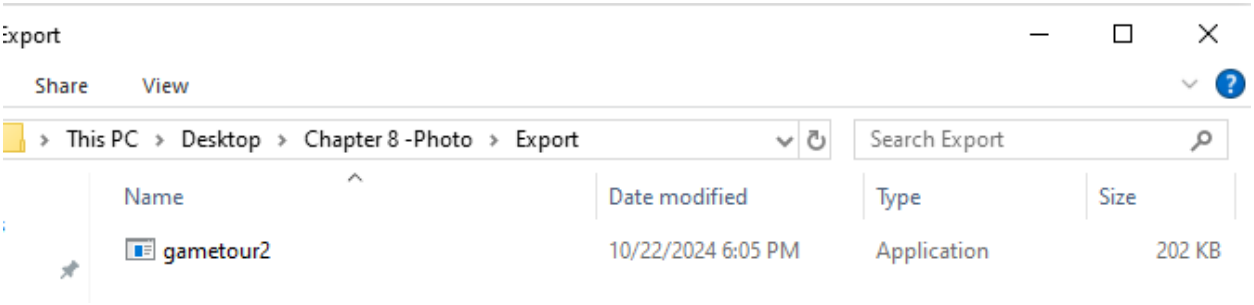



I selected “gametour2.exe” and viewed the hex tab in the preview pane. I observed that the hex file type is “zFIF” which is unusual.



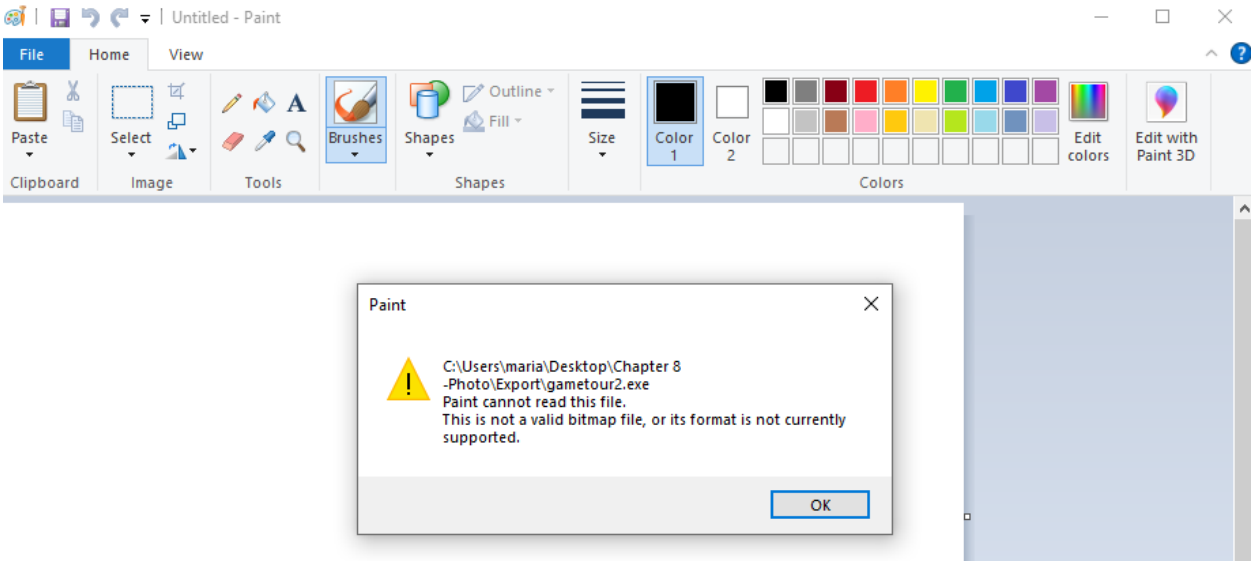
With the “gametour.exe” still selected, I right clicked and selected Extract file(s). I saved the folder in my case directory’s Export Folder.




Step 4: Repair the File Header

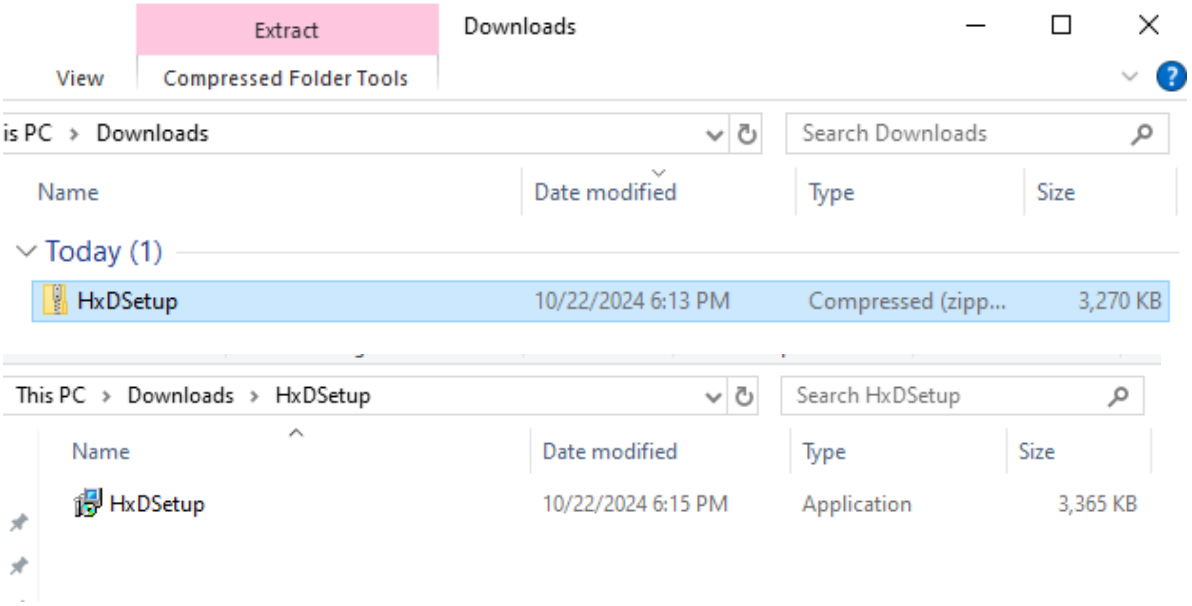
I Launched Microsoft paint, selected file, open, and navigated to my case’s extract folder, selected all and selected “gametour2” file. I observed that Paint can't open the file.



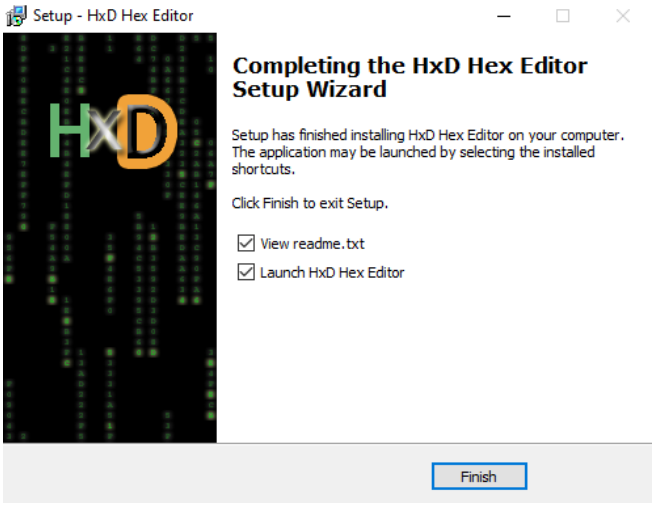

I Downloaded the HxD installer by navigating to [https://mh-nexus.de/en/downloads.php?product=HxD20to](https://mh-nexus.de/en/downloads.php?product=HxD20to) the English download link and selected the download link.

| | | | | | | |
|---|---|---|---|---|---|---|
| HxD20, Greek | installable | 2.5.0.0 | Stefanos Kiourkoulis | February 11, 2021 | Download per HTTPS 3.19 MiB | SHA-1 and SHA-512 |
| HxD20, English | installable | 2.5.0.0 | Maël Hörz | February 11, 2021 | Download per HTTPS 3.19 MiB | SHA-1 and SHA-512 |

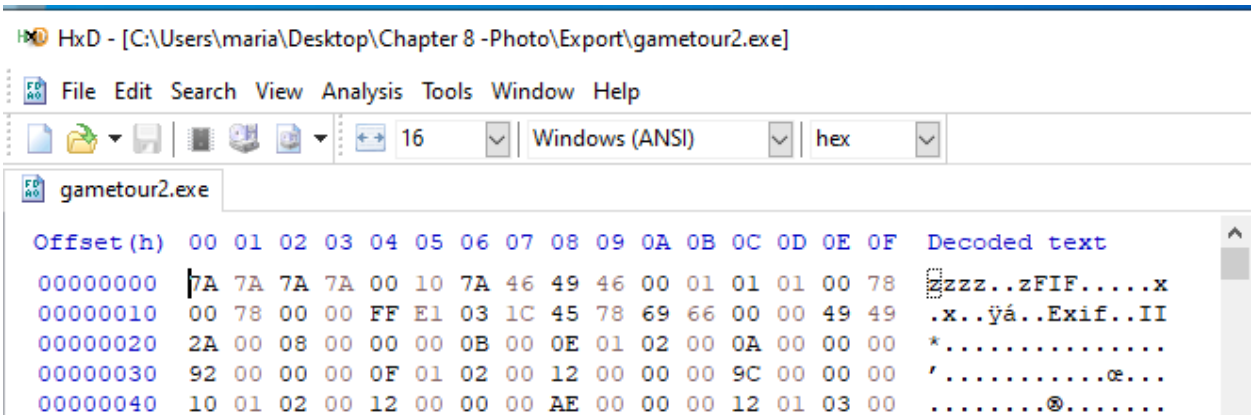
After it was installed, I extracted HxD from the download HxDSetup.zip file in my Downloads folder.



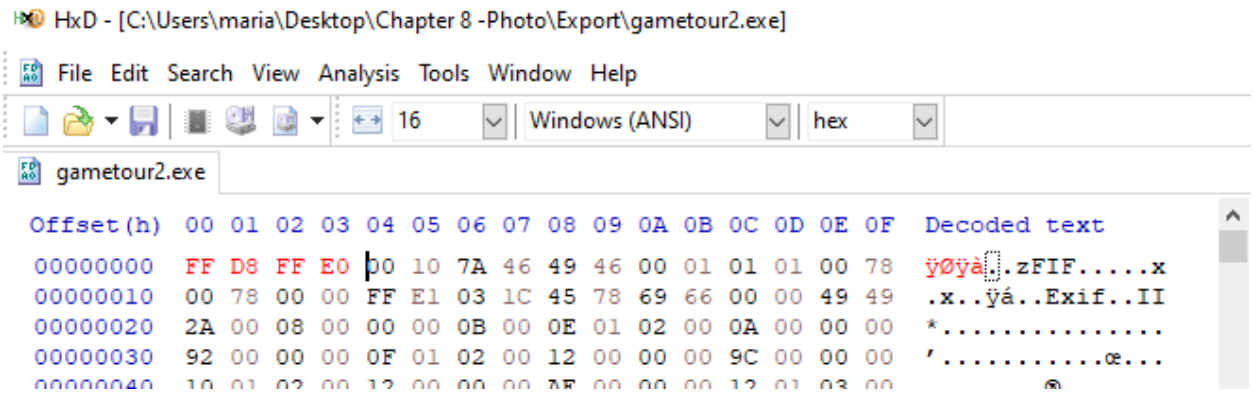

I double-clicked the HxDSetup.exe to begin the installation. I followed the installation wizard defaults by choosing the language, accepted the license agreement and next until the installer is finished.



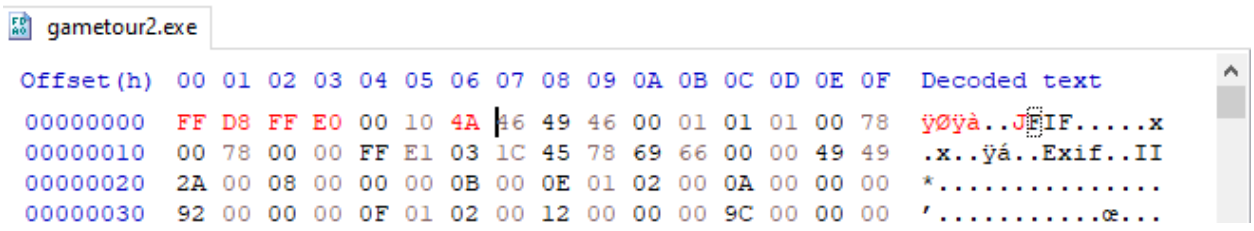

With HxD running, open the “gametour2.exe” by selecting file, open, and navigate to my case folder’s extract folder.



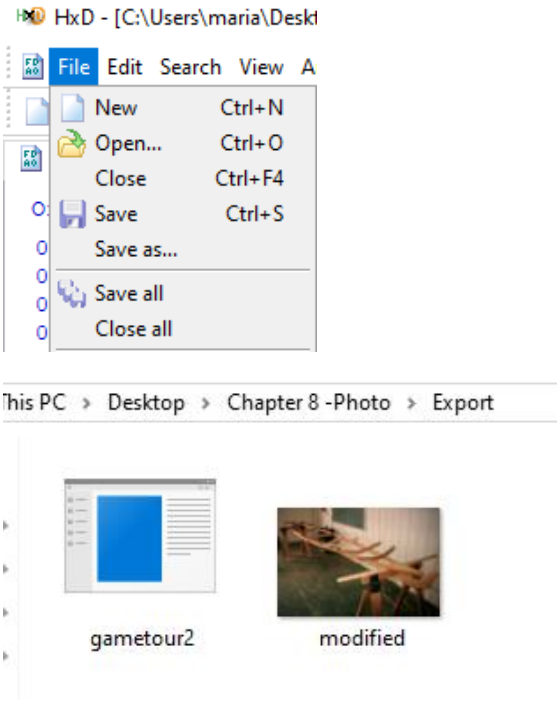

I selected byte 0 (Value 7A) and replace the first 4 bytes with the JFIF header values “FF D8 FF E0”. I observed the change contents have red font.




I selected the “z” in the “zFIF” decoded text section and replaced it with the letter “J”. I observed that this replaces the 6th byte with the value “4A” also now in red font.




I saved the file as “modified.jpeg” under the File and Save as feature.

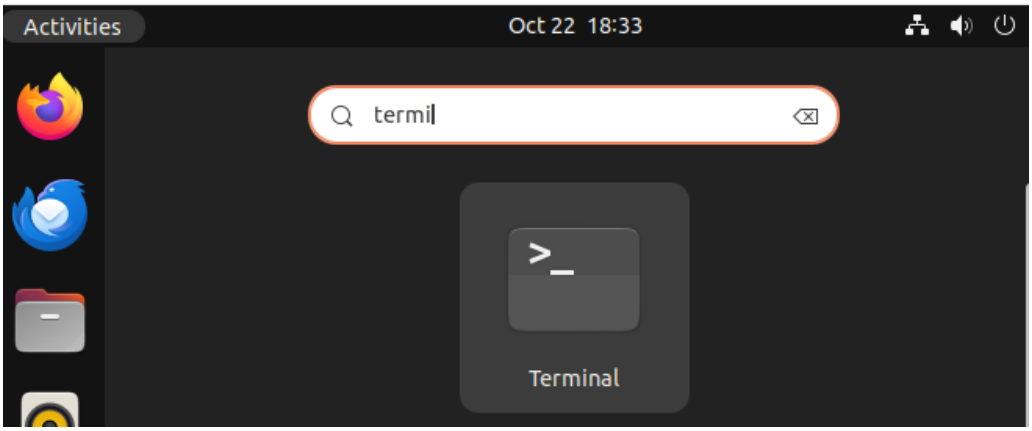



**Task 2: Hide a Message**

I installed Steghide and embedded a secret message in an image from my Ubuntu VM in this task.

Step 1: Install Steghide

I opened a terminal from the Activities menu (upper left corner) and searched for the word Terminal. I installed steghide by updating my system and then downloading and installing the package.

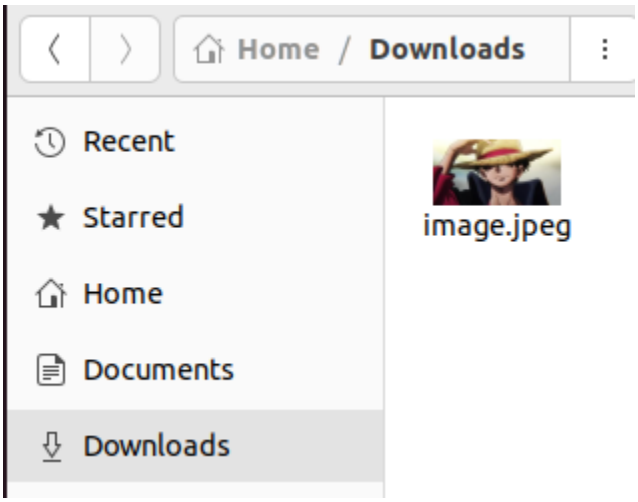


```
maria@ubuntu:~$ sudo apt update -y
[sudo] password for maria:
Hit:1 http://us.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us.archive.ubuntu.com/ubuntu jammy-updates InRelease
 [128 kB]
Get:3 http://security.ubuntu.com/ubuntu jammy-security InRelease
[129 kB]
Get:4 http://us.archive.ubuntu.com/ubuntu jammy-backports InRelea
se [127 kB]
```

```
maria@ubuntu:~$ sudo apt install steghide -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libmcrypt4
Suggested packages:
  libmcrypt-dev mcrypt
The following NEW packages will be installed:
  libmcrypt4 steghide
0 upgraded, 2 newly installed, 0 to remove and 36 not upgraded.
Need to get 213 kB of archives.
After this operation, 701 kB of additional disk space will be use
d.
Get:1 http://us.archive.ubuntu.com/ubuntu jammy/universe amd64 li
```

Step 2: Obtain a JPG

I downloaded a JPG file from the internet and named it “image.jpg” and ensured it is a JPG file.



Step 3: Create a secret message

I created a secret message to hide in the JPG image.

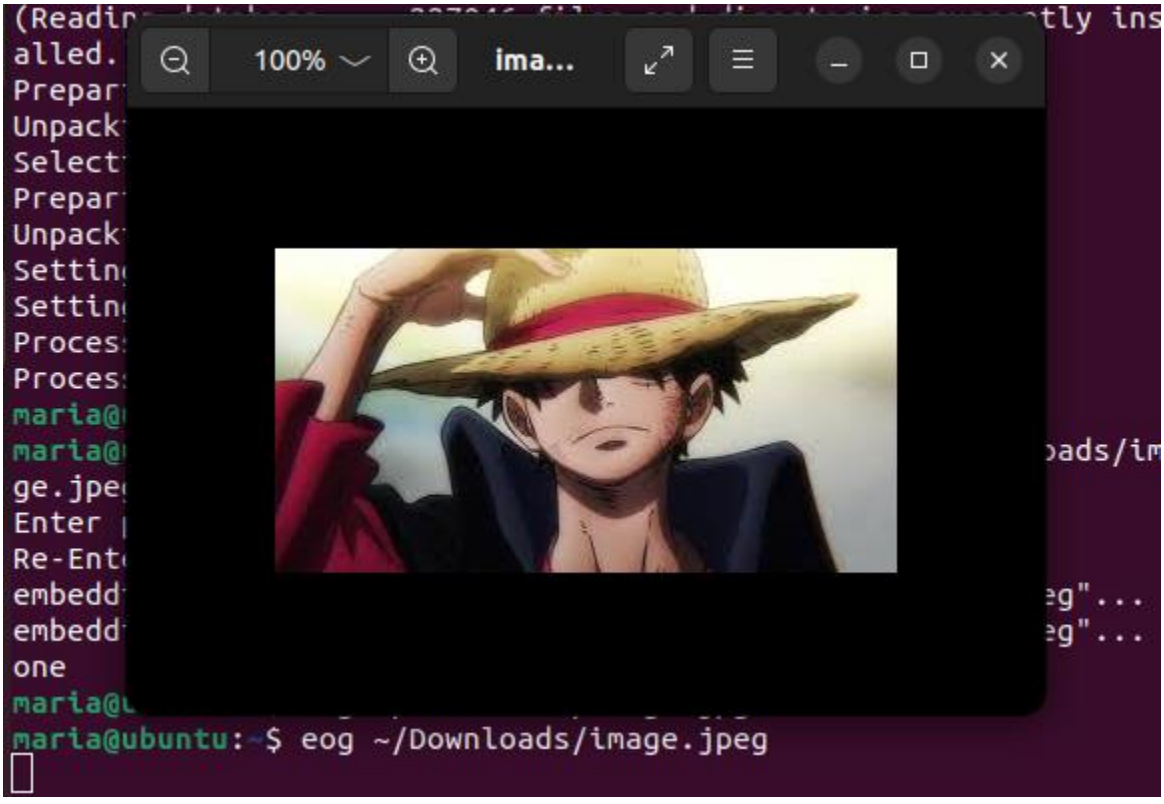
Echo “launch codes: 123123” > secret.txt

```
maria@ubuntu:~$ echo "Launch Codes: 123123" > secret.txt
maria@ubuntu:~$
```

Step 4: Hide the Message

I hid the secret message created in the previous within the JPG image from step 2. I opened the image using eog and observed there is no observable difference from the original image.

```
maria@ubuntu:~$ steghide embed -ef secret.txt -cf ~/Downloads/ima
ge.jpeg
Enter passphrase:
Re-Enter passphrase:
embedding "secret.txt" in "/home/maria/Downloads/image.jpeg"... 0
embedding "secret.txt" in "/home/maria/Downloads/image.jpeg"... d
one
maria@ubuntu:~$
```



Step 5: Extract the Secret

I navigated to my image file and extracted the secret file from the image. I observed the image.