

Incident Response and Steganography

Written by Tyler Weiss

Exercise 1

Define and explain the following elements related to incident response.

Task 1 - Incident Response Steps

List all the steps in a numbered list, and then separately explain each step in your own words.

1. Preparation: Practice, training, planning, document preparation, coordinating in anticipation or preparation for an incident. The preparation is all central to responding to a security breach. Preparation would involve conducting risk assessments, evaluating potential vulnerabilities, establishing appropriate communication channels, and ensuring that business continuity plans are in place. Tool testing, parallel testing, and tabletop exercises are performed during this step. Regular testing and evaluation of the incident response plan ensures that it stays current and viable. This ensures that weakness in the plan are identified and improved ultimately improving the overall security posture.
2. Detection: Involves detecting and verifying a breach. During this step an organization must decide if an event is actually an attack. Intrusion prevention and detection systems play a key role in this step. A threshold IoC/IoA must be reached in order to conclude that an incident has taken place.
3. Analysis: The extent of the attack and classification of the severity and mechanism has to be defined. The impact scope of impact has to be defined as well.
4. Containment: Once identified you need to contain the effected environment. This would involve taking infected systems offline or isolation in order to contain the threat and prevent the further spread of malware and attacker movement. The quicker this

is done the more likely further damage is prevented. Do not delete evidence during this step.

5. **Eradication:** Investigation of the root cause and removing threats from the system. After all malware has been removed and there are no more traces of the threat, the systems need to be returned to their original configuration. Insight into protecting future threats and IoC/IoA can be gleaned from this step.
6. **Recovery:** The system must be returned to its original state. Reintegrating the previously infected system and returning the network to normal as fast as possible are crucial to reducing impact to the company. Data may have been recovered or rebuilt at this point as well.
7. **Lessons Learned:** After the incident is over a post-mortem must be gathered together to present their findings and issues during the process. What was done well and what areas need improvement are questions asked during this step. Does policy need to be changed in light of this new information. The information here will help dictate how incidents are handled in the future.

Task 2 - Incident Response Phases

List all the phases in a numbered list, and then separately explain phase step in your own words.

1. **Preparation:** Preparation and planning for an incident. Collect information about your system and vulnerabilities taking action on your findings to reduce risk. This phase is constant, performing table top exercises and tools testing, to be the most prepared for an incident at any moment. Information from the post-incident phase is fed into the preparation phase for constant development. If an incident does occur you go directly into the Detection and Analysis phase.
2. **Detection and Analysis:** Identification of suspicious activity and verification that the activity is truly an attack. Once identified, collection of data becomes primary so that it can be analyzed. The analysis must attempt to discover how severe the attack was and which parts of the system have been effected. Once the effected systems are identified and the malware pinpointed you move to containment, eradication and Recovery.
3. **Containment, Eradication, and Recovery:** The previous phase should be as quick as possible in order to prevent further spread of the infected surface. The effected systems should be isolated and the malware eradicated. While removing the malware more indicators trigger hidden IoCs or malware to surface. In this case

The information is fed back to the detection and Analysis phase because more infected systems may be discovered armed with this new information. Once the root cause is discovered and removed the systems need to be brought back online. The original configurations need to be restored and data must be recovered or rebuilt. Once done, move to the post-incident activity phase.

4. Post-Incident Activity: Review of incident response process needs to be conducted. The incident response team and others involved are pulled together for an after action report. What went right or wrong? Improvements that need to be made and lessons learned. Did anything impede the process? Can information from this incident and the response be applied to prevent future attacks or to reduce recovery time? The information from the after action report is then implemented by passing it off to the preparation phase.

Task 3 - Incident Response Policy

Explain what the purpose of an incident policy is in your own words.

An incident response policy specifies actions to be taken under if certain conditions are met. It establishes procedures necessary in dealing with a threat or breach. The more fine tuned this policy is the quicker the response time may be, which in turn will help reduce the downtime and minimize damage.

Task 4 - Incident Response Plan

Explain what an IRP is, and what purpose does it play in your own words.

The incident response plan is similar to incident response policy. It is a written document that details how to deal with a security incident once it occurs. The goal is to reduce the overall impact, contain the breach, remediate the cause, return the system back to normal functionality, and reduce future risk.

Task 5 - Communication Plan

Explain what a communication plan is, and what purpose does it play in your own words.

A communication plan outlines who to contact and what information is needed to be presented during response to an incident. Incident response requires careful coordination between response teams, management, authorities, stakeholders, and spokespeople. This insures the information presented is consistent and true without damaging the company's reputation or impeding an investigation.

Task 6 - Recon

Explain what an enumeration is, and what purpose does it play in your own words.

Enumeration is the act of establishing connection in order to gather information of network assets. This information could consist of network address, protocol specifics, web footprint, firewall detection, account credentials, and so on. It can be used to find weak or unknown parts of a network.

Task 7 - Exfiltration

Explain what tunneling is, and what purpose does it play in your own words.

Tunneling is the act of creating a secure connection between two points, across multiple networks acting as if they were directly connected, for the purpose of data transmission. To facilitate this secure connection the data packet is wrapped inside and encrypted within a second packet. Traffic is essentially isolated from exterior networks during transport. Tunneling is used to bypass firewalls, create VPN connections, and to enable the use of unsupported network protocols.

Task 8 - Communication

Explain what a pem file is, and what purpose does it play in your own words.

A Privacy Enhanced Mail or PEM file is a datafile containing cryptographic information, such as keys or certificates, that is stored in a standardized format. PEM files are part of Public Key Infrastructure or PKI. They are normally used for secure communication such as TLS/SSL, identification and verification of users and assets, data encryption, and code signing.

Task 9 - The Dominican Republic Incident

1. Listen to this podcast on Incident Response:
 - Darknet Diaries, Episode 135.
2. Explain what your take aways are from this podcast.

There are few major takeaways here. Firstly communication is key. Growing and nurturing contacts within the cyber field can give you an edge. These contacts can be called upon to help and will share threat information or response techniques that could be critical in the defense of your own systems. This is especially true if you are a smaller player and don't have the same resources as your competitors.

Omar talks about switching his focus from building defense to improving detection. I think this highlights the fact that an adversary may be able to penetrate your network, no matter how well you think you are protected, but if you can not detect the intrusion properly you can not respond appropriately.

Dissemination of information is also necessary to make the community aware of the current threat landscape and to let others know who to ask if they detect similar issues. if the Dominican Republic Cyber Division had not contacted the other Latin countries, those countries may have not been able to fend off the attack.

Not only is communication and information sharing important between organizations and individuals, it's critical to contact and share issues with vendors and service providers so that vulnerable products can be fixed and appropriate action can be taken swiftly.

Exercise 2 - Steganography (Windows)

Task 1 - Install openstego

1. Download OpenStego from: <https://www.openstego.com/>
2. Download the Source File from Github:
<https://github.com/ajay63/BlackTowerAcademy/blob/main/maxresdefault-2003057562.jpg>




































Task 2 - Dependencies





It is likely if OpenStego fails to run, its because it requires JAVA and you don't have it installed.

One can easily and quietly install Java from <https://ninite.com/>

Often you may run into a situation where software a program is dependent on is not installed. If this is the case you need to find and install the proper software or dependencies before the program will function correctly. This is the case with OpenStego because it is missing the Java Development Kit (JDK) software dependency. Once installed OpenStego should run just fine.

1. Pick the apps you want

Runtimes	Imaging	Documents
<input checked="" type="checkbox"/>  Java (AdoptOpenJDK) x64 8	<input type="checkbox"/>  Krita	<input type="checkbox"/>  Foxit Reader
<input type="checkbox"/>  Java (AdoptOpenJDK) 8	<input type="checkbox"/>  Blender	<input type="checkbox"/>  LibreOffice
<input type="checkbox"/>  Java (AdoptOpenJDK) x64...	<input type="checkbox"/>  Paint.NET	<input type="checkbox"/>  SumatraPDF
<input type="checkbox"/>  Java (AdoptOpenJDK) x64...	<input type="checkbox"/>  GIMP	<input type="checkbox"/>  CutePDF
<input type="checkbox"/>  Java (AdoptOpenJDK) x64...	<input type="checkbox"/>  IrfanView	<input type="checkbox"/>  OpenOffice
<input type="checkbox"/>  .NET 4.8	<input type="checkbox"/>  XnView	
<input type="checkbox"/>  .NET Desktop Runtime x64 5	<input type="checkbox"/>  Inkscape	
<input type="checkbox"/>  .NET Desktop Runtime 5	<input type="checkbox"/>  FastStone	
<input type="checkbox"/>  .NET Desktop Runtime x64 6	<input type="checkbox"/>  Greenshot	
<input type="checkbox"/>  .NET Desktop Runtime 6	<input type="checkbox"/>  ShareX	
<input type="checkbox"/>  .NET Desktop Runtime x64 7		
<input type="checkbox"/>  .NET Desktop Runtime 7	Developer Tools	
<input type="checkbox"/>  .NET Desktop Runtime x64 8	<input type="checkbox"/>  Python x64 3	
<input type="checkbox"/>  .NET Desktop Runtime 8	<input type="checkbox"/>  Python 3	
	<input type="checkbox"/>  Python	
	<input type="checkbox"/>  FileZilla	
	<input type="checkbox"/>  Notepad++	
	<input type="checkbox"/>  JDK (AdoptOpenJDK) x64 8	

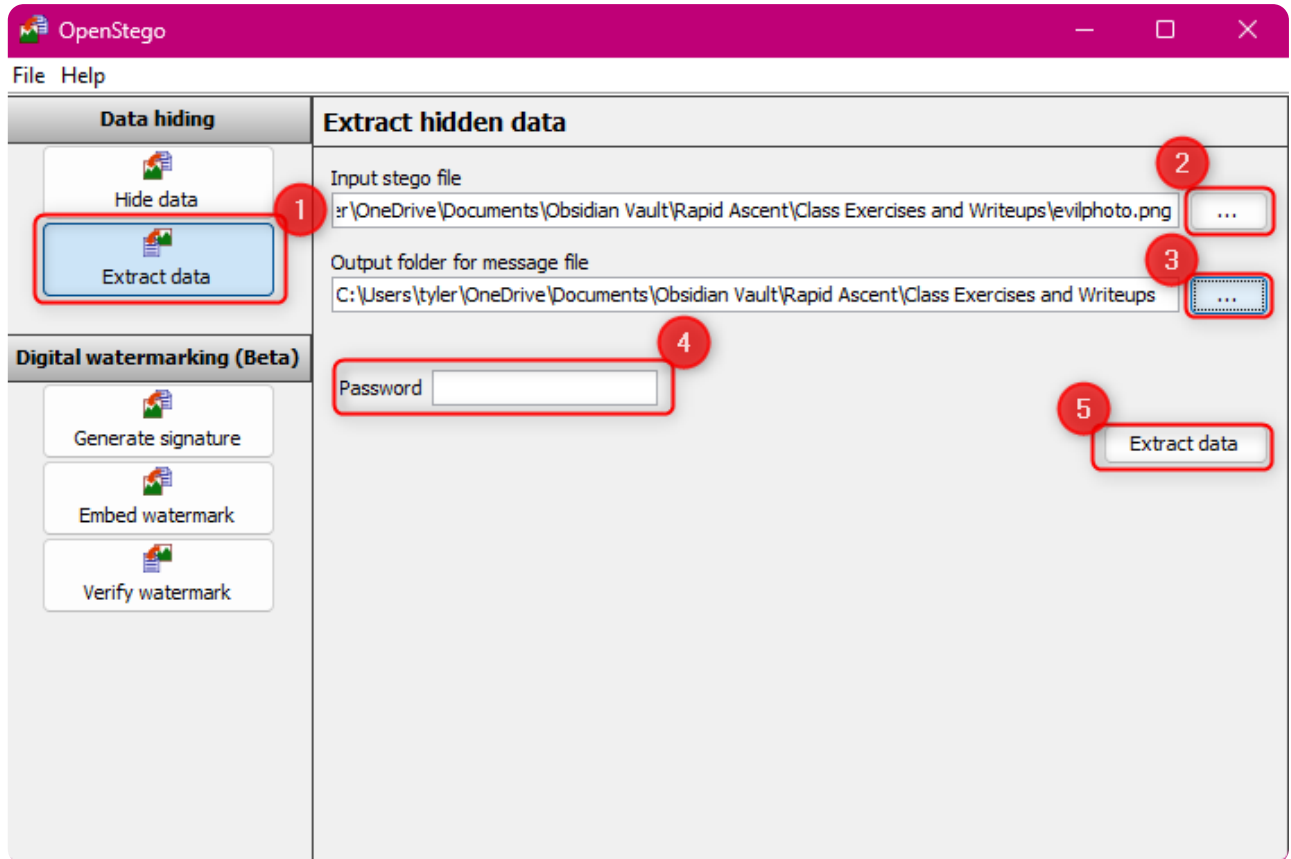
Online Storage
☐  Dropbox
☐  Google Drive for Desktop
☐  OneDrive
☐  SugarSync

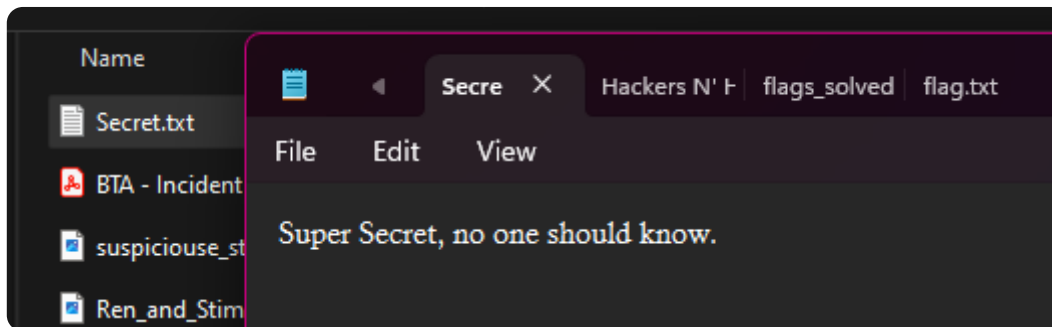
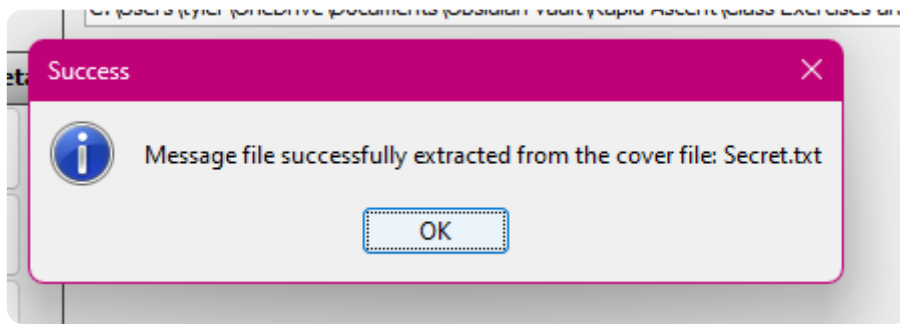
Task 3 - Data Extraction

To extract data perform the following steps.

1. Select the 'Extract Data' option on the left

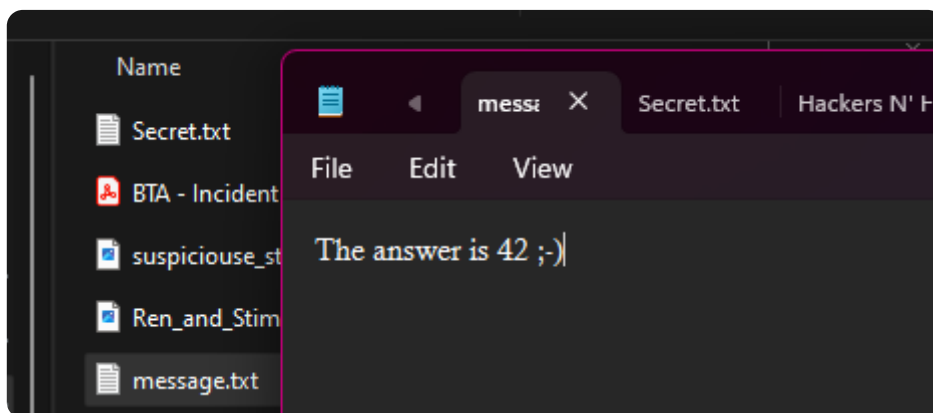
2. Click the button to the right 'Input stego file' to select the file with the embedded message.
3. Click the button to the right of 'Output folder for message file' and select the directory where the secret message file will be saved once extracted.
4. Add a password if needed. This would be the password used to encrypt the embedded message.
5. Click on EXTRACT DATA.
6. A text file should appear in the directory that was specified earlier.
7. Open the text file to read the message.

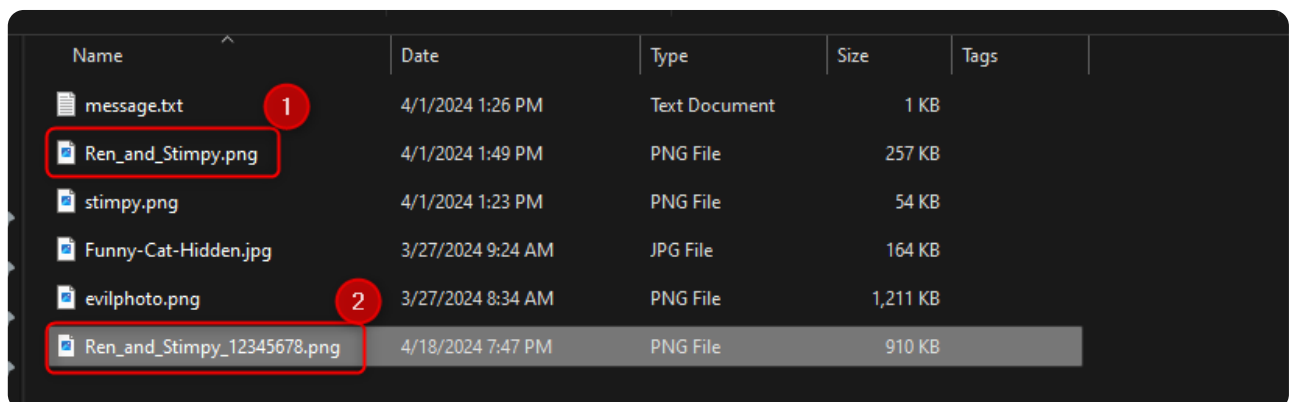
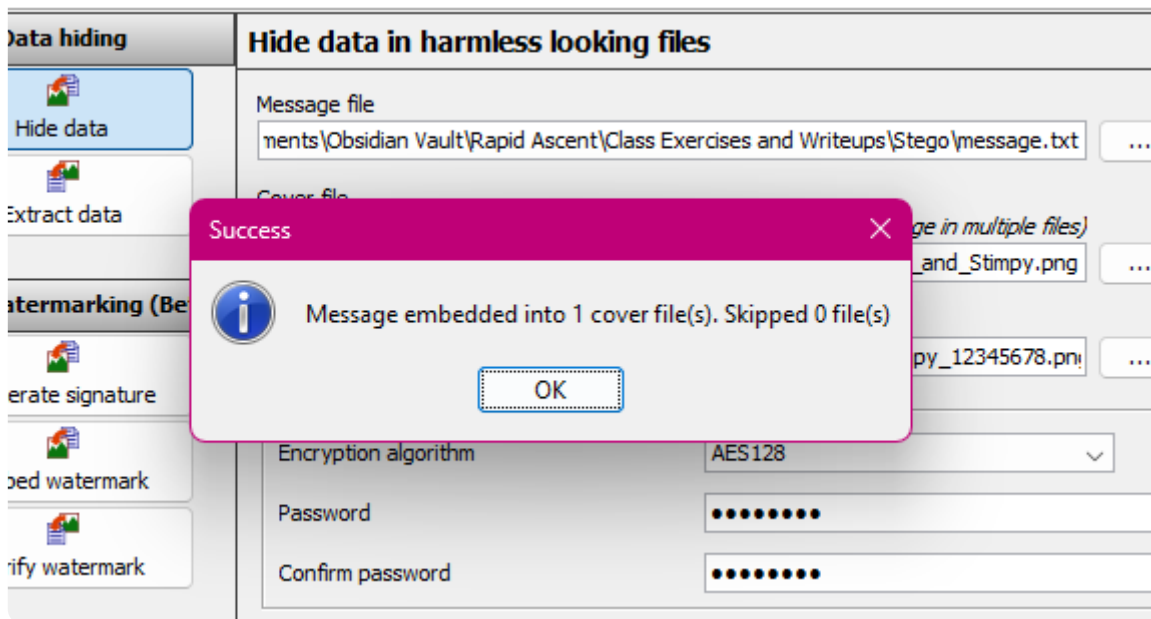
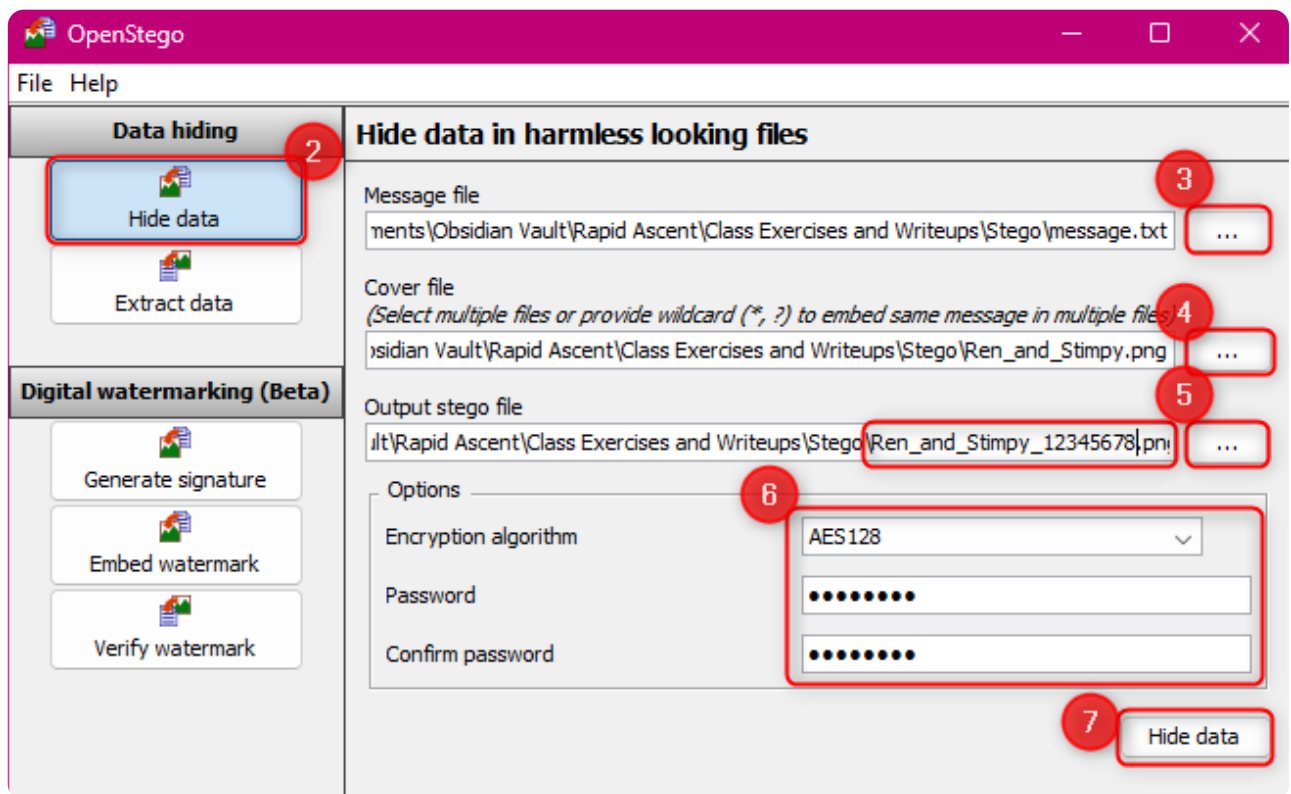




Task 4 - Implement Steganography

1. Create a text file with a hidden message. (Save the file)
2. Select the 'Hide Data' on the left.
3. Select the text file you created with the hidden message in it. (Message File)
4. Select a funny picture you already have. (Cover File)
5. Select a location and file name for your stenographic file. (Output File)
 1. Save it as a .png file.
6. Select an encryption algorithm and input a password
 1. This is not necessary but it adds a layer of security.
7. Click 'Hide Data'





8. This is the original image file.

9. After running OpenStego this is the file that was produced.

OpenStego will hide the message data within the raw data of the image file. You must use the same program to extract the image because knows how to find the hidden data and properly extract it.

Task 5 - Data Extraction

Download the image file at the following location:

<https://github.com/ajay63/BlackTowerAcademy/blob/main/evilphoto.png>

Extract the secret message and display it. Note that there is no password to input or order to complete extraction.







Refer to task 3 where we have already accomplished this task.

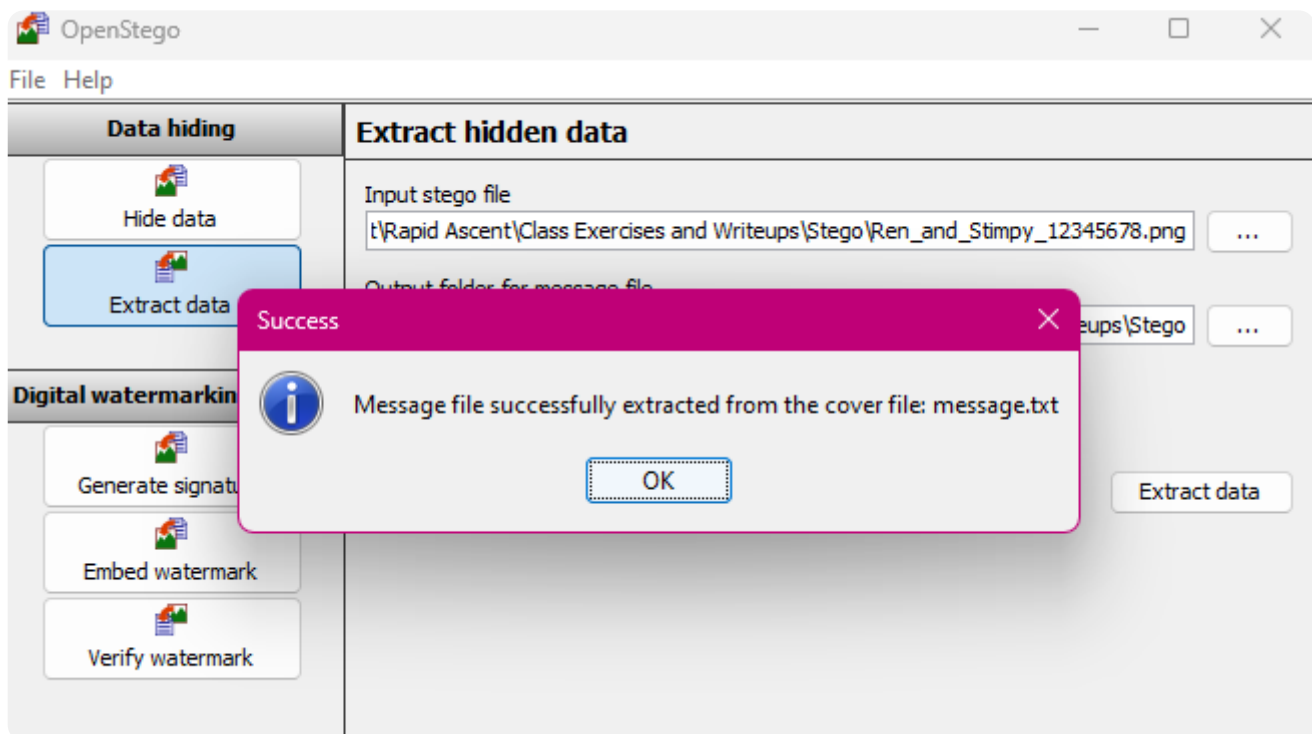
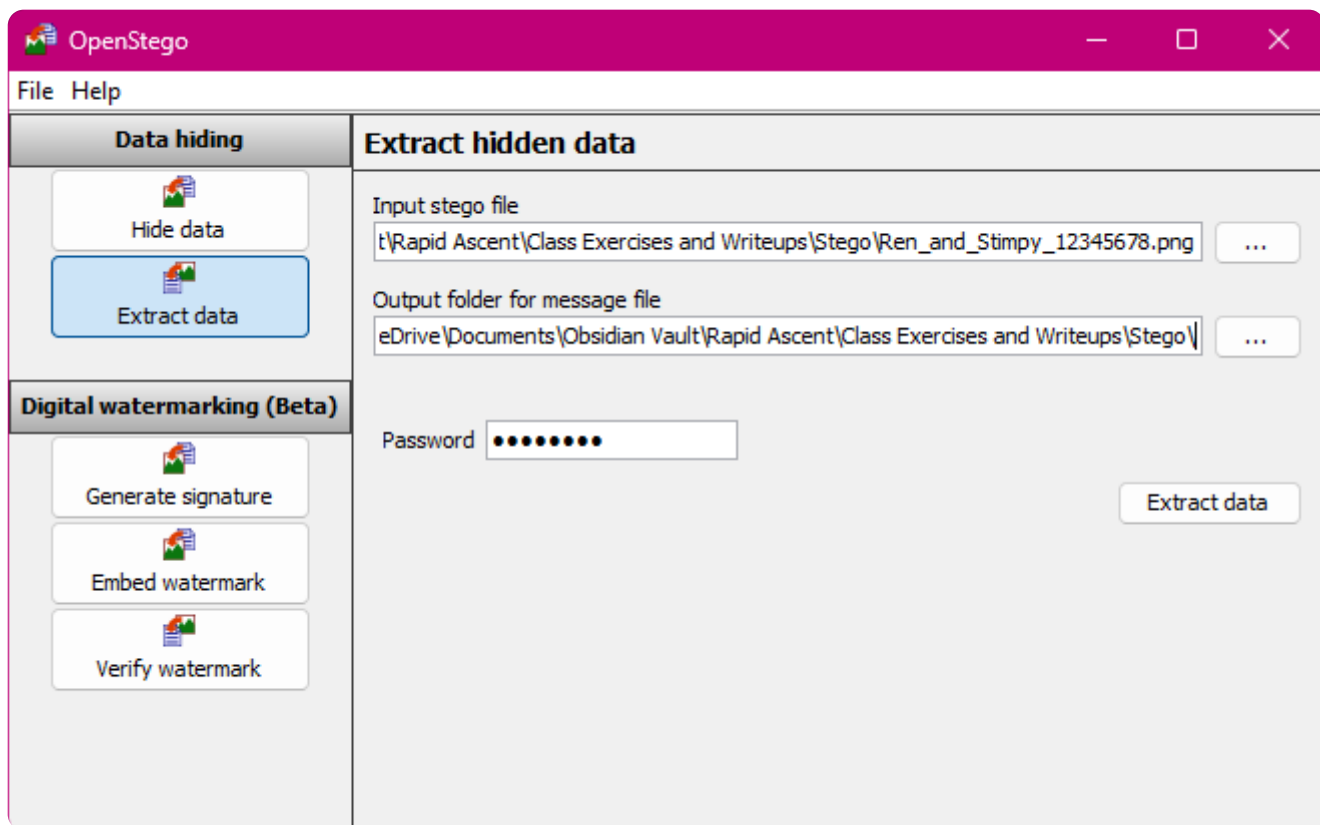
Task 6 - Classmate Data Extraction

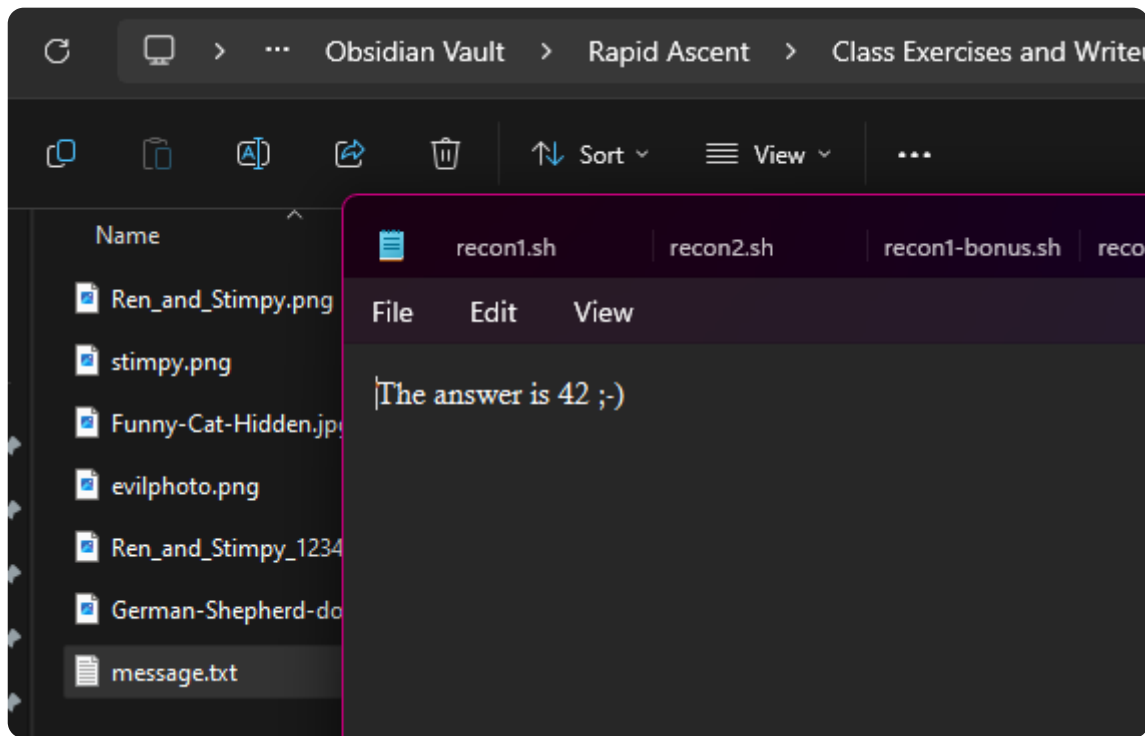
Embed a secret in an image, then share the file with a classmate, and have him read the secret message back to you.

The screen shots in task 4 show the steps I took to embed a message in an image.

Below is the process of extracting a message from an image.

Name	Date	Type	Size	Tags
 Ren_and_Stimpy.png	4/1/2024 1:49 PM	PNG File	257 KB	
 stimpy.png	4/1/2024 1:23 PM	PNG File	54 KB	
 Funny-Cat-Hidden.jpg	3/27/2024 9:24 AM	JPG File	164 KB	
 evilphoto.png	3/27/2024 8:34 AM	PNG File	1,211 KB	
 Ren_and_Stimpy_12345678.png	4/18/2024 7:47 PM	PNG File	910 KB	
 German-Shepherd-dog-Alsatian.j...	4/18/2024 9:28 PM	JPG File	340 KB	





Exercise 3 - Steganography (Linux)

Install steghide and use it to hide and extract a message from an image file.

Task 1 – Installation

To install steg hide use 'apt'. run the following command on your linux server.

```
sudo apt install steghide -y
```

Task 2 – Download and extract the secret

Download the following image:

<https://github.com/ajay63/BlackTowerAcademy/blob/main/Funny-Cat-Hidden.jpg>

The image file needs to be downloaded to your host machine. using wget will not work. Use scp to copy the image to your Linux server. Below is an example of the scp

command.

```
rec0nrat@demosever1:~/Stego$  
logout  
Connection to 192.168.40.80 closed.  
PS C:\Users\tyler> scp '.\OneDrive\Documents\Obsidian Vault\Rapid Ascent\Class Exercises and Writeups\Stego\Funny-Cat  
-Hidden.jpg' rec0nrat@192.168.40.80:/home/rec0nrat/Stego/  
(rec0nrat@192.168.40.80) Password:  
Funny-Cat-Hidden.jpg 100% 163KB 40.9MB/s 00:00  
PS C:\Users\tyler>
```

Extract the message from the image using the following command.

```
rec0nrat@demosever1:~/Stego$ ls  
Funny-Cat-Hidden.jpg  
rec0nrat@demosever1:~/Stego$ steghide --extract -sf Funny-Cat-Hidden.jpg -xf secret.txt -p hello  
wrote extracted data to "secret.txt".  
rec0nrat@demosever1:~/Stego$ ls  
Funny-Cat-Hidden.jpg secret.txt  
rec0nrat@demosever1:~/Stego$ cat secret.txt  
I love donuts  
rec0nrat@demosever1:~/Stego$
```

Task 3 - Embed a secret into an image

Using steghide embed a secret into a file & share it with a classmate for them to decode

Embedding a message in an image 'stimpy.jpg'

Use scp to copy the image to the Linux server. Create a message using vim and store it in 'message.txt'. Use steghide to imbed message using 'loki97' for the encryption algorithm and '123456' as the password. The stego file produced is 'stimpy_123456.jpg'.

```
PS C:\Users\tyler> scp '.\OneDrive\Documents\Obsidian Vault\Rapid Ascent\Class Exercises and Writeups\Stego\stimpy.png'  
g' rec0nrat@192.168.40.80:/home/rec0nrat/Stego/  
(rec0nrat@192.168.40.80) Password:  
stimpy.png 100% 53KB 4.4MB/s 00:00  
PS C:\Users\tyler>
```

```
rec0nrat@demosever1:~/Stego$ vim message.txt  
rec0nrat@demosever1:~/Stego$ ls  
Funny-Cat-Hidden.jpg German-Shepherd-dog-Alsatian.jpg message.txt secret2.txt secret.txt stimpy.png  
rec0nrat@demosever1:~/Stego$
```

```
rec0nrat@demosever1: ~/Stego  
I'm Naruto Uzumaki. Believe It!!!  
~  
~
```

```
rec0nrat@demosever1:~/Stego$ steghide --embed -ef message.txt -cf stimpy.jpg -sf stimpy_123456.jpg -p 123456 -e loki97  
embedding "message.txt" in "stimpy.jpg"... done  
writing stego file "stimpy_123456.jpg"... done  
rec0nrat@demosever1:~/Stego$ ls  
Funny-Cat-Hidden.jpg message.txt secret.txt stimpy.jpg  
German-Shepherd-dog-Alsatian.jpg secret2.txt stimpy_123456.jpg  
rec0nrat@demosever1:~/Stego$
```

Extracting shared Image German-Shepherd-dog-Atlassian.jpg

The below screen shots show the process of using scp to transfer a shared image to my

Linux server from my host.

```
PS C:\Users\tyler> scp '.\OneDrive\Documents\Obsidian Vault\Rapid Ascent\Class Exercises and Writeups\Stego\German-Shepherd-dog-Alsatian.jpg' rec0nrat@192.168.40.80:/home/rec0nrat/Stego/
(rec0nrat@192.168.40.80) Password:
German-Shepherd-dog-Alsatian.jpg                               100% 339KB 85.7MB/s 00:00
PS C:\Users\tyler> ssh rec0nrat@192.168.40.80
(rec0nrat@192.168.40.80) Password:
(rec0nrat@192.168.40.80) Password:
```

Once logged into the server extracted the message from the shared image file using steghide with flags '-sf' for the image and '-xf' for the extracted message file. there was no password for decryption.

```
rec0nrat@demosever1:~/Stego$ ls
Funny-Cat-Hidden.jpg  German-Shepherd-dog-Alsatian.jpg  secret.txt
rec0nrat@demosever1:~/Stego$ steghide --extract -sf German-Shepherd-dog-Alsatian.jpg -xf secret2.txt
Enter passphrase:
wrote extracted data to "secret2.txt".
rec0nrat@demosever1:~/Stego$ ls
Funny-Cat-Hidden.jpg  German-Shepherd-dog-Alsatian.jpg  secret2.txt  secret.txt
rec0nrat@demosever1:~/Stego$ cat secret2.txt
This is my Secret Message
rec0nrat@demosever1:~/Stego$
```