STEGANOGRAPHY

**STEGANOGRAPHY :**

Unlike encryption, where it's obvious that a message is being hidden, Basically steganography hide the actual data or info inside the other files in Raw format ,for example (image file). For users with no knowledge about the steganography will just consider that this is just an image rather experienced people can understand that.

Steganography is useful in situations where sending encrypted messages might raise suspicion, Steganography is used for hiding and sharing the sensitive data without any key or encryption standard. steganography can be used for eventual purpose.

* For more information about Steganography :

YouTube video : <https://youtu.be/9UZh-4Er7BQ> .

**Embed Hidden Data into a File :**

Step 1: Using Steghide is very easy. To install it from the terminal in Linux.

* apt-get install steghide

Step 2 : Once it's installed, in order to embed data in a file, type the command below.

* steghide embed -ef secretFile -cf coverFile -sf outputFile -z compressionLevel -e scheme

The arguments are broken down as follows:

* **-ef** specifies the path of the file that you want to hide. You can embed any kind of file inside of the cover file, including Python scripts or shell files.
* **-cf** is the file that the data is embedded into. This is restricted to BMP, JPEG, WAV, and AU files.
* **-sf** is an optional argument that specifies the output file. If this is omitted, the original cover file will be overwritten by your new steganographic file.
* **-z** specifies the compression level, between 1 and 9. If you prefer not to compress your file, use the argument **-Z** instead.
* **-e** specifies the type of encryption. Steghide supports a multitude of encryption schemes, and if this argument is omitted by default, Steghide will use 128-bit AES encryption. If you prefer not use encryption, simply type **-e none**.

In my example, I'm hiding secret text inside an image of a cat. I'm not overwriting the original image or compressing it, nor do I care about encryption right now.

steghide embed -ef secret.txt -cf StegoCat.jpg -e none –Z

Once you have executed the Steghide command, you will be prompted to set a password that will allow you to extract the embedded data later. So enter your passphrase and re-enter it to confirm. Once you get used to this process, it'll only take seconds to hide your data inside an image or audio file with Steghide.



**Extract Hidden Data From the File**

**STEP 3:**

Extracting hidden data from a steganographic image is even easier. The command uses the syntax below.

steghide extract -sf stegoFile -xf outputFile

Once you run this command, you'll be prompted to enter the same password you created above in order to create the extracted file. It's that simple!