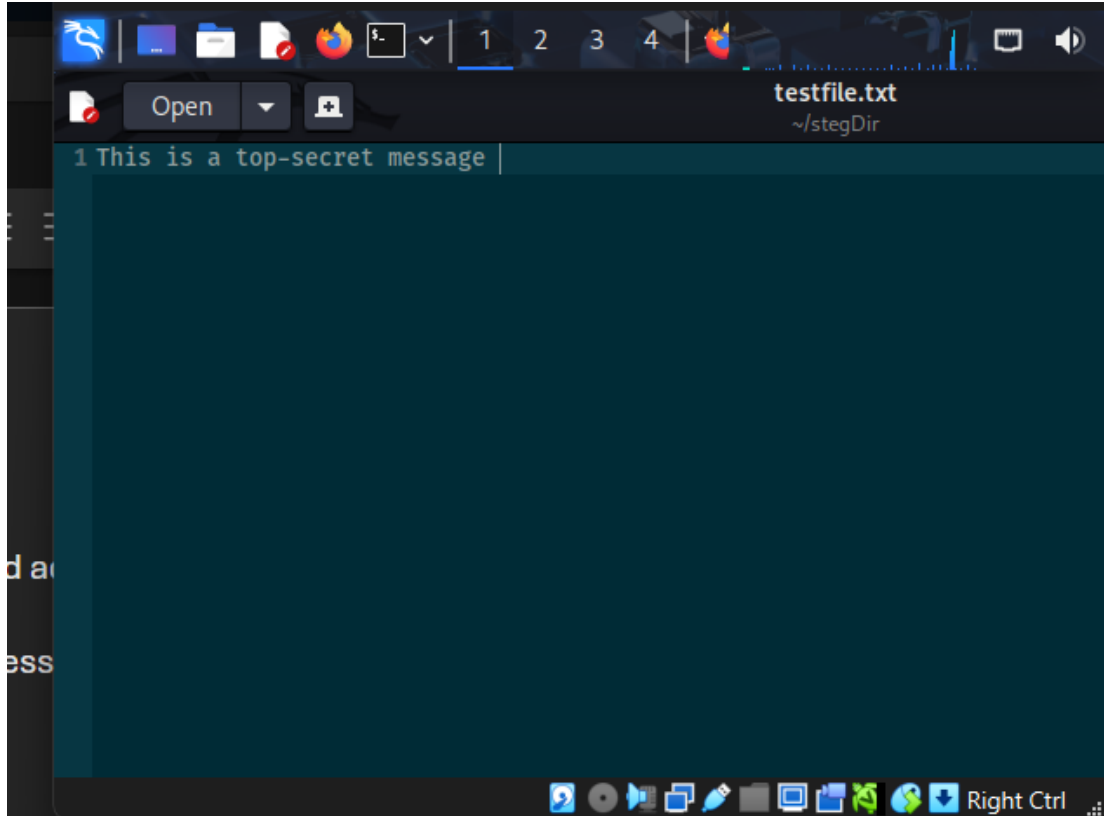


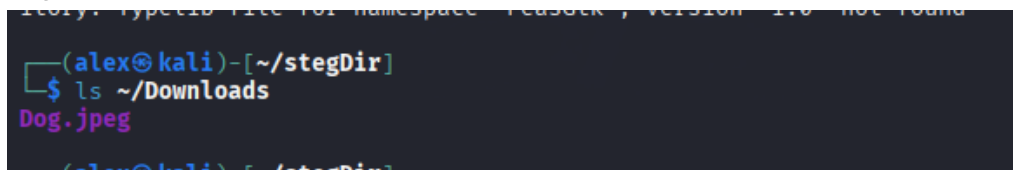
4. Open the file testfile.txt using gedit and add some secret message there as the file content.

Take a screenshot showing the secret message you added.



6. Copy the image from the Downloads directory to the stegDir directory using the cp command. The stegDir directory should have two files by now: testfile.txt and dog.jpeg.

Use ls command to show the contents of the stegDir directory and take a screenshot to attach it in your submission.



7. Execute the md5sum command to check the checksums for both testfile.txt and dog.jpeg.

Learn about MD5 here: <https://phoenixnap.com/kb/md5sum-linux>). Take a screenshot similar to the following screenshot

```
(alex@kali)~[~/stegDir]
$ md5sum testfile.txt Dog.jpeg
b8cdb0a0e4c6e3e142b10fca42376a7 testfile.txt
ff5d7f9463524cf6bb6aaca4178a8103 Dog.jpeg
```

8. Learn about steghide command here:

<https://manpages.ubuntu.com/manpages/trusty/man1/steghide.1.html>.

Use the steghide command to embed your testfile.txt (with secret message) into the image file dog.jpeg as shown in the following example screenshot (note: when prompted for the passphrase, you may type any password of your choice)

```
(alex@kali)~[~/stegDir]
$ steghide embed -cf Dog.jpeg -ef testfile.txt
Enter passphrase:
Re-Enter passphrase:
embedding "testfile.txt" in "Dog.jpeg" ... done
```

9. Execute the command md5sum for dog.jpeg to check the hash for the image file. Do you see any difference? Take a screenshot showing the command and the output hash.

```
(alex@kali)~[~/stegDir]
$ md5sum Dog.jpeg
0d9ef1059cc52fdd9fb410dbff5a5d1b Dog.jpeg
```

10. Execute the steghide command to get some information about dog.jpeg before extracting it, use the info command as shown in this following example screenshot:

```
(alex@kali)~[~/stegDir]
$ steghide info Dog.jpeg
"Dog.jpeg":
  format: jpeg
  capacity: 534.0 Byte
Try to get information about embedded data ? (y/n) y
Enter passphrase:
  embedded file "testfile.txt":
    size: 30.0 Byte
    encrypted: rijndael-128, cbc
    compressed: yes
```

11. Now, delete the file testfile.txt using the rm command. Use the ls command to show the contents of the stegDir directory and take a screenshot.

```
Compressed: yes
(alex@kali)-[~/stegDir]
$ rm testfile.txt

(alex@kali)-[~/stegDir]
$ ls
Dog.jpeg
```

12. . Extract the secret message by executing the steghide command with - - extract option as shown in the following example screenshot

```
(alex@kali)-[~/stegDir]
$ steghide --extract -sf Dog.jpeg
Enter passphrase:
wrote extracted data to "testfile.txt".

(alex@kali)-[~/stegDir]
```

13. Execute the ls command to list the contents in the stegDir directory. You should see testfile.txt there because it was hidden in the dog.jpeg image file and appeared after extracting the image file in the previous step (step-12). Take a screenshot showing the contents of the stegDir directory.

```
wrote extracted data to "testfile.txt".

(alex@kali)-[~/stegDir]
$ ls
Dog.jpeg testfile.txt
```

See the

14. contents of the file testfile.txt using gedit. Take a screenshot showing the contents.

```
Open ~/stegDir
1 This is a top-secret message
```

15. See the metadata of the file dog.jpeg using the Exif tool command as shown in the following example screenshot:

```
└─$ exiftool Dog.jpeg
ExifTool Version Number      : 13.25
File Name                    : Dog.jpeg
Directory                   : .
File Size                    : 9.5 kB
File Modification Date/Time   : 2025:10:30 17:09:18-04:00
File Access Date/Time        : 2025:10:30 17:10:30-04:00
File Inode Change Date/Time   : 2025:10:30 17:09:18-04:00
File Permissions              : -rw-rw-r--
File Type                    : JPEG
File Type Extension          : jpg
MIME Type                    : image/jpeg
JFIF Version                 : 1.01
Resolution Unit               : None
X Resolution                  : 1
Y Resolution                  : 1
Image Width                   : 275
Image Height                  : 183
Encoding Process              : Baseline DCT, Huffman coding
Bits Per Sample               : 8
Color Components              : 3
Y Cb Cr Sub Sampling          : YCbCr4:2:0 (2 2)
Image Size                    : 275x183
Megapixels                    : 0.050
```

16. Change the author of the file dog.jpeg using the exiftool command as shown in the following example screenshot:

Note: when you enter the exiftool command in the terminal to update the author's name, make sure you replace "Alice" with your own name.

Old Dominion University

CYSE 450: Ethical Hacking and Penetration Testing

15. See the metadata of the file dog.jpeg using the exiftool command as shown in the following example screenshot:

example screenshot:

17. Repeat the step-15 and take a screenshot showing the updated metadata of the file dog.jpeg. Highlight the author's name in the screenshot

```
(alex@kali)-[~/stegDir]
$ exiftool -author=Alice Dog.jpeg
1 image files updated

(alex@kali)-[~/stegDir]
$ exiftool Dog.jpeg
ExifTool Version Number      : 13.25
File Name                    : Dog.jpeg
Directory                    : .
File Size                    : 12 kB
File Modification Date/Time   : 2025:10:30 17:30:25-04:00
File Access Date/Time        : 2025:10:30 17:30:25-04:00
File Inode Change Date/Time   : 2025:10:30 17:30:25-04:00
File Permissions              : -rw-rw-r--
File Type                    : JPEG
File Type Extension          : jpg
MIME Type                    : image/jpeg
JFIF Version                 : 1.01
Resolution Unit               : None
X Resolution                  : 1
Y Resolution                  : 1
XMP Toolkit                   : Image::ExifTool 13.25
Author                       : Alice
Image Width                  : 275
Image Height                  : 183
Encoding Process              : Baseline DCT, Huffman coding
Bits Per Sample               : 8
Color Components              : 3
Y Cb Cr Sub Sampling          : YCbCr4:2:0 (2 2)
Image Size                   : 275x183
```

18. Execute the md5sum command for dog.jpeg. Do you see any change in the hash value? If yes, take a screenshot of the new hash and compare it with the previous hash you received in step-9 (**There is a huge change in the harsh values as I compare it to step 9s screenshot to step 18s screenshot it has drastically.**

```
.Step 18 (alex@kali)-[~/stegDir]
$ md5sum Dog.jpeg
d18b2768d227821124ff8e7c4447801d Dog.jpeg
```

```
Step 9 (alex@kali)-[~/stegDir]
$ md5sum Dog.jpeg
0d9ef1059cc52fdd9fb410dbff5a5d1b Dog.jpeg
```