Lab 14: Cryptography Concepts

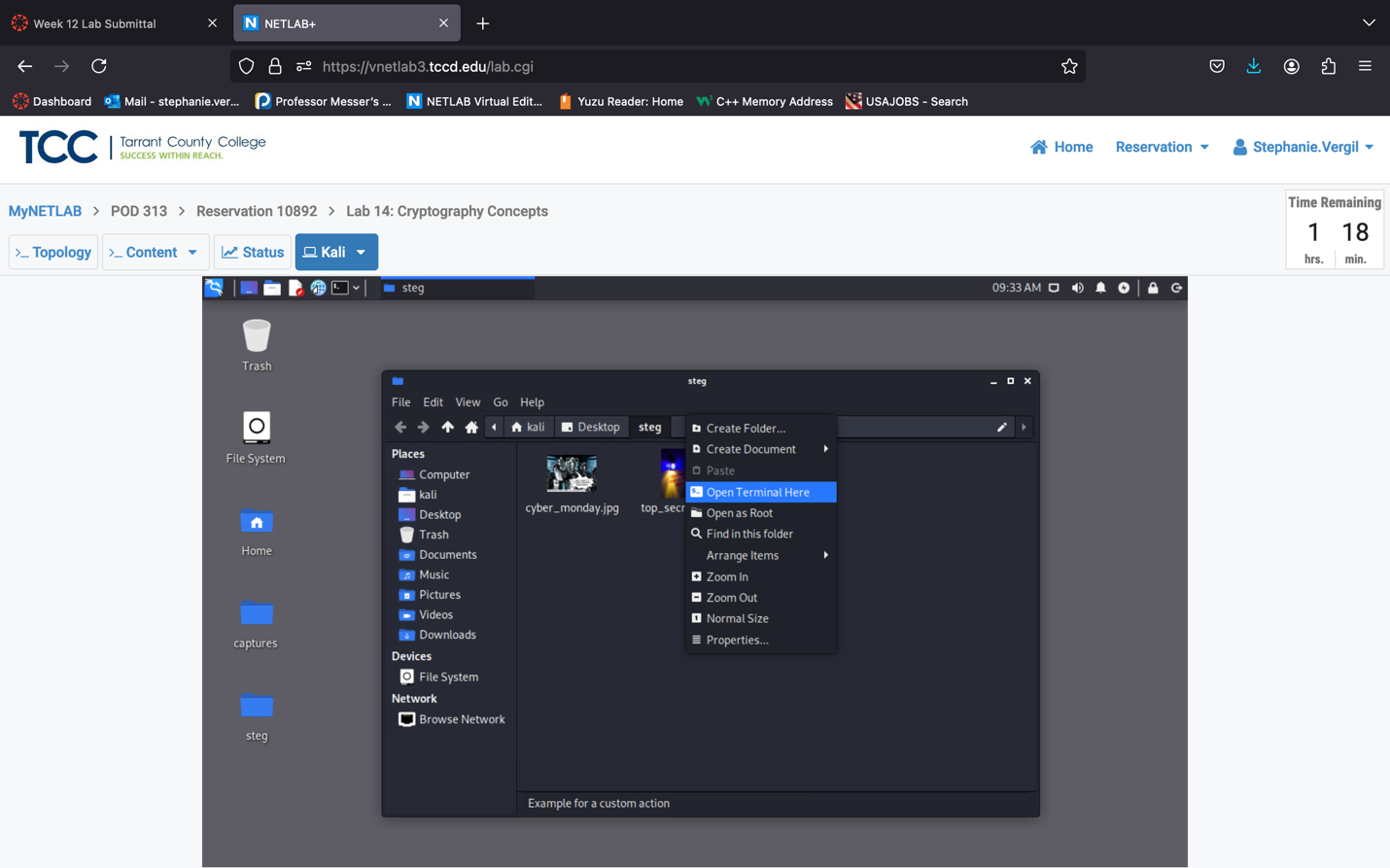
1 Hiding a Hidden Message Within a Picture

* 1. Using Steghide to Hide Hidden Messages

1. Launch Kali VM
2. Log in with credentials provided
3. Launch the steg folder located on the Desktop, right click on an empty space and click on open terminal here.

A screenshot of a computer

Description automatically generated



1. Enter the following command on the terminal to create a new text document with a secret text string;

echo “The password to the WinOS is NDGlabpass123\!” > secret.txt

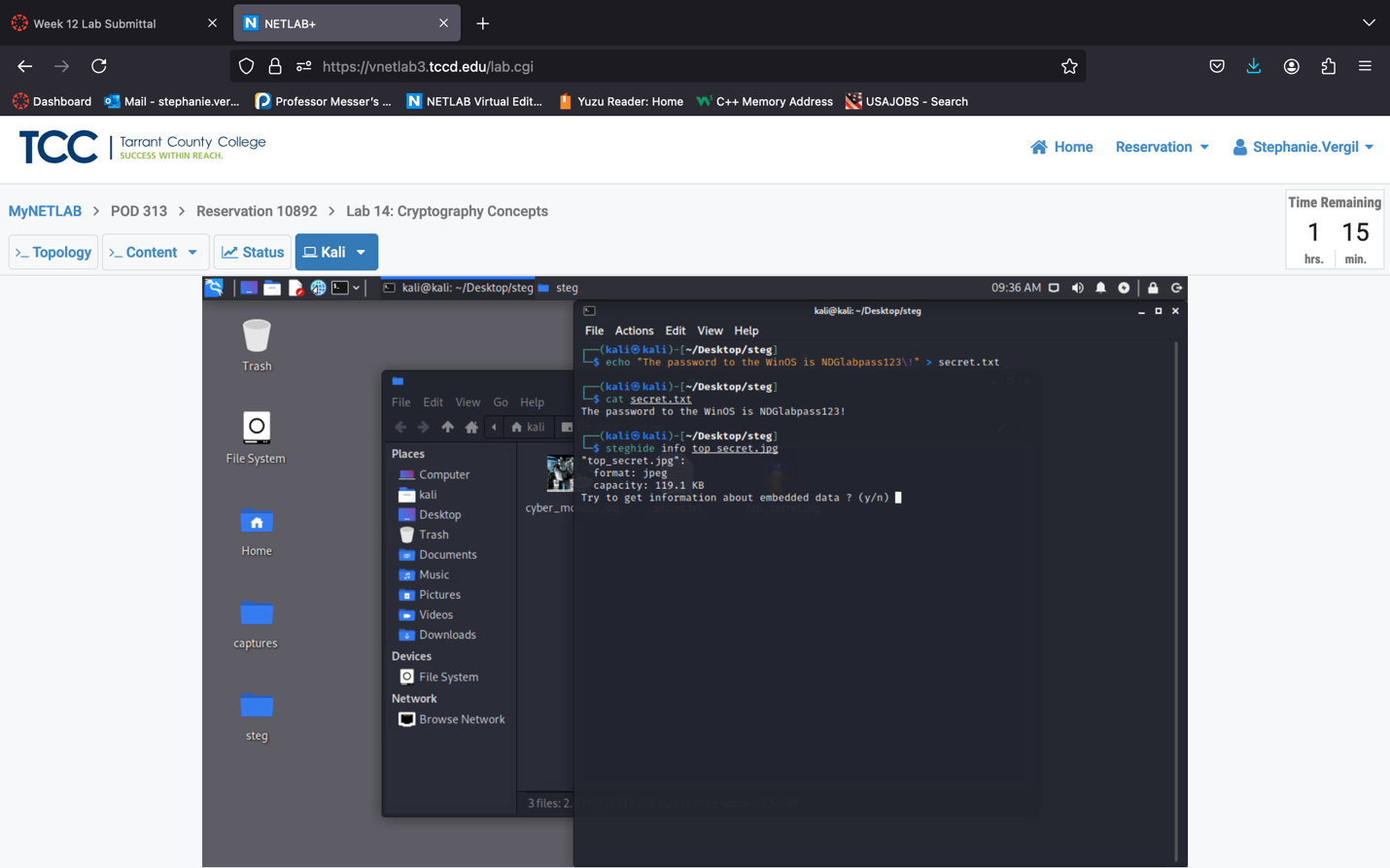
1. Enter the following command to double-check the contents of the secret.txt file:

cat secret.txt

1. Enter the following command to verify the capacity of the top\_secret.jpg image to see what the capacity is for being able to hide a message within the image itself:

steghide info top\_secret.jpg

1. Check the capacity amount. When asked to get information about embedded data type n

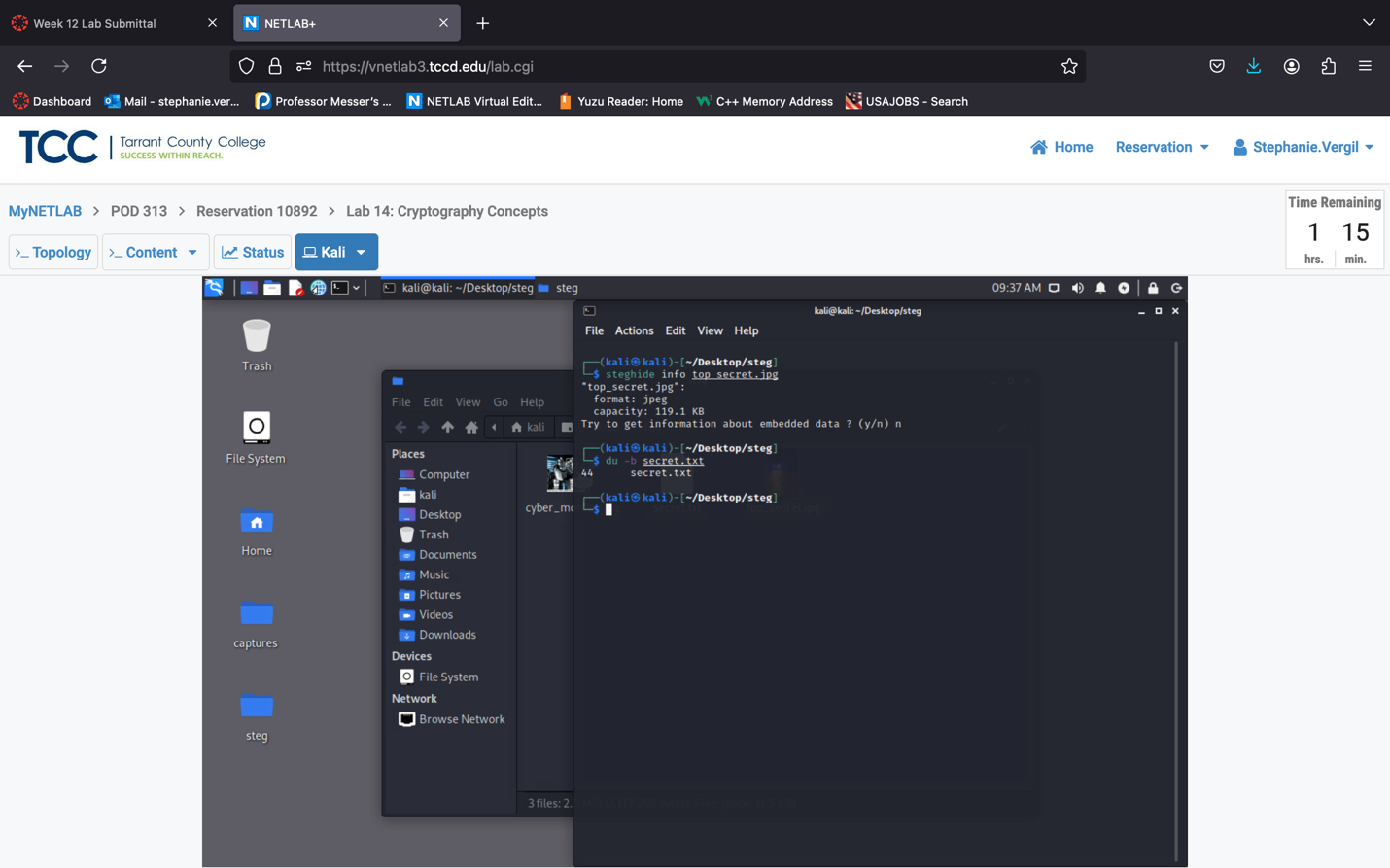


1. Enter the following command to verify how large the secret.txt file is to confirm whether we can hide it within the top\_secret.jpg image. Using the du with -b options to display the file size in terms of bytes:

du -b secret.txt

Note: Notice that we should be able to fit the 44 byte sized secret.txt file in

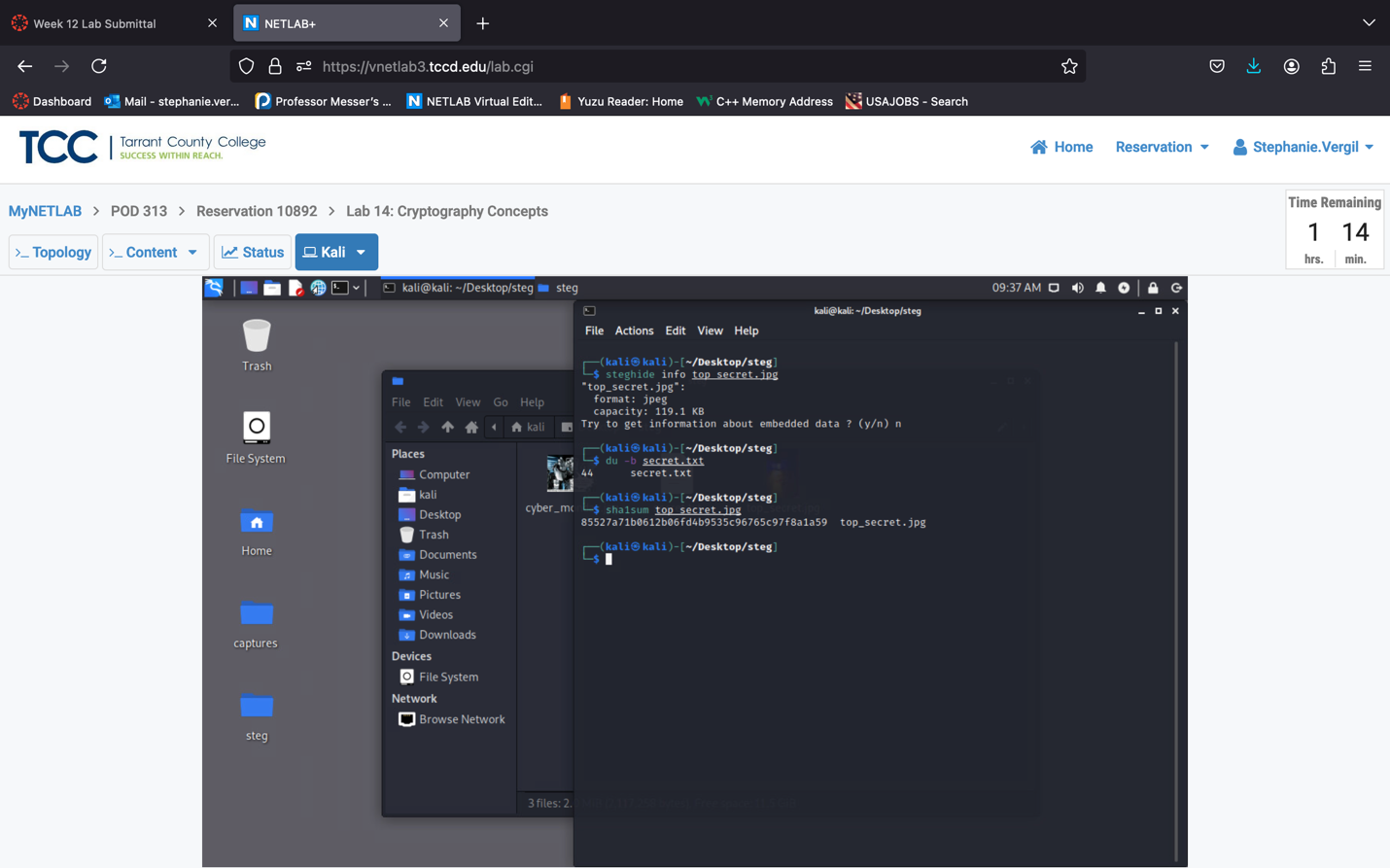
the 3.1 KB top\_secret.jpg image file.



1. Enter the following command before we embed the secret message, to confirm the sha1 hash value for the top\_secret.jpg image file:

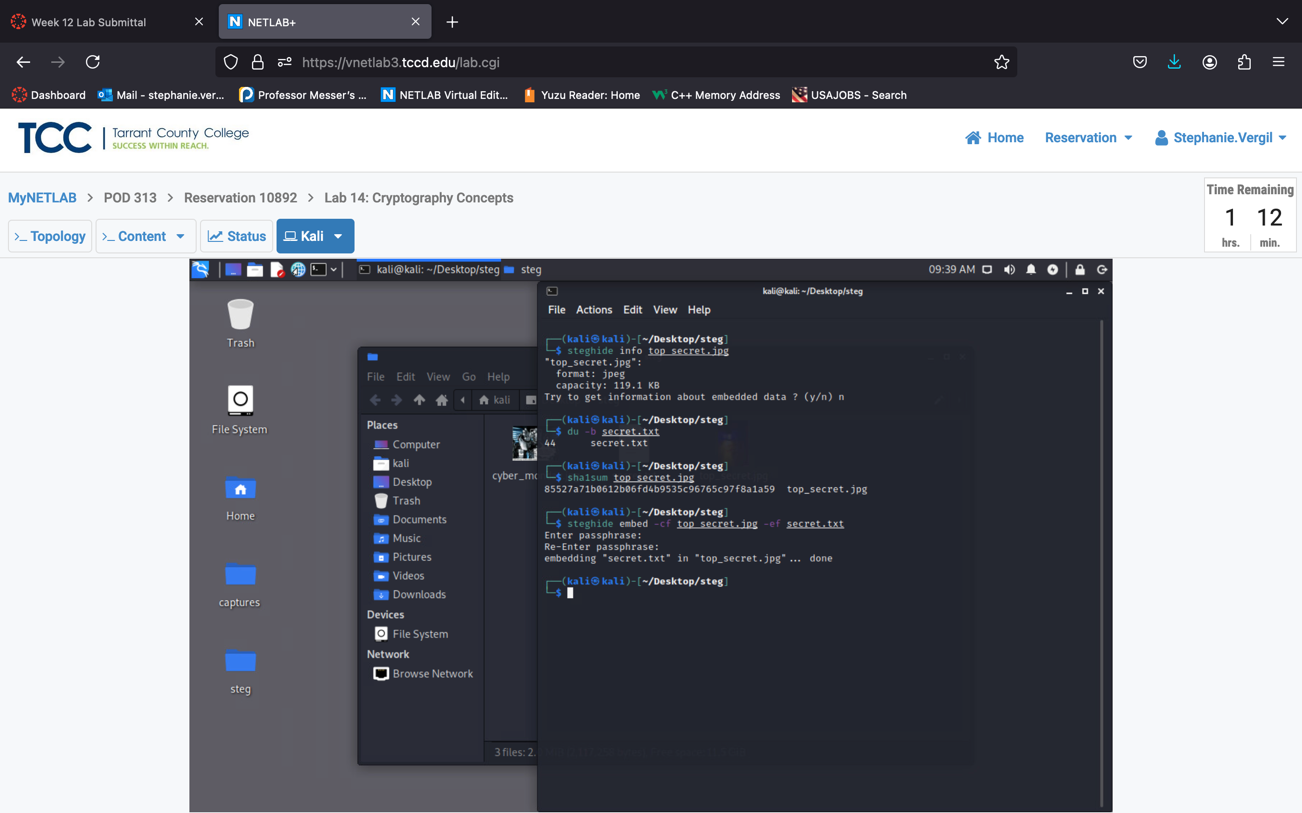
sha1sum top\_secret.jpg

Note: Take note of this hash value for later comparison.



1. Enter the following command to initialize the process of hiding the secret message, when prompt for a passphrase type secret press enter and reverify passphrase:

steghide embed -cf top\_secret.jpg -ef secret.txt

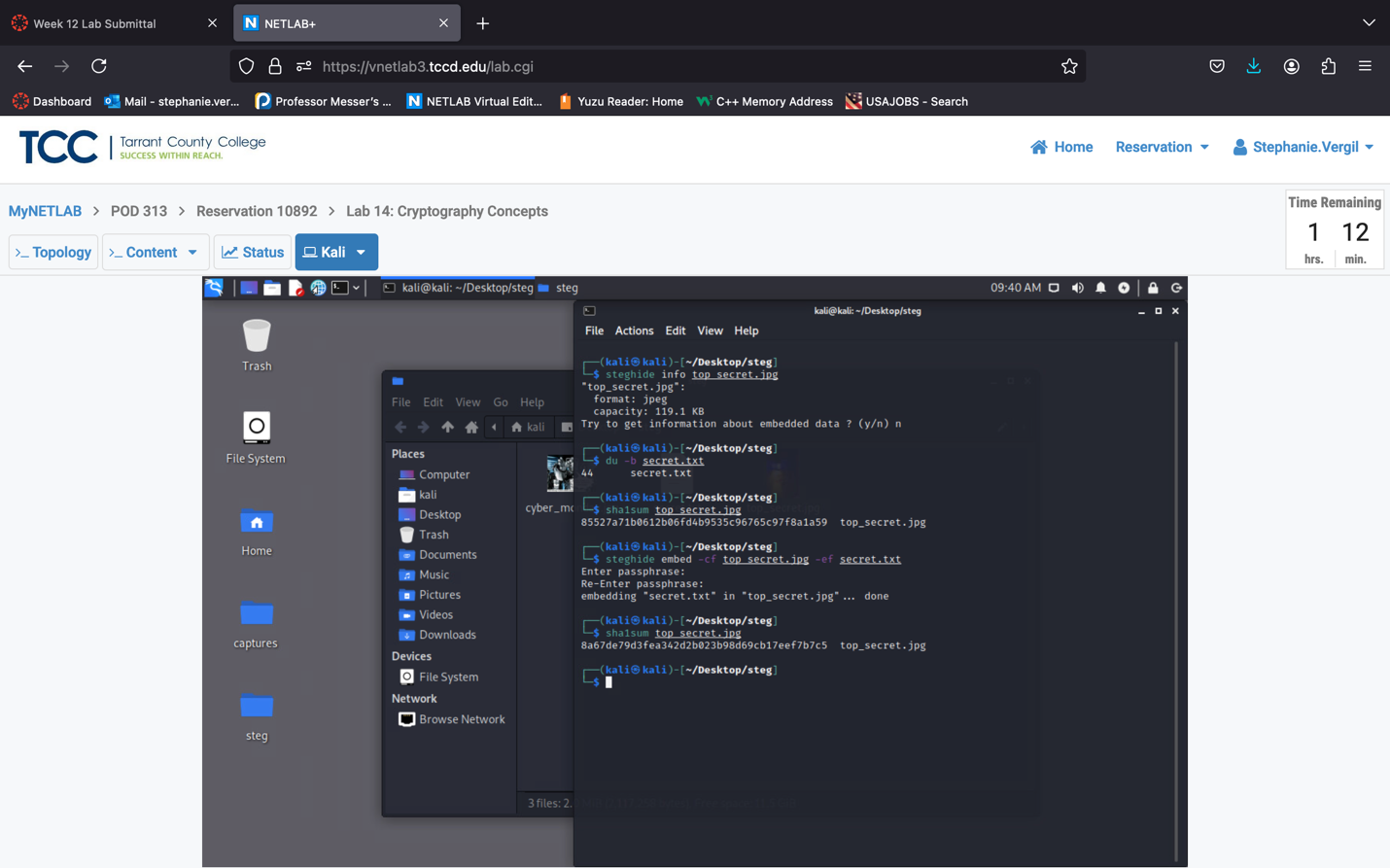


1. Enter the following command to verify the hash value once again with the same top\_secret.jpg image file:

sha1sum top\_secret.jpg

Note: Notice how the integrity has been lost in the steganography process due to

a different hash value.

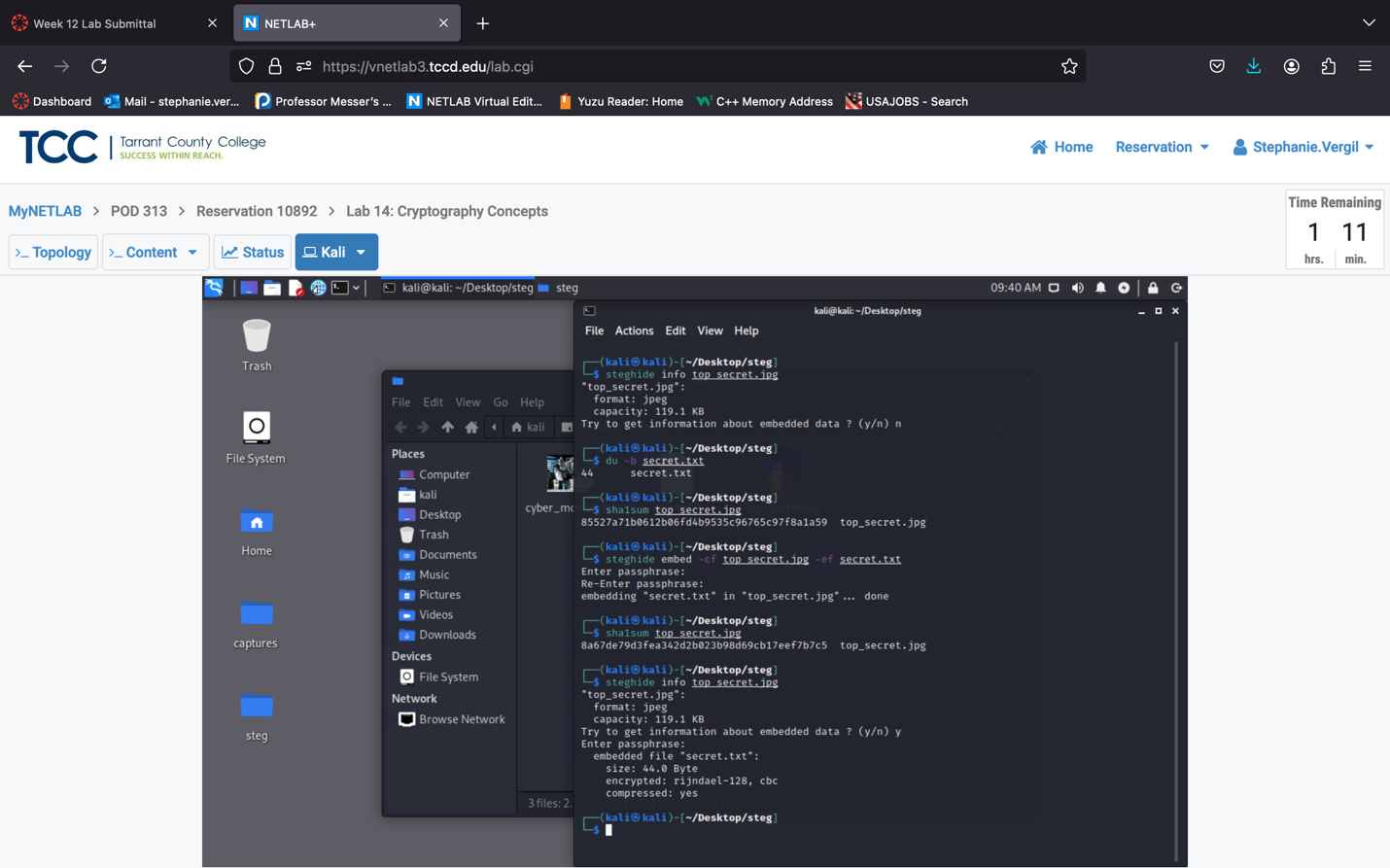


1. Enter the following command to gather information on the embedded data within the top\_secrete.jpg file. When asked to get information about embedded data, type y and enter the passphrase and press enter:

steghide info top\_secret.jpg

Note: Notice the output, highlighting that there is a secret.txt file present

within the top\_secret.jpg file.



1. Enter the following commands to delete the secret.txt file just created. Type rm secret.txt and press enter. Then run ls to see that the secret.txt file was gone

A screenshot of a computer

Description automatically generated

1. Enter the following command to extract the secret data within the top\_secret.jpg image file. Enter the passphrase when prompt:

steghide extract -sf top\_secret.jpg

A screenshot of a computer

Description automatically generated

1. Enter the following command to examine the content of the extracted file:

cat secret.txt

A screenshot of a computer

Description automatically generated

1. Leave terminal open.

2 Hiding Multiple Files Within an Image File

2.1 Using Basic Linux Commands to Hide Zipped Archives

1. While still on the Kali system, in the Terminal window, verify that you are still in the

~/Desktop/steg directory. List the files in the current directory using the ls note of the file sizes for both cyber\_monday.jpg and secret.txt.

A screenshot of a computer

Description automatically generated

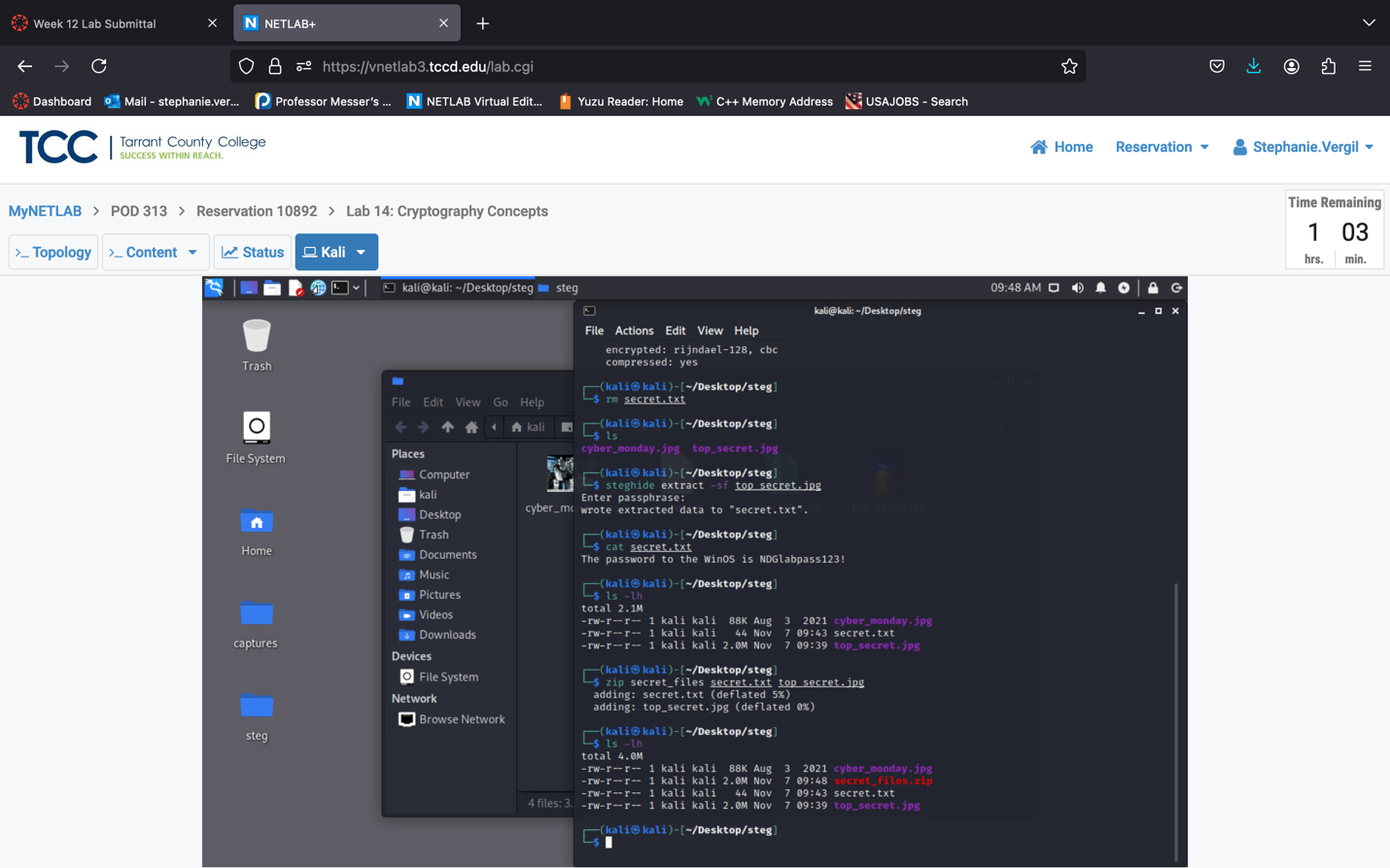
1. To create a zipped archive named secret\_files to include the files: secret.txt and dc.jpg enter the following command:

zip secret\_files secret.txt top\_secret.jpg

A screenshot of a computer

Description automatically generated

1. Enter the following command to list the current files in the directory and verify that the secret\_files.zip is present.



1. Enter the following command to hide the zipped archive within the image file called cyber\_monday.jpg. This will enable the cat command to concatenate the image and zip file together in a new file named cybersec.jpg:

cat cyber\_monday.jpg secret\_files.zip > cybersec.jpg

1. Enter the following command to list the files in the current directory and verify that the cybersec.jpg image file has been successfully created:

ls -lh

Take note of the file size and compare it to the original cyber\_monday.jpg

A screenshot of a computer

Description automatically generated

3 Observe the Avalanche Effect in Hashing Operation

3.1 Observe the Avalanche Effect

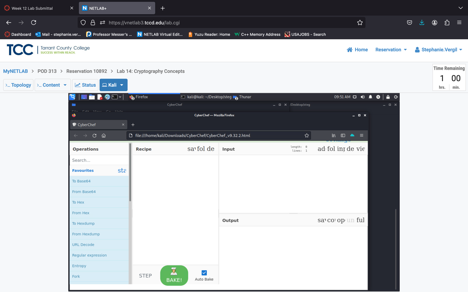
The avalanche effect in cryptography means that if one bit in the plaintext changes, at least half of the bits in the encrypted text should flip. This effect is important because it shows strong randomization in an encryption algorithm, making it more secure. If an algorithm doesn't have a strong avalanche effect, its randomization is weaker, and it's less secure. In this section, we’ll use a tool called CyberChef to see how the MD5 hash function behaves when we change bits.

1. First, let’s go to the Downloads folder by clicking the Folder icon, then Downloads, then open folder.
2. A window will open, showing a folder named CyberChef. Double-click to open the folder.

A screenshot of a computer

Description automatically generated

1. Inside the CyberChef folder, there is a CyberChef\_v9.32.2.html file. Double-click to launch it in the browser.
2. A browser window will open. You will now see the CyberChef main page.



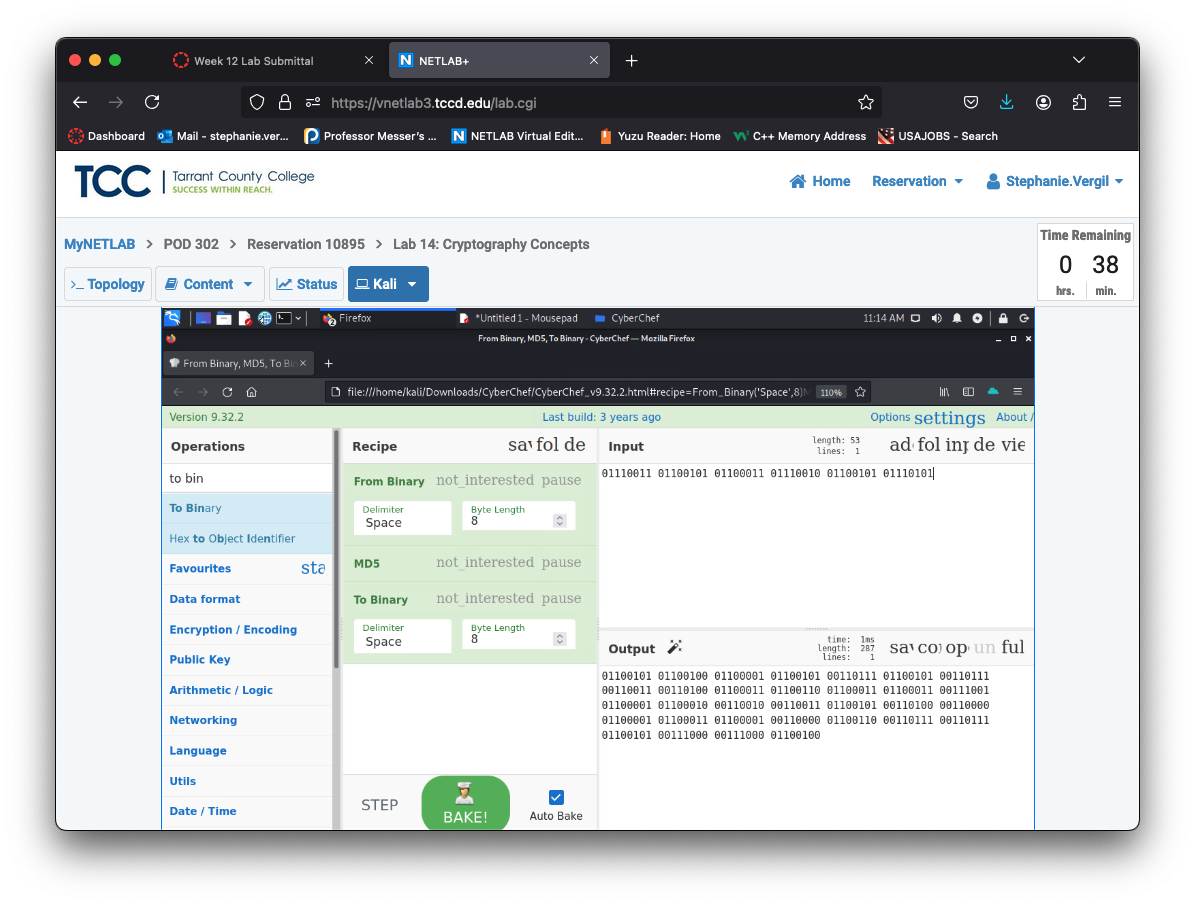
1. Now let’s first observe the MD5 hash function. First, type from Binary in the search field.
2. In the search result list, double-click on From Binary to add it to the Recipe. We will leave the default setting.
3. After adding the From Binary recipe(function), let’s delete everything in the search field and type MD5. Same as before, double-click to add the MD5 recipe(function).
4. Once again, go to the search field, type To Binary, double-click to add to the Recipe. Leave the default setting.
5. To the right side, we will now enter the binary form of the word secret; make sure you include the spaces:

01110011 01100101 01100011 01110010 01100101 01110100

A screenshot of a computer

Description automatically generated

1. As you type the binary in, the Output hash changes. Let’s start a Text Editor. Click the Applications and select to open the Text Editor.
2. Go back to the browser window. Copy the output and paste it to the Text Editor.
3. With the output stored, let’s go back to the browser window. We are going to flip one bit in the input and observe the bit changes in the output.
4. In the input field, change the rightmost bit from 0 to 1. Leave everything else untouched.



1. To compare the bit flips in the output, let’s first copy the output. Then switch to the Text Editor window. Press Enter to start a new line. Then paste the copied output.
2. To make the comparison easier, in the menu, click Document, then uncheck Word Wrap.

A screenshot of a computer

Description automatically generated

1. The two lines of binary numbers are now aligned. The top is the original output; the bottom is the one-bit flipped output. Drag the horizontal scroll bar to observe the result and count the bit flips.

A screenshot of a computer

Description automatically generated

1. The same process can also be used to test the SHA1 hash function.
2. Lab is now completed.