Incident Response Intro Exercise

7 Steps of Incident Response

1. Preparation
2. Detection
3. Analysis
4. Containment
5. Eradication
6. Recovery
7. Lessons Learned

The following 7 step incident response plan we will be covering is the more traditional and comprehensive approach.

Incident response plans are important and help an organization before, during, and after a security incident or potential incident occurs.

CYBERSECURITY INCIDENT RESPONSE PLAN

Preparation

- Essential and all organizations should be prepared for the worst
- Establish a notification process
- Create a checklist
- Is your disaster recovery plan up to date
- Proper training for members and teams

Detection

- Data collection
  - Systems (logs, error messages)
  - Security Tools (IDS, IPS, firewalls)
- Precursors
  - Signs of potential incident in future
- Indicators
  - Signs that an attack happened or is happening

Analysis

- Correlate related events for deviations
  - Compare to baseline / normal activity
- What data, if any, has been compromised
- Who breached
  - Insider? APT?
- How did they breach
  - Check the attack surface

Containment

- Limit scope and magnitude
- Prevent further spread
- Disconnect from network
    - Possibly shut down completely
    - Quarantine infected systems
    - If unable, monitor thoroughly

Eradication

- Eliminate any malicious code
- Reset passwords
- Remove accounts that may have been breached

Recover

- Rebuild OS or replacing system drive
- Backups
- Restoring operations
- Validate as systems come back online

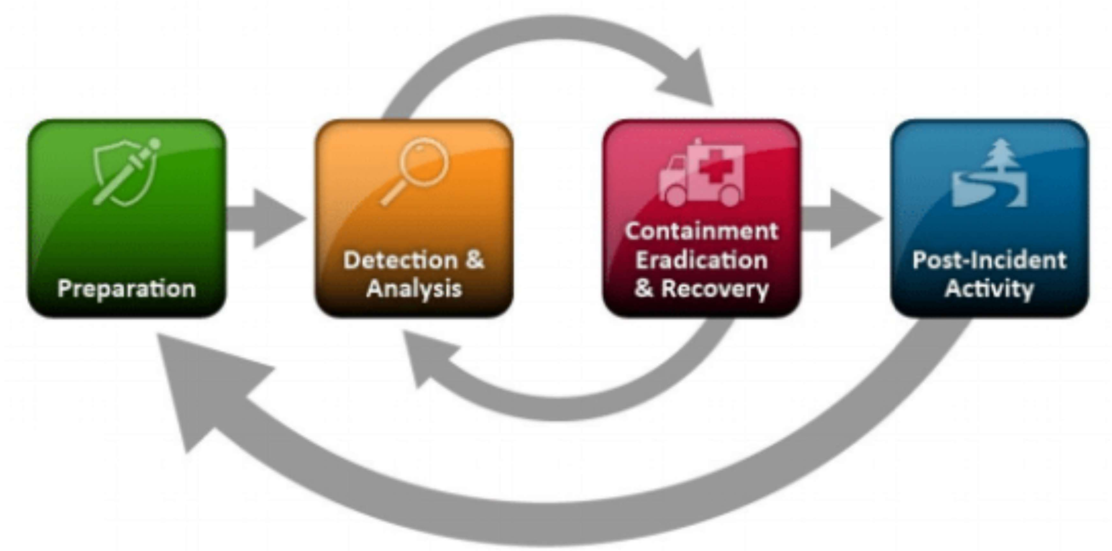
Lessons Learned

- Timelines
- Communication effectiveness
- What worked and what didn't
  - How to improve what didn't
- Lacking tools or resources

4 Stages of Incident Response

1. Preparation
2. Detection & Analysis
3. Containment, Eradication, Recovery
4. Post-Incident Activity

Next we will go over the 4 stages of incident response. Very similar to the 7 step process in many ways. However, this is a cyclical process with continued learning and discovery on how to best protect ones organization.



Preparation

- Essential and all organizations should be prepared for the worst
- Establish a notification process
- Create a checklist
- Is your disaster recovery plan up to date
- Proper training for members and teams

Detection & Analysis

- Data collection
  - Systems (logs, error messages)
  - Security Tools (IDS, IPS, firewalls)
- Precursors / Indicators
  - Signs of potential incident in future
  - Signs that an attack happened or is happening
- Analyze nature of attack
- Correlate related events for deviations
- What data, if any, has been compromised

Containment, Eradication, Recovery

- Limit scope and magnitude
- Prevent further spread
- Disconnect from network
  - Possibly shut down completely
  - Quarantine infected systems
  - If unable, monitor thoroughly
- Eliminate any malicious code

- Reset passwords
- Remove accounts that may have been breached
- Rebuild OS or replacing system drive
- Backups
- Restoring operations
- Validate as systems come back online

Post-Incident Activity

- Timelines
- Communication effectiveness
- What worked and what didn’t
- How to improve what didn’t
- Lacking tools or resources

Incident Response Policy

Knowing your organizations incident response policy is paramount to success. The policy should inform one on **what** to do. It should achieve this through clearly defined roles and responsibilities for before, during, and after an incident. The purpose being to minimize the impact, threat containment, and returning to normal as quickly as possible.

Additionally, it is safe to assume that failure to comply with the organizations policy will likely result in severe consequences to include termination.

Incident Response Plan (IRP)

Where a policy tells one what to do, the Incident Response Plan tells us **how** to do ones role. The IRP should be as comprehensive as the policy, if not more so, to ensure the organization can quickly contain and recover from an incident.

Communication Plan

Having a well thought out communication plan can play an important factor in incident response. You will want streamlined communication to enable improved collaboration, decision making, and ultimately leading to a faster resolution and recovery.

Take advantage or the various communication channels such as email, instant messaging, and dedicated incident response platforms to facilitate quick dissemination of information among members, leaders, stakeholders, and more.

Templates help teams communicate clear and consistent messages and increase the speed in which they communicate. In the military we had numerous templates to help us communicate like a SALUTE report or Nine line medevac card. During critical times when every second counts templates can be game changing, especially for anything time sensitive.

You will also want to ensure escalation paths are known and taken into account. The right people must be notified promptly to take appropriate action swiftly.

Communication plans also help minimize confusion and manage the overall impact of the incident.

Recon

Before a burgler robs a bank they preform recon to come up with a game plan. Cyber attackers are no different and one way they preform recon is through enumeration.
Enumeration is where you gather information/data about the network, IP addresses, protocols, and more. Thinking about the tools we have learned recently can you think of any that may do just this?

If you said NMAP then you are correct. It does an excellent job for network mapping and finding potential vulnerabilities in your system/network architecture.

Exfiltration

Sticking with our burgler analogy sometimes to get the “gold” they may tunnel in to try and not be seen. Once again, cyber threat actors will look do something similar and tunnel in. There are numerous ways where they may tunnel in OR out to hide their presence. Have you ever “tunneled” into a system? When you SSH into a VM that is a form of tunneling. There are also VPNs that use tunneling as well as DNS tunneling.

Also, remember the threat actor does not want to be caught. Tunneling through DNS or other ways hides the data in plain sight. Using various methods they navigate to restricted or confidential data and then embed that data as they send it back out. Many times doing this makes it very hard to detect and has a high chance of leaving without triggering the firewall, IDS, or other system.

Communication

PEM – Privacy Enhanced Mail is a type of Public Key Infrastructure file which was initially used for making e-mail secure. Being text based it tends to be less prone to translation/transmission errors and can have a variety of extensions.

Though depending you may be use to seeing PKCS12, pfx, or p12 and this is that would be due to the enhanced security it offers. This contains private keys but can also be freely converted to PEM through use of openssl.

So why bring this up? Well, as we talk about the threat we need to think of additional ways to protect our data. This can be done through encryption which PEM is just one way to do this.

Dominican Republic Incident

Recently we listened to a podcast about an incredible cybersecurity incident. I highly encourage all to listen and research it. https://darknetdiaries.com/episode/135/

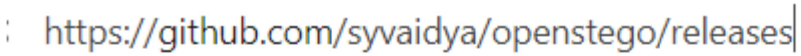
The incident obviously was in the Dominican Republic and it is incredible to hear from Omar Avilez, one of their national CSIRT members and how his team dealt with an incredible cybersecurity incident. There were multiple high level cyber organizations, persistence, zero day attack, remote access malware, Cobalt Strike (hacker suite), and a lot more.

Through all of the podcast and my reading, there is one thing that really stood out to me, and that is how in tune Omar was with other incidents happening around the world. He researched the other incidents, went to where they occurred, worked with multiple teams, and attended conferences. Cybersecurity is not a one person job and takes teammates and a network of the “good guys” to combat the threat.

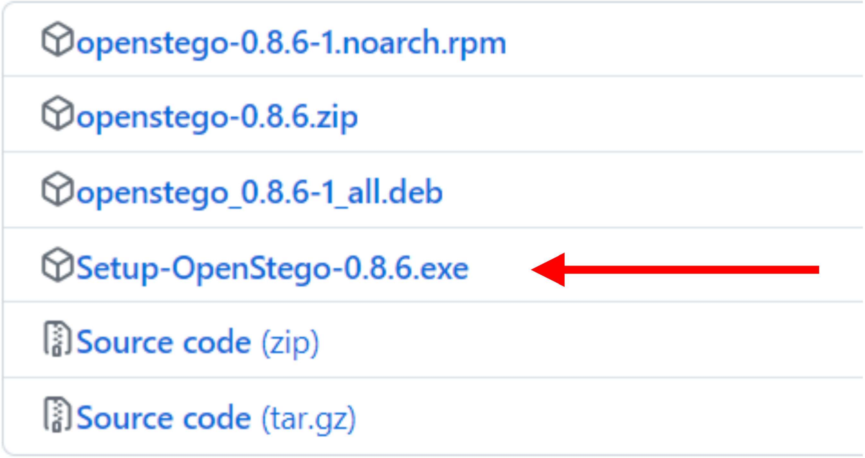
There are many learning points and key take-a-ways to be gleaned from this. However, what will stick with me the most is the importance of paying attention to cybersecurity incidents happening around the world. You never know if someone else may have suffered a similar attack and what you can learn from them. Two teams across the world may see different pieces and with a little collaboration you may be able to shut a cyber criminal organization down.

Ok, lets shift gears a bit and work on hiding in plain sight…

To do this we will download openstego first


https://github.com/syvaidya/openstego/releases

Then, working in windows just grab the .exe file


openstego-0.8.6-1.noarch.rpm

openstego-0.8.6.zip

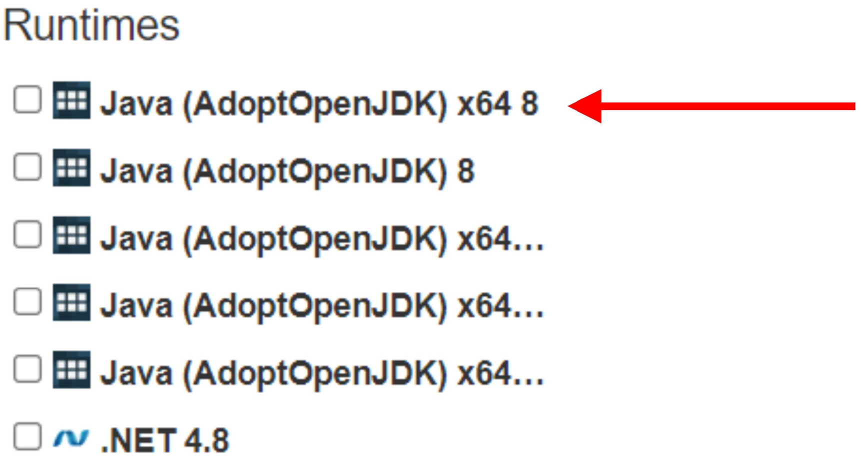
openstego_0.8.6-1_all.deb

Setup-OpenStego-0.8.6.exe

Source code (zip)

Source code (tar.gz)

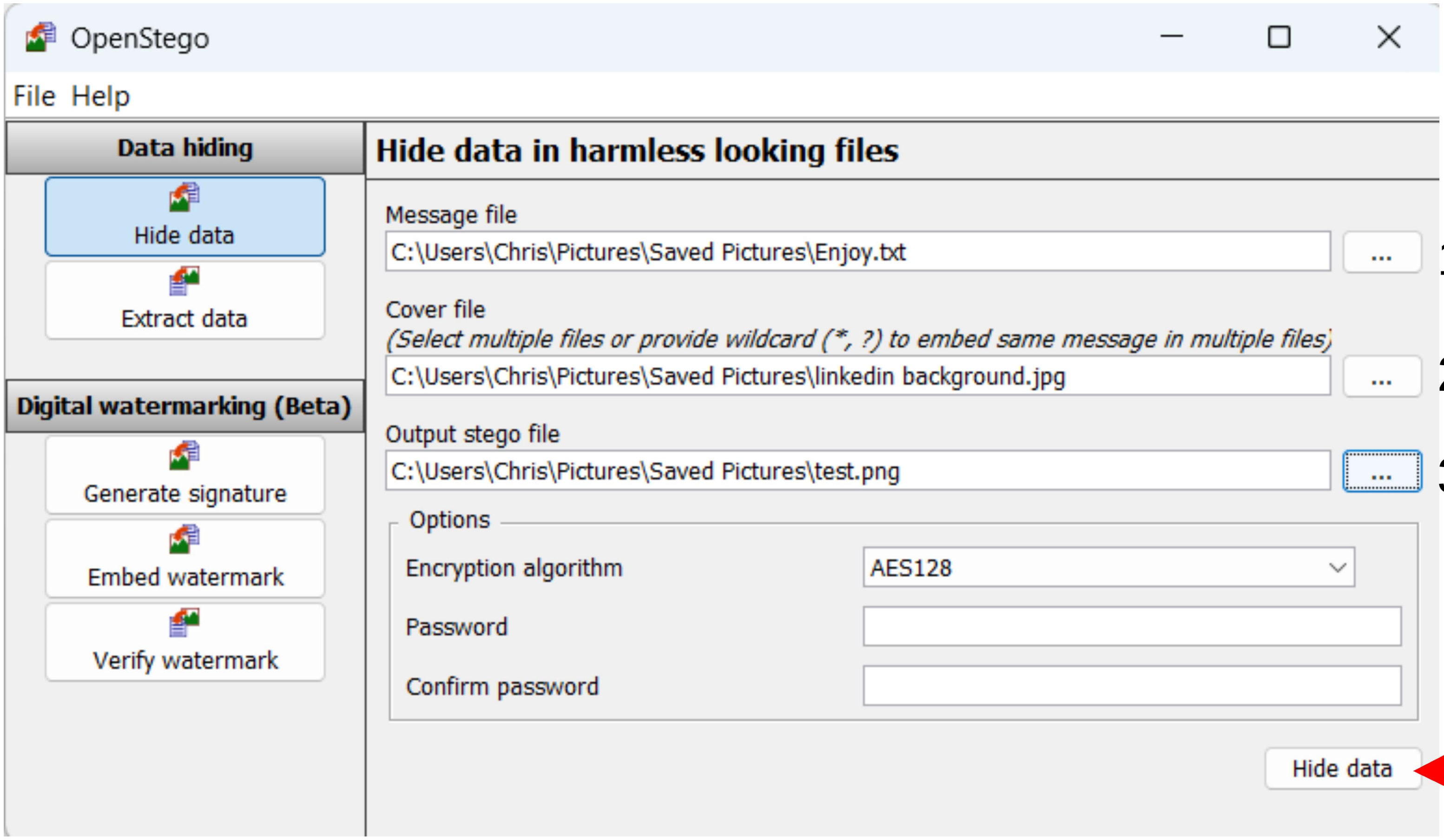
Once installed you may still not be able to run the program. If you are like me then you are missing a dependency. No worries, this just means we need to download it. I would suggest heading over to Ninite and grabbing Java runtime.



Runtimes

- □ Java (AdoptOpenJDK) x64 8
- □ Java (AdoptOpenJDK) 8
- □ Java (AdoptOpenJDK) x64...
- □ Java (AdoptOpenJDK) x64...
- □ Java (AdoptOpenJDK) x64...
- □ .NET 4.8

Ok, we should be good to go. Lets run openstego.



Message file
C:\Users\Chris\Pictures\Saved Pictures\Enjoy.txt

Cover file
(Select multiple files or provide wildcard (, ?) to embed same message in multiple files)*
C:\Users\Chris\Pictures\Saved Pictures\linkedin background.jpg

Output stego file
C:\Users\Chris\Pictures\Saved Pictures\test.png

Options

Encryption algorithm: AES128

Password

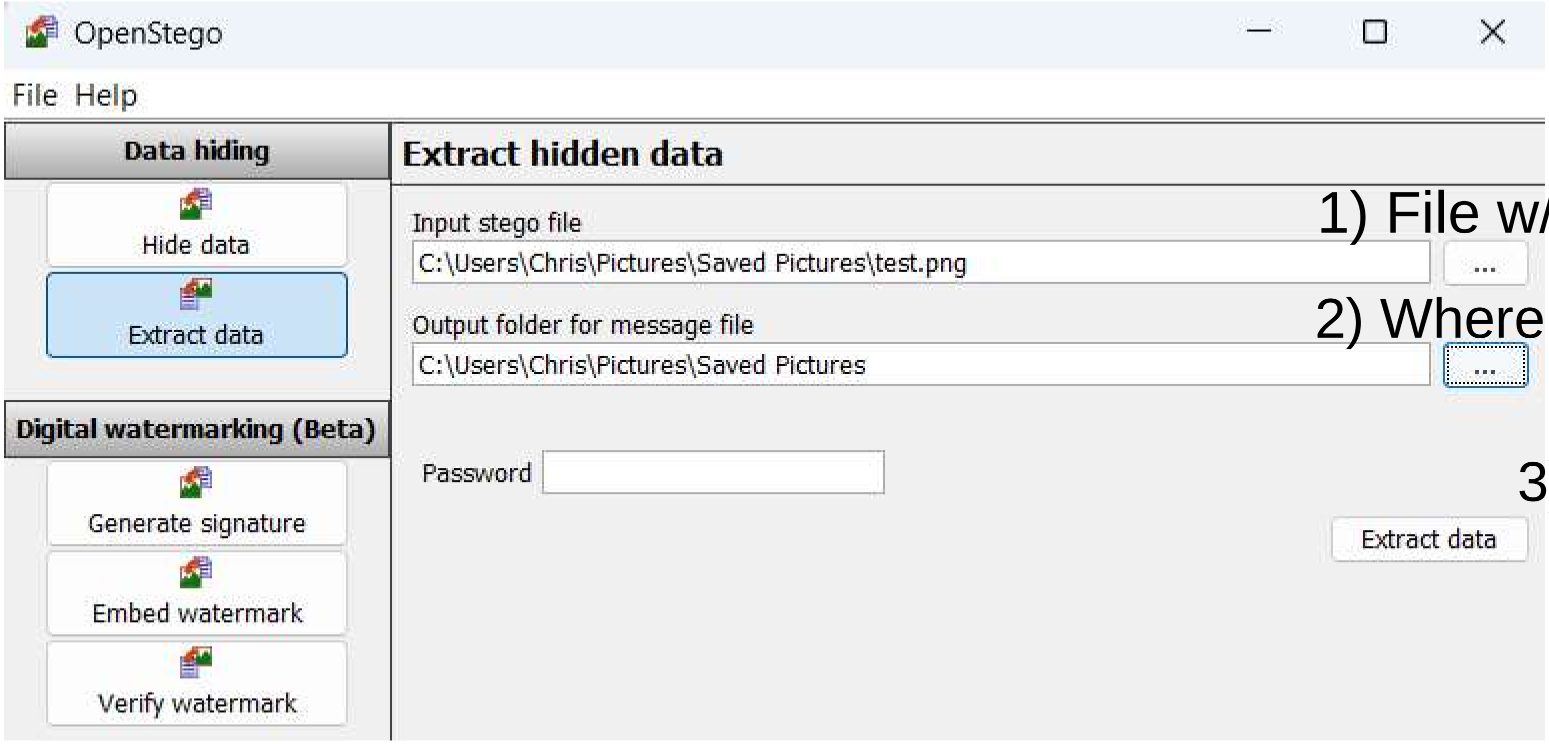
Confirm password

1) Message to hide

2) Picture to hide it in

3) New file

Then Hide that data!



Extract hidden data

Input stego file: C:\Users\Chris\Pictures\Saved Pictures\test.png

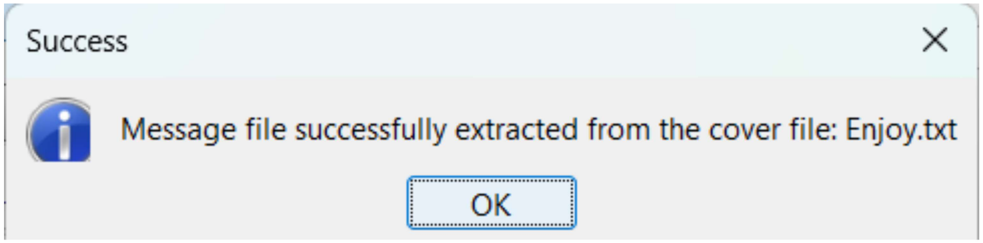
Output folder for message file: C:\Users\Chris\Pictures\Saved Pictures

Password:

1) File w/embed secret

2) Where to send secret

3) Get that secret!



Success

Message file successfully extracted from the cover file: Enjoy.txt

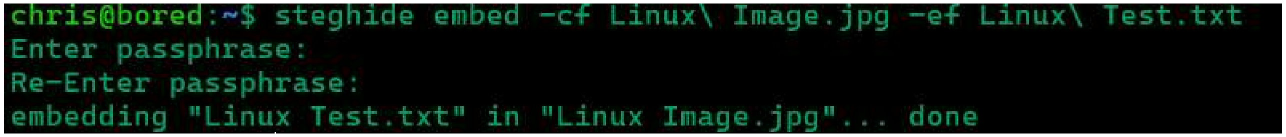
4) Success! Now to read the secret.

Another option we have is to use steghide from the CLI. It is a fairly simple process, so lets get to it.

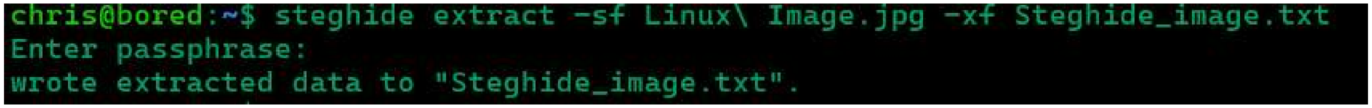
1) tell steghide we want to embed a secret then follow it with cf (coverfile) and then ef (embedfile)

2) Enter a passphrase

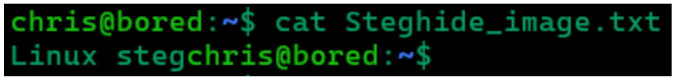
3) Secret message hidden!



```
chris@bored:~$ steghide embed -cf Linux\ Image.jpg -ef Linux\ Test.txt
Enter passphrase:
Re-Enter passphrase:
embedding "Linux Test.txt" in "Linux Image.jpg"... done
```



```
chris@bored:~$ steghide extract -sf Linux\ Image.jpg -xf Steghide_image.txt
Enter passphrase:
wrote extracted data to "Steghide_image.txt".
```



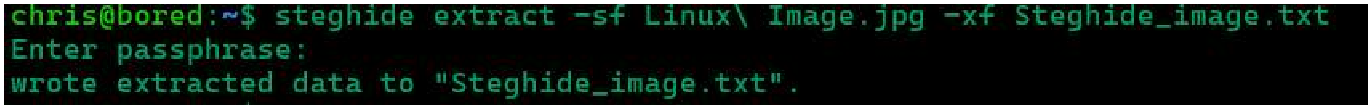
```
chris@bored:~$ cat Steghide_image.txt
Linux stegchris@bored:~$
```

This time lets rip that secret message back out.

1) tell steghide we want to extract a secret then follow it with sf (stegofile) and then xf (extractfile)

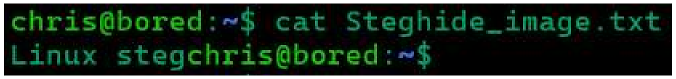
2) Enter a passphrase

3) Secret message hidden!



```
chris@bored:~$ steghide extract -sf Linux\ Image.jpg -xf Steghide_image.txt
Enter passphrase:
wrote extracted data to "Steghide_image.txt".
```

4) Cat that file...



```
chris@bored:~$ cat Steghide_image.txt
Linux stegchris@bored:~$
```

5) Boo lame secret

Ok, openstego and steghide complete! Fairly straight forward and pretty simple to use.

Play around with it and enjoy!