# Report: Securing Data Using Steganography

**Author:** Atharva Deshpande

**Date:** October 6, 2025

---

## 1. Introduction

If you've ever wanted to send a message that no one else could read, you've probably thought about some form of encryption. In the digital world, keeping our information safe is a huge deal, and two major techniques for doing this are **cryptography** and **steganography**. They might sound similar, but they go about protecting secrets in completely different ways.

- **Cryptography**:
    - You've almost certainly used cryptography today without even knowing it. It's the technology behind things like the little padlock icon in your browser. At its core, cryptography is about scrambling a message to make it unreadable. You take your plain text, run it through an encryption algorithm with a secret key, and what comes out is a jumbled mess of characters called ciphertext.

      Think of it this way, you write a letter, lock it in a super-strong safe, and send the safe to your friend. Anyone who intercepts the package will just see a safe. They know something is inside, but they have no hope of opening it without the unique key that only you and your friend have. The main goal here is confidentiality—making the content of your message impossible to read.

- **Steganography**:
    - Steganography is the real spy-craft of the digital world. Instead of scrambling your message, the goal is to hide the fact that a message even exists. The word itself comes from Greek for "concealed writing," and that’s exactly what it is. You take your secret information and hide it inside a completely normal file, like an image, a video, or an audio file.

> Think of it this way, instead of putting your letter in a safe, you write it in invisible ink in the margins of a regular book. To anyone else, it’s just a book. No one would even suspect there's a secret message there to begin with. The main goal of steganography is secrecy. If no one knows you're sending a secret message, no one will try to crack it.

This report documents the process of using steganography to hide a text file inside a digital image. The primary tool used for this task is **steghide**, a command-line program available on Linux-based operating systems like Kali Linux and ParrotOS.

---

## 2. Methodology and Tools

The objective was to embed a secret text file into a cover image and then successfully extract it to prove the process works.

- **Operating Systems:** Kali Linux (for the main procedure) and ParrotOS (for verification).
- **Tool:** steghide command-line utility.
- **Cover File:** A JPEG image named wallpaper.jpg.
- **Secret File:** A text file named encryption.txt containing instructions on how to use steghide.

The procedure involved three main stages: **Embedding**, **Verifying**, and **Extracting**.

---

## 3. The Practical Process

Here are the step-by-step actions taken to perform steganography.

### Step 1: Embedding the Secret File

First, the secret file (encryption.txt) was hidden inside the cover image (wallpaper.jpg). This was done using the embed command in steghide.

### Command Used:

Bash

steghide embed -cf /home/kali/Documents/steg/wallpaper.jpg -ef /home/kali/Documents/steg/encryption.txt

- embed: This tells steghide that we want to hide a file.
- -cf (cover file): This specifies the image that will hide our data.
- -ef (embed file): This specifies the secret file we want to hide.

The tool then prompted for a **passphrase**, which acts like a password to protect the hidden data. This passphrase is required to extract the data later.

### Step 2: Verifying the Hidden Data

After embedding, it's important to confirm that the data is actually hidden inside the image. The info command was used for this.

### Command Used:

Bash

steghide info /home/kali/Documents/steg/wallpaper.jpg

The tool asked for the passphrase. After entering the correct passphrase, it displayed information about the embedded file, including its name (encryption.txt), its size, and the encryption algorithm used. This confirms the process was successful.

### Step 3: Extracting the Secret File

Finally, to prove the data could be recovered, the extract command was used. This pulls the hidden file out of the cover image and saves it.

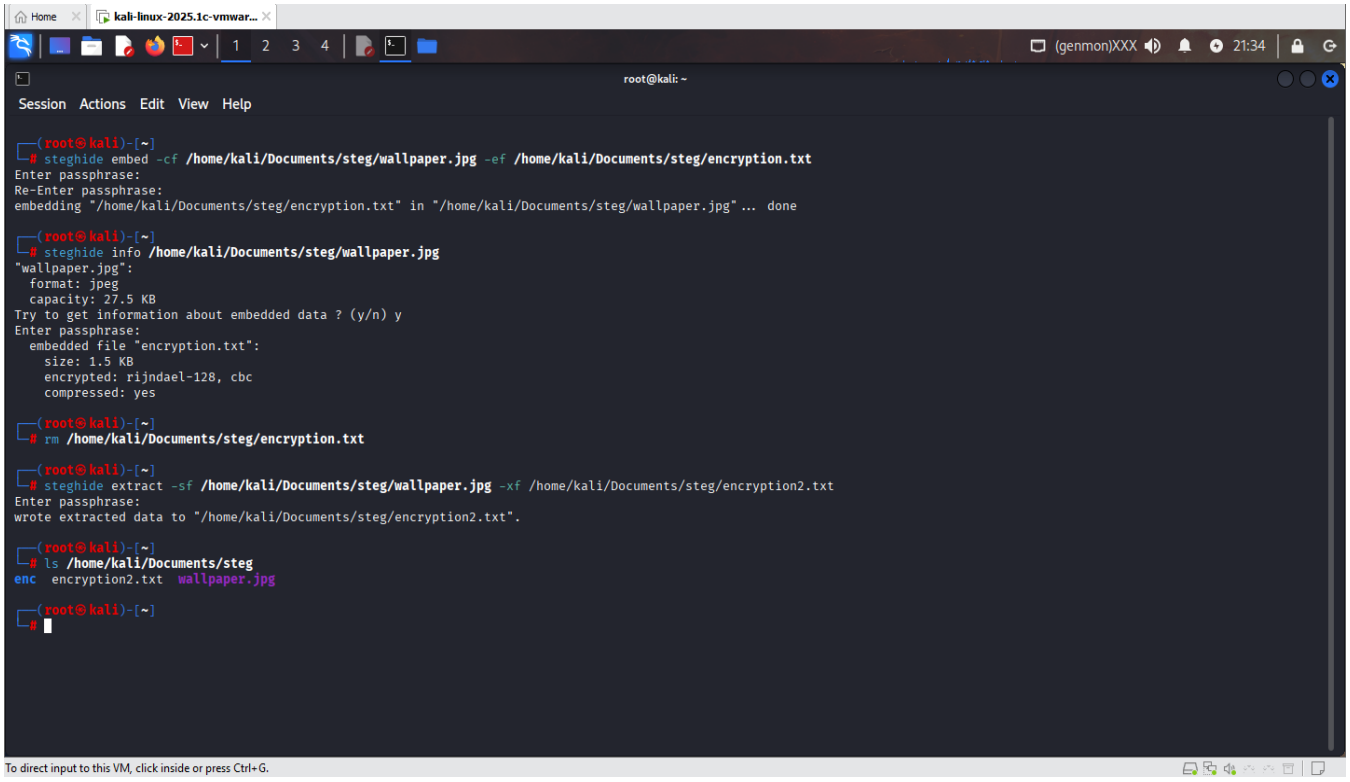
**Command Used:**

Bash

steghide extract -sf /home/kali/Documents/steg/wallpaper.jpg -xf /home/kali/Documents/steg/encryption2.txt

- extract: This tells steghide that we want to retrieve a hidden file.
- -sf (stego file): This specifies the image containing the hidden data.
- -xf (extract file): This tells steghide what to name the recovered file.

After entering the passphrase, the tool successfully extracted the hidden data and saved it as encryption2.txt. This process was also tested on ParrotOS to confirm that the steganographed image is portable and works across different systems, the steganographed image will be attached as well to verify.



## 4. Advantages of Cryptography and Steganography

Both methods provide security, but they do so in different ways.

| Feature | **Cryptography** | **Steganography** |
|---|---|---|
| **Main Goal** | **Confidentiality.** Makes data unreadable. | **Secrecy.** Hides the very existence of the data. |
| **Appearance** | The output is scrambled, obviously encrypted data (gibberish). | The output is a normal-looking file (e.g., an image). |
| **Detection** | Easy to detect that a message is encrypted, but hard to break. | Hard to detect that a secret message even exists. |
| **Best Use Case** | Best for protecting the content of a known communication. | Best for hiding the fact that a communication is happening at all. |

The biggest advantage of combining them is **layered security**. You can first encrypt a message (cryptography) and then hide that encrypted message inside an image (steganography). An attacker would first have to find the hidden message and then figure out how to decrypt it, making it extremely secure.

---

## 5. Conclusion

This task successfully demonstrated that steganography is an effective method for hiding data in plain sight. Using the steghide tool, a text file was successfully embedded within a JPEG image, verified, and extracted on both Kali Linux and ParrotOS. This proves that steganography is not only a powerful concept but also a practical technique for covertly securing and transporting information.

---

## 6. Attachments

The Wallpaper.jpg, used in this demonstration can be accessed via the following link:

- Wallpaper.jpg