Introduction

Steganography is the practice of hiding private or sensitive information within something that appears to be nothing out of the usual. The word *steganography* comes from ancient Greece (steganos), where hiding hidden messages within seemingly harmless messages became an art form. Over the years, steganography has taken on many clever and effective analog forms. One example that is often depicted in movies is invisible ink, which is not actually ink at all but a liquid, such as vinegar, that dries invisibly on paper but reappears when heated by a small flame. Another analog method is newspaper code. Popular among the working class of the Victorian era, newspaper code consisted of holes poked just above specific letters in a newspaper, such that when the dots were transferred and written together, the secret message would be revealed.

In the digital age, steganography can be used for digital watermarking, hiding data within images, or to identify the source of a given image or document (embedded copyright). Businesses sometimes employ steganography when they want to supplement the protection of encryption. In countries where encryption is not permitted (see the Crypto Law Survey at http://www.cryptolaw.org), steganography can often be used instead. While cryptography involves special encoding and decoding of messages or information, steganography replaces useless or unused data with bits of different, invisible information. This hidden information can be plain text, cipher text, or even images. Special software, such as the tools used in this lab, is needed to apply or decipher steganography.

In a forensic investigation, investigators will explore a targeted machine in search of steganographic evidence, but when they do this, they risk changing the very data they seek, potentially invalidating evidence. For this reason, they will often make a copy of an evidence drive and conduct the investigation on that image. In this lab, you will use a variety of free tools to discover possible steganographic activity in image and audio files located on a suspect's drive image. You will properly identify and extract embedded data in a carrier image and document your findings.

Lab Overview

SECTION 1 of this lab has three parts, which should be completed in the order specified.

1. In the first part of the lab, you will use Paraben's E3 to detect the presence of known steganography tools on a suspect's drive image.

2. In the second part of the lab, you will export files from the suspect's drive and use the StegExpose tool to detect the presence of steganographically-concealed data in an image file.

3. In the third part of the lab, you will use the OpenPuff steganography tool to extract the concealed data from the image file.

SECTION 2 of this lab allows you to apply what you learned in **SECTION 1** with less guidance and different deliverables, as well as some expanded tasks and alternative methods. You will experiment with different steganography tools and file formats.

Finally, you will explore the virtual environment on your own in **SECTION 3** of this lab to answer a set of questions and challenges that allow you to use the skills you learned in the lab to conduct independent, unguided work - similar to what you will encounter in a real-world situation.

Learning Objectives

Upon completing this lab, you will be able to:

1. Conduct searches using hash codes to detect the presence of steganographic software on an evidence drive.
2. Use specialized tools to search for hidden data embedded in image and audio files.
3. Identify the use of steganographic data concealment techniques for covert communication and potential injected data.
4. Extract steganographically sequestered data from identified files while conserving their integrity.
5. Report on key details of hidden files and their relevance to a forensic investigation.

Topology

This lab contains the following virtual machines. Please refer to the network topology diagram below.

- vWorkstation (Windows: Server 2019)

Tools and Software

The following software and/or utilities are required to complete this lab. Students are encouraged to explore the Internet to learn more about the products and tools used in this lab.

- Paraben's E3
- StegExpose
- OpenPuff
- S-Tools
- Xiao
- OpenStego

Deliverables

Upon completion of this lab, you are required to provide the following deliverables to your instructor:

SECTION 1

1. Lab Report file, including screen captures of the following:

- Search result and its description
- StegExpose results
- Suspicious file in Microsoft Paint
- Contents of the file extracted by OpenPuff

2. Any additional information as directed by the lab:

- Record the passphrase saved in the ReadMe file.
- Describe the contents of the hidden file. How might it be relevant to a forensic investigation?

SECTION 2

1. Lab Report file, including screen captures of the following:

- Search result and its description
- WAV file sizes and hash values in E3
- Contents of the hidden file extracted by S-Tools
- Contents of the hidden file extracted by Xiao

2. Any additional information as directed by the lab:

- Identify the image file with concealed data according to the StegExpose steganalysis tool.
- Describe the contents of the two hidden files. How might they be relevant to a forensic investigation?

SECTION 3

1. Lab Report file, including screen captures of the following:

- First file extracted by OpenStego
- Second file extracted by OpenStego

2. Any additional information as directed by the lab:

- Record the names of the files that contain concealed data.

Section 1: Hands-On Demonstration

Note: In this section of the lab, you will follow a step-by-step walk-through of the objectives for this lab to produce the expected deliverables.

1. **Review** the **Tutorial**.

   Frequently performed tasks, such as making screen captures and downloading your Lab Report, are explained in the Cloud Lab Tutorial. The Cloud Lab Tutorial is available from the User menu in the upper-right corner of the Student Dashboard. You should review these tasks before starting the lab.

2. **Proceed** with **Part 1**.

Part 1: Detect Steganography Software on a Drive Image

Note: The setting of this lab is an ongoing investigation into the activities of Beverly Gates, the HR Manager at Intricate Solutions, Inc. Senior leadership has reason to believe that Beverly is involved in a sophisticated drug trafficking operation and has recently brought in the police to investigate the matter further. As a forensic analyst working with the police, you have been given a drive image taken from Beverly's work laptop and tasked with reviewing its contents for forensic evidence of suspicious activity.

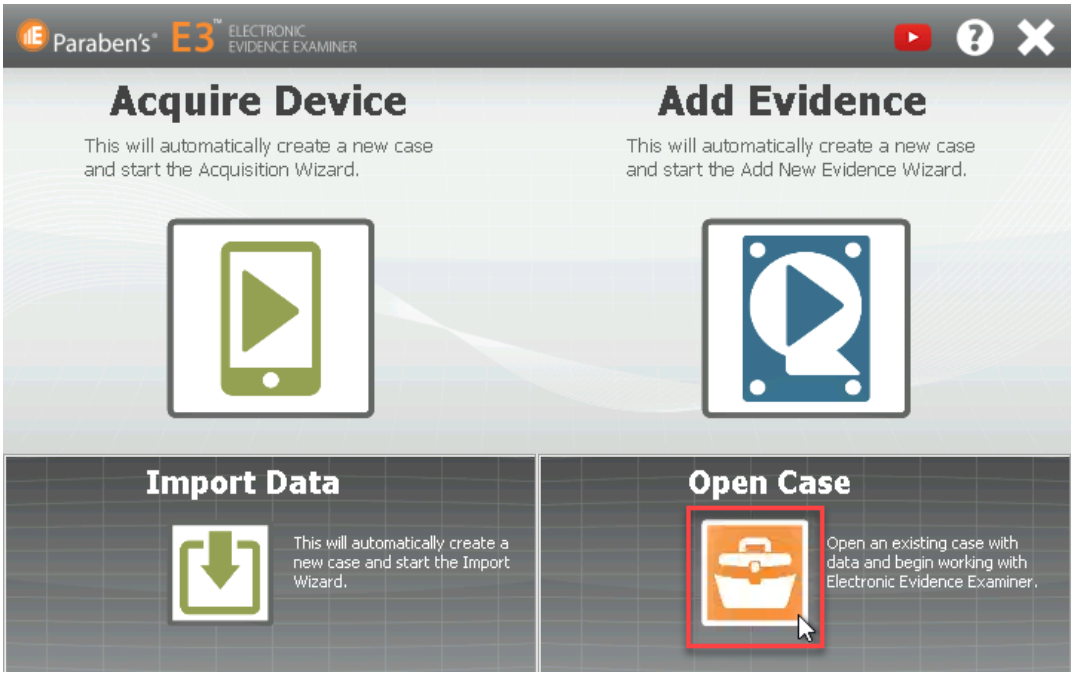
In this part of the lab, you will use Paraben's E3 and a specialized hash database to search for steganography tools on the suspect's drive image. The hash database has been compiled using the MD5 hash values for several popular steganography tools. The presence of steganography tools on a drive image that is being examined as part of a forensic investigation is a strong indicator that the drive image may also contain files with steganographically-concealed data. In the next steps, you will open an existing E3 case file and attach the hash database, which will enable E3 to utilize the hash database to search for steganography tools on the Gates drive image.

1. On the vWorkstation desktop, **double-click** the **Electronic Evidence Examiner icon** to open the E3 application.

E3 icon

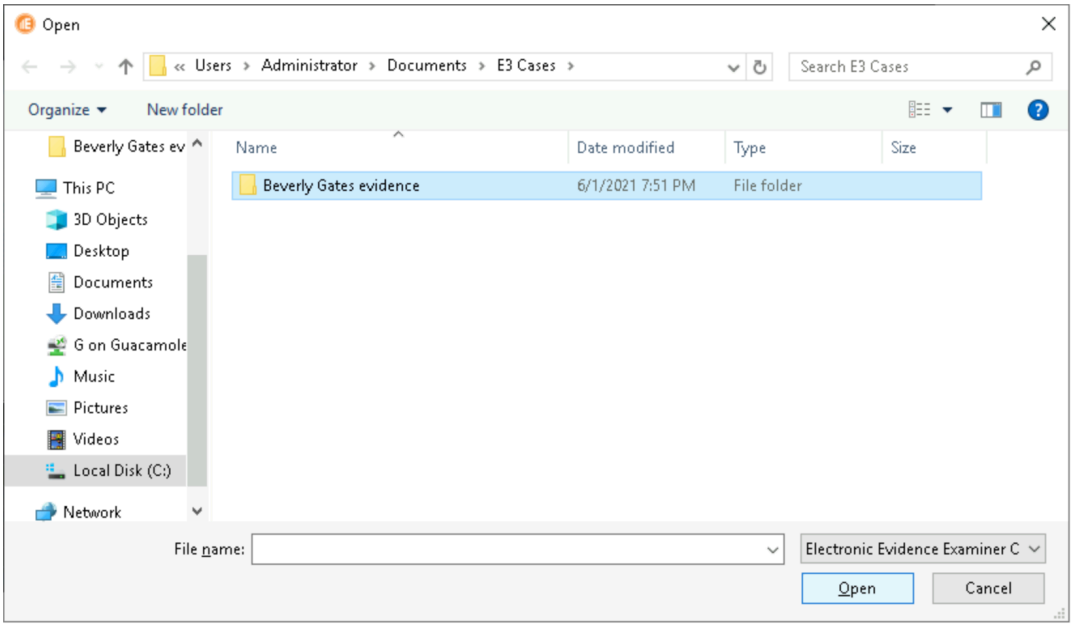
Note: E3 may take several minutes to load. The E3 welcome screen opens with shortcuts to the most common activities that forensic investigators will perform within the tool.

2. At the Welcome screen, **click** the **Open Case button** to open the Open Case dialog box.



Welcome page - Open Case

3. In the Open Case dialog box, **navigate** to **C:\Users\Administrator\Documents\E3 Cases\Beverly Gates evidence**, then **select** the **Beverly Gates evidence** case file and **click Open** to add the case to E3.

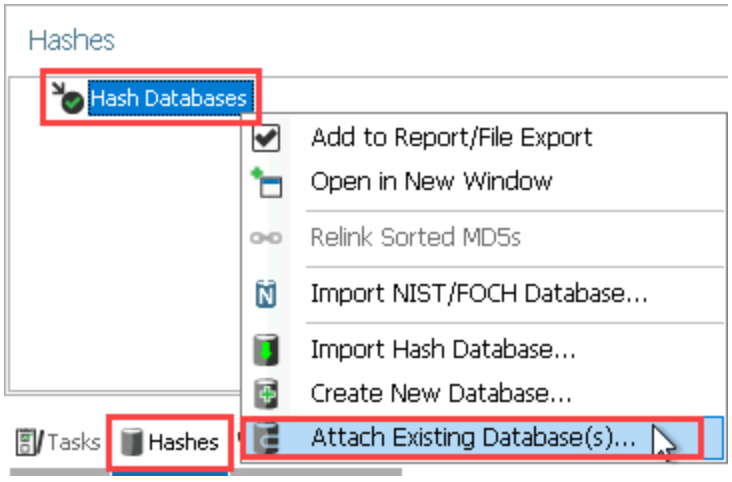


Open Case dialog box

Note: The Beverly Gates evidence case will appear in the Case Content pane. The Case Content pane is the primary display and navigation area for all evidence in the open case file. Within the Case Content pane's tree-view structure, you can access case nodes, evidence nodes, evidence type nodes, folder nodes, and data grids/streams. Technically, the Case Content pane does not store the actual evidence, but provides a series of links to elements within the physical evidence.

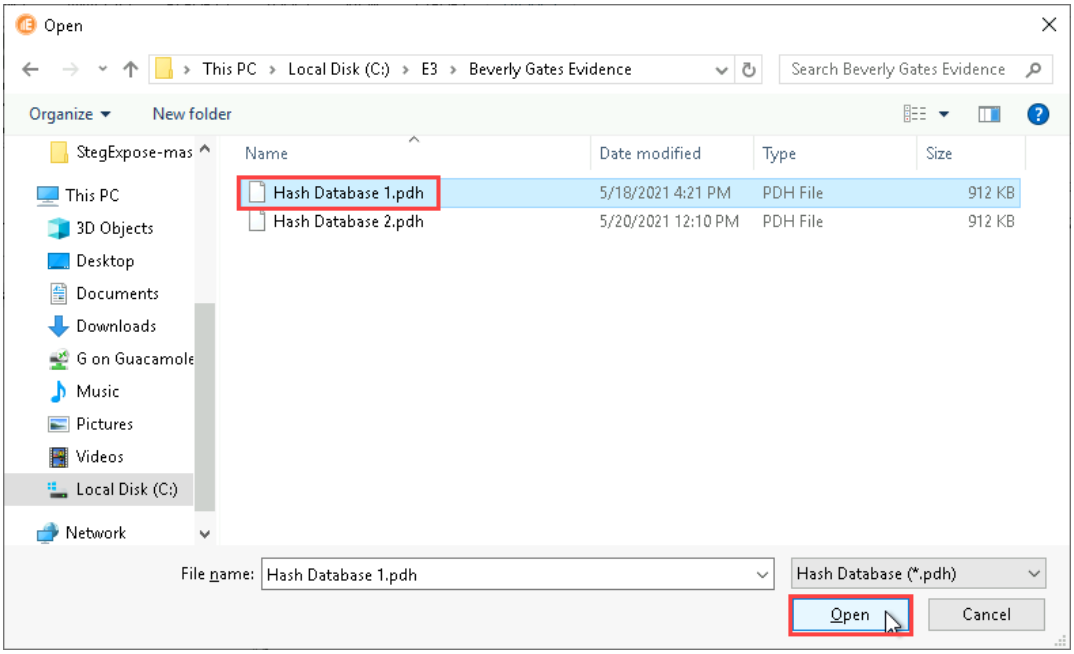
When you select a specific node or grid within the Case Content pane, the contents of the enclosing folder or database will be displayed in the Data Viewer in the center pane. Within the Data Viewer, you can select individual files, folders, and records. Additional information about the currently selected node, grid, file, folder, or record will be displayed in the Viewers pane on the right, which provides multiple ways to view your current selection.

4. In the lower-left corner, **click** the **Hashes tab**, then **right-click Hash Databases** and **select Attach Existing Database(s)** to attach the hash database containing the hash for the steganography tool detection.



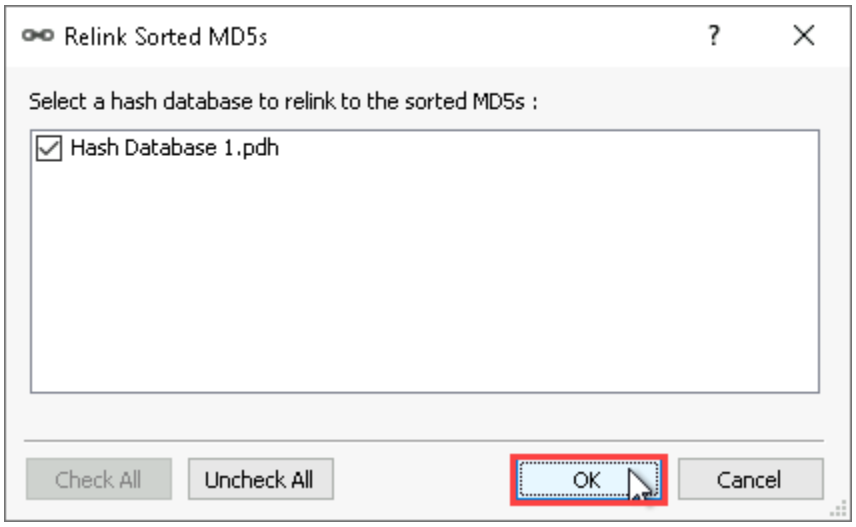
Attach a hash database

5. In the Open dialog box, **navigate** to **C:\E3\Beverly Gates evidence**, then **select** the **Hash Database 1.pdh** file and **click Open** to attach the database.



Open dialog box

6. When prompted, **click OK** to close the Relink Sorted MD5s dialog box.



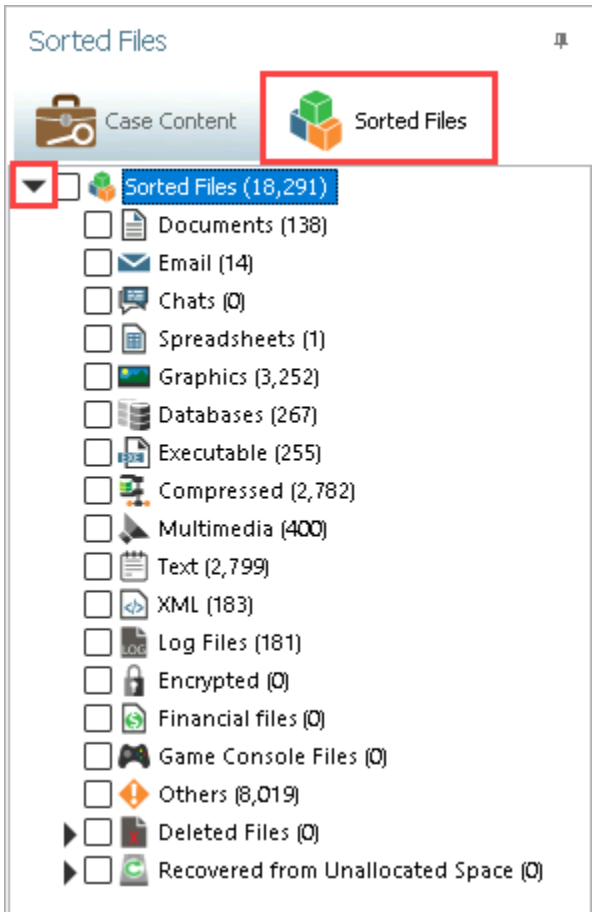
Relink Sorted MD5s dialog box

Note: This process will take about one minute to complete. You can check the status of the process by clicking the Tasks tab in the lower-left corner, then selecting the Content Analysis category. When the Task Status Notification dialog box appears, you can continue to the next step.

7. When prompted, **click OK** to close the Task Status Notification dialog box.

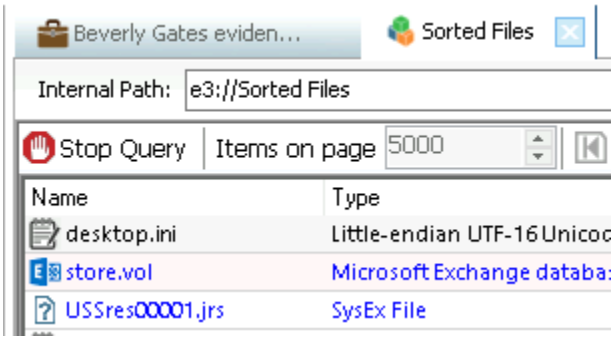
Note: The hash database is now attached. In the next steps, you will use E3's Sorted Files Search feature to perform a search of the sorted files on the suspect's drive image using the hash values contained in the newly attached hash database. The Sorted Files Search feature allows investigators to run a variety of different types of file searches within sorted evidence. In this case, the Sorted File Search feature will compare the hash values in the attached database against the previously computed hash values of sorted files and attempt to identify matches. Running a search using hash values rather than file names is a valuable technique for forensic investigators, as a hash comparison will uniquely identify files and applications that have had their names altered.

8. In the left pane, **click** the **Sorted Files tab**, then **expand** the **Sorted Files node** and **review** the sorted files.



Sorted Files

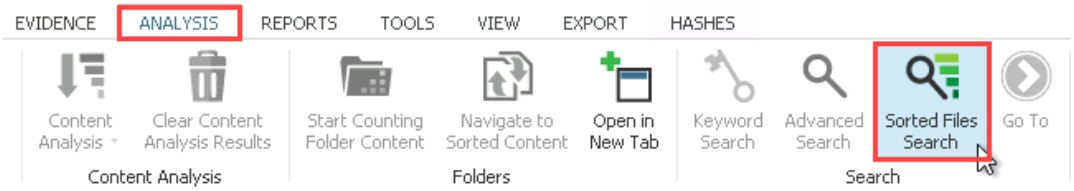
9. In the Data Viewer pane, **click** the **Stop Query button** to end the current query.



Stop Query

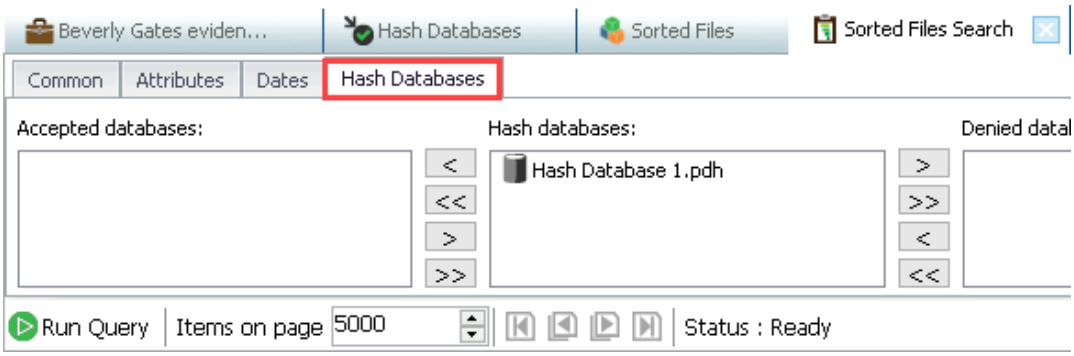
Note: Because sorting can take a lot of time, this case file has been provided with pre-sorted files. E3's sorting functions are critical in any investigation. They will save you considerable time and effort by categorizing all the various file types in the evidence as well as scanning, identifying, and preparing the evidence in the case file to be more rapidly searched.

10. On the E3 toolbar, **click** the **Analysis tab**, then **click** the **Sorted Files Search button** to open the Sorted Files Search pane in the center console.



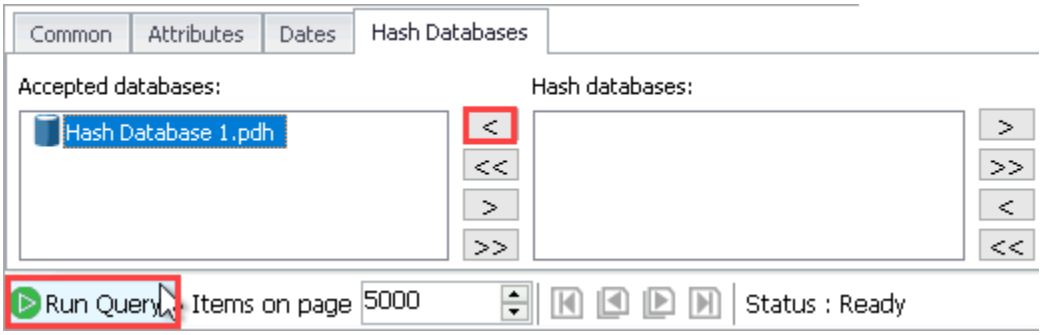
Analysis / Sorted Files Search

11. In the Sorted Files Search pane, **click** the **Hash Databases tab** to perform a search using hash values.



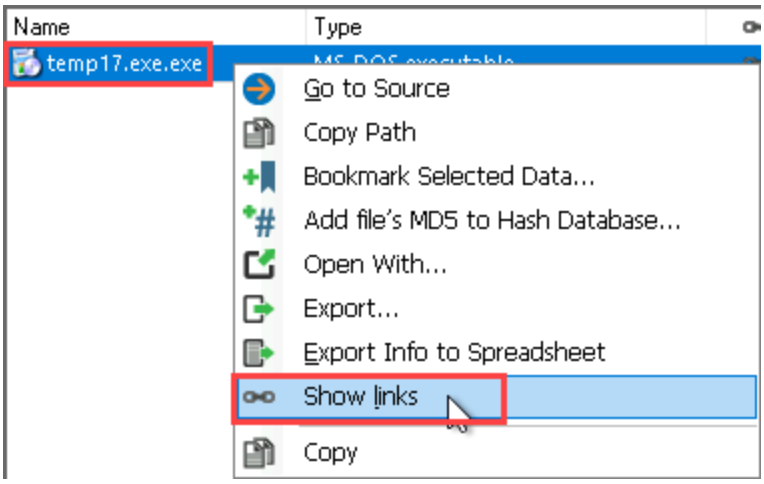
Hash Databases tab

12. On the Hash Databases tab, **select** the **Hash Databases 1.pdh** hash database and **click** the **left arrow button** to move it to the Accepted databases column, then **click Run Query** to start the search process.



Start the search process

13. When the search is complete, **right-click** the **search result** and **select Show links** from the context menu to display the external links to attached hashes.



Show links

Note: Based on the description, you should see that this file is actually the OpenPuff steganography tool, despite the fact that the file has been renamed, possibly to conceal its true identity.

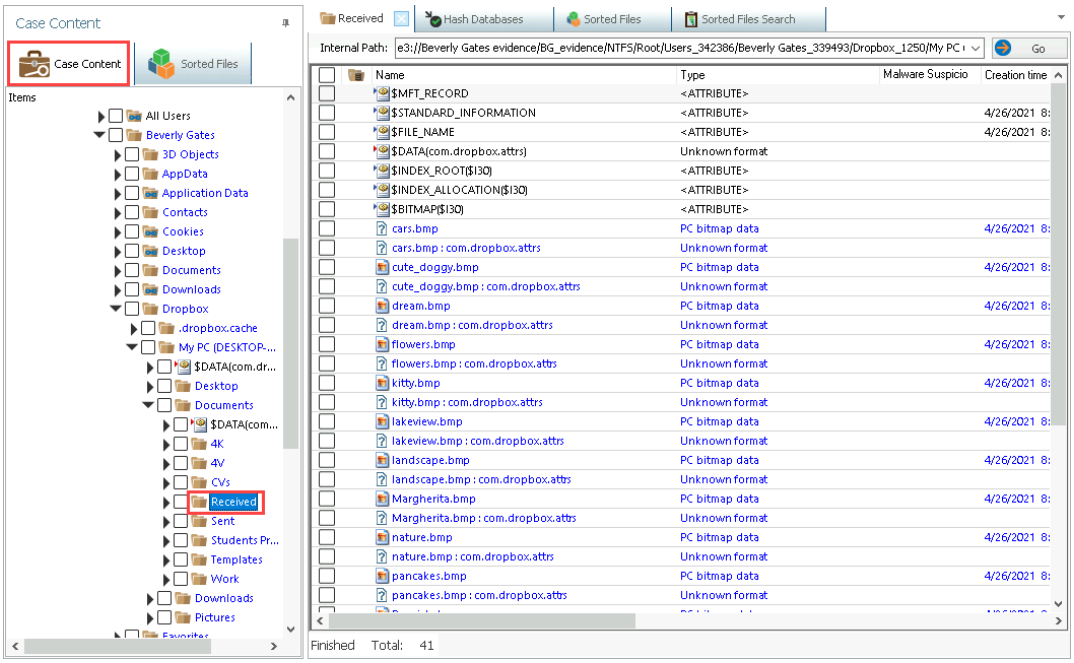
14. **Make a screen capture** showing the **search result and its description**.

15. **Close** the **External links dialog box**.

Part 2: Detect Hidden Data in Image Files

Note: In this part of the lab, you will use the StegExpose command line steganalysis tool to detect the use of LSB steganography in image files. StegExpose allows forensic investigators to scan image files in bulk and identify which ones appear to contain steganographically-concealed files. First, you will export evidence – in this case, a series of image files – from an E3 case file.

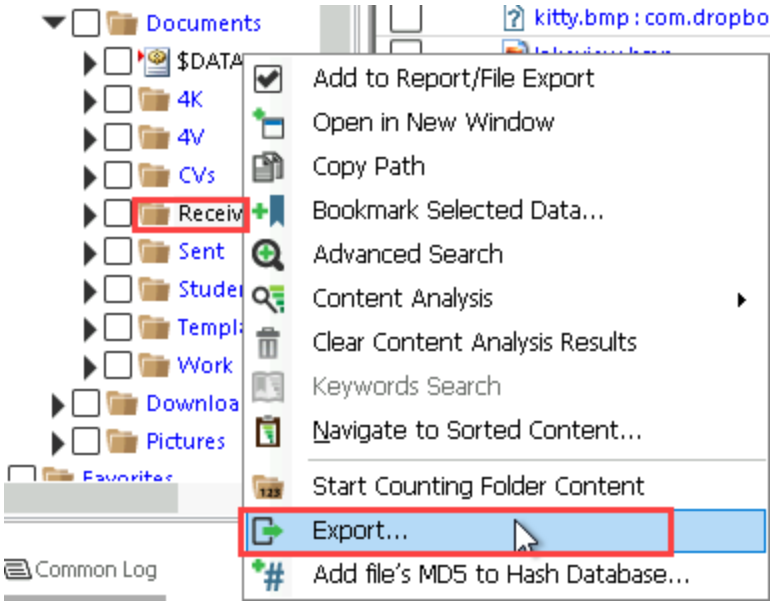
1. In the left pane, **click** the **Case Content tab**, then **navigate** to **Beverly Gates evidence\BG_evidence\NTFS\Root\Users\Beverly Gates\Dropbox\My PC (DROPBOX-RD1DJLI\Documents** and **select** the **Received folder** to open this folder in the center pane.



Received folder

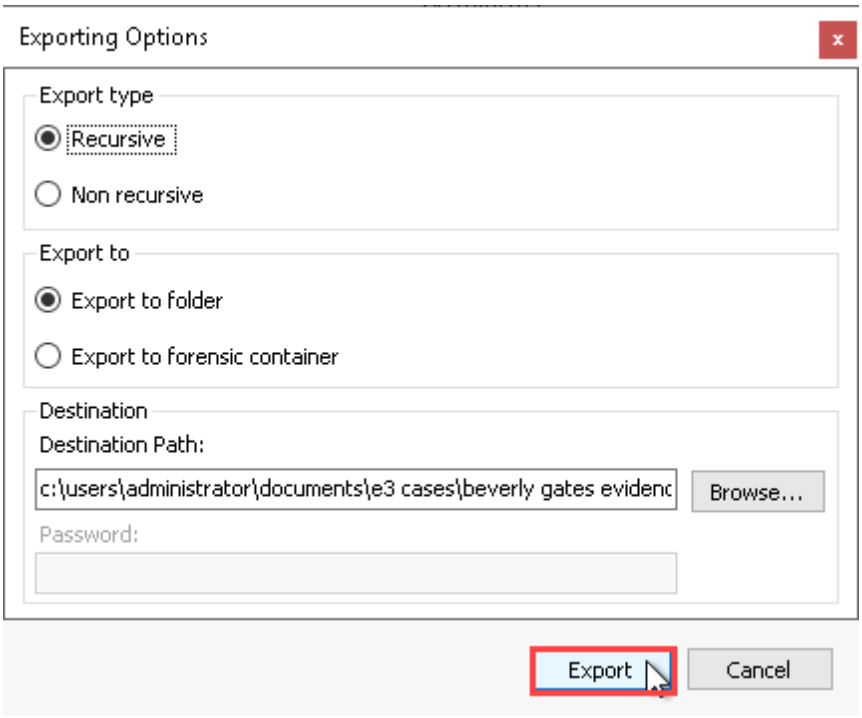
Note: If you observe the sub-folder path, you will notice that this folder synchronizes with a Dropbox account. In this example, we will look in the “Received” folder, as this is the default folder where new files sent to Beverly Gates would arrive. The folder contains several .bmp image files. For the purpose of the lab, assume that you had previously uncovered other evidence that led you to suspect this folder may contain suspicious files.

2. In the Case Content pane, **right-click** the **Received folder** and **select Export** from the context menu to open the Exporting Options dialog box.



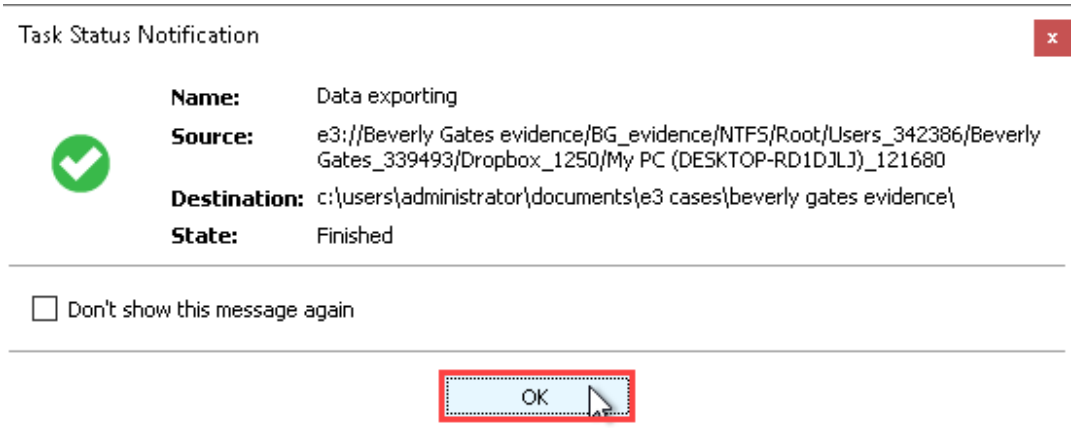
Export the Received folder

3. In the Export Options dialog box, **click Export** to export the Received folder to the Beverly Gates evidence folder.



Export Options dialog box

4. When prompted, **click OK** to close the Task Status Notification.



Task Status Notification

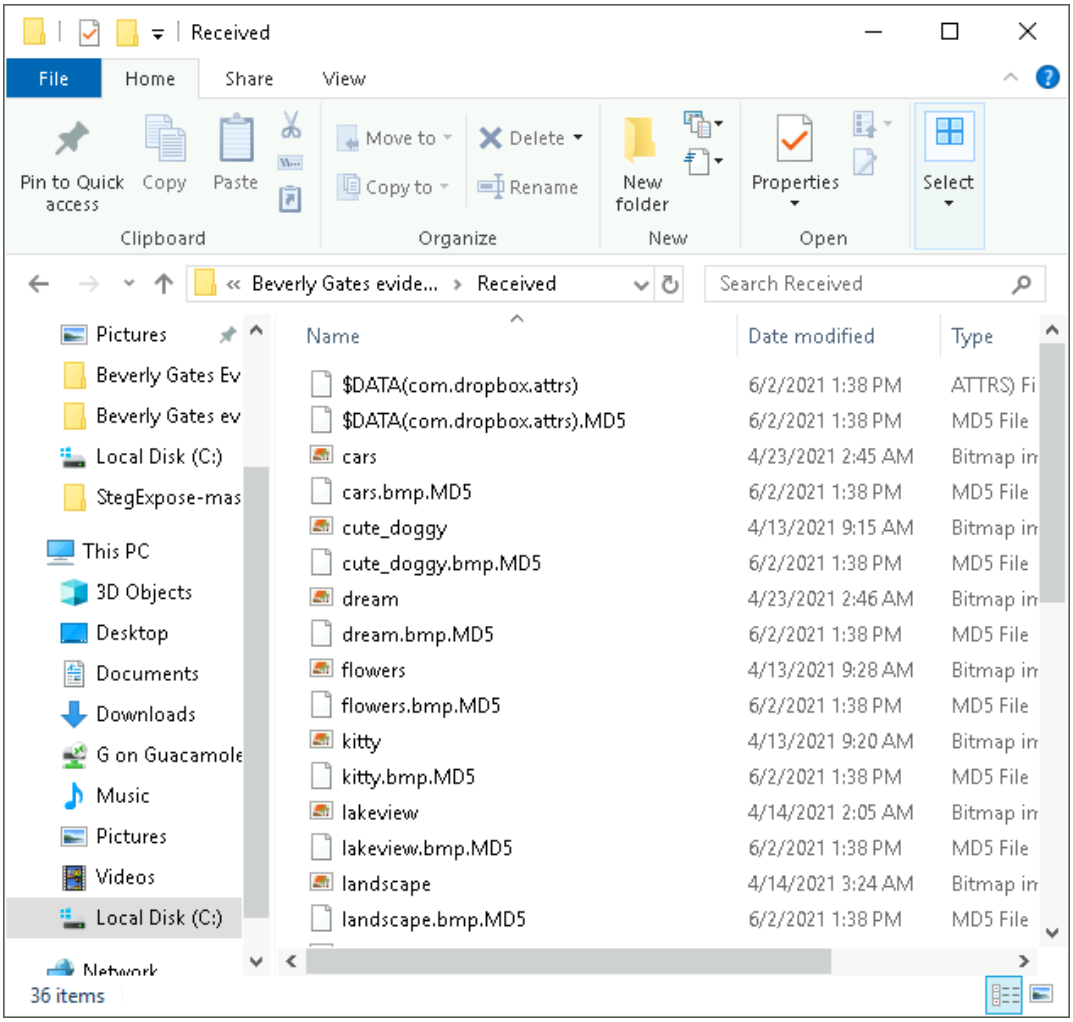
5. **Close** the **E3 application**.

6. On the vWorkstation taskbar, **click** the **File Explorer icon** to open a new File Explorer window.

File Explorer icon

7. In the File Explorer, **navigate** to **C:\Users\Administrator\Documents\E3 Cases\Beverly Gates evidence\Received** and **review** the folder contents.

   The exported folder should contain the same image files you saw in E3, as well as text files containing the MD5 hash values for each exported file.



Received folder

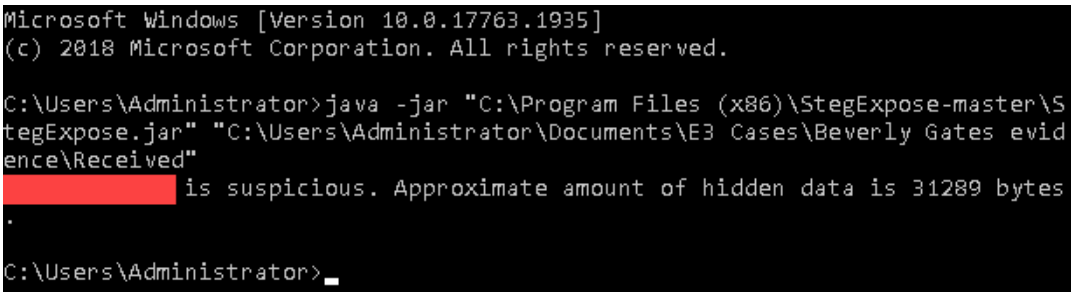
Note: The MD5 files are text files containing MD5 hash values and are used to prove the exported files' integrity – in other words, proving that the files have not been altered.

8. From the vWorkstation taskbar, **click** the **Command Prompt icon** to open a new command prompt window.

Command Prompt icon

9. At the command prompt, **type** **`java -jar "C:\Program Files (x86)\StegExpose-master\StegExpose.jar" "C:\Users\Administrator\Documents\E3 Cases\Beverly Gates evidence\Received"`** and **press Enter** to run the StegExpose tool on the contents of the Received folder.

   StegExpose will return the name of the file that it suspects contains hidden data, as well as the size of the hidden data.



Run StegExpose on the Received folder

10. **Make a screen capture** showing the **StegExpose results**.

11. **Close** the **Command Prompt window**.

12. In the File Explorer, **double-click** the *file identified by StegExpose as containing hidden data* to open it in Microsoft Paint.

13. **Make a screen capture** showing the **suspicious file in Microsoft Paint**.

14. **Close** the **Microsoft Paint window**.

Part 3: Extract Hidden Data from Image Files

Note: Now that you know Beverly Gates had the OpenPuff steganography tool installed on her computer and that there appears to be hidden data in one of her files, you should have enough information to attempt to extract the hidden data. For convenience, the OpenPuff tool has already been installed on your workstation. In this part of the lab, you will use OpenPuff's Unhide Data function to retrieve the hidden file concealed within the Lakeview.bmp file. However, you will first need to obtain the passphrase and/or encryption algorithm that were used to encrypt the file. In the next steps, you will review the remaining files in the Received folder and attempt to locate the passphrase.

1. In the File Explorer, **double-click** the **ReadMe file** within the Received folder.

Note: This file contains a single word. You know that the suspect had a steganography tool installed on her computer, and having used StegExpose to identify a file in the same folder that contains hidden data, you can reasonably conclude that this ReadMe file and its contents may be relevant to retrieving the hidden data from the suspicious file. You could reasonably conclude it may even be the passphrase required to decrypt the hidden data.

2. **Record** the passphrase saved in the ReadMe file.

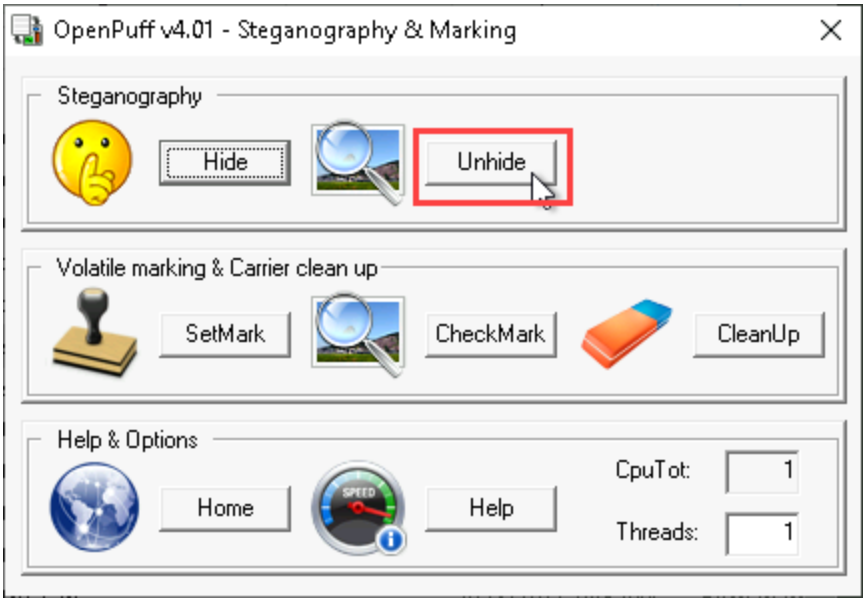
3. On the vWorkstation desktop, **double-click** the **OpenPuff - Shortcut icon** to open the OpenPuff application.

OpenPuff icon

Note: Originally released in 2004, OpenPuff is a professional-grade digital steganography tool. At its most basic level, digital steganography uses various techniques to blend or replace the underlying binary code (1s and 0s) of a digital carrier file, such as an image, audio clip, or video file, with the 1s and 0s of a payload file. Cryptographic techniques are often used in conjunction with digital steganography techniques to encrypt the hidden data and prevent unwanted access to the sensitive information contained therein.

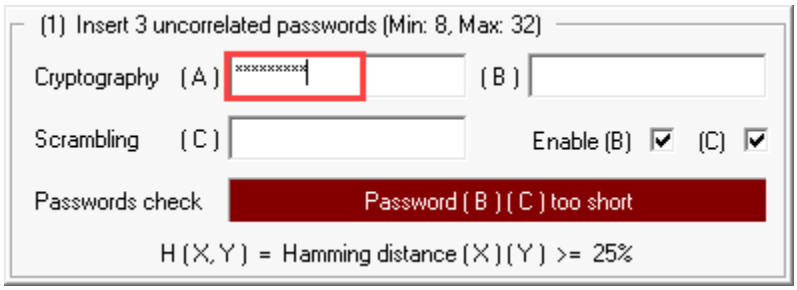
One common method of steganography employed by OpenPuff and similar tools is replacing the least-significant bit (LSB). The term *least-significant bit* refers to the left-most or right-most bit of a given byte, depending on the computer architecture. This bit is the lowest bit within the byte, which is to say that changing it would have the least impact on the numerical value of the byte. LSB steganography takes the least-significant bit within a given byte in the carrier file, and then replaces it with a bit of the payload. This method is commonly used in conjunction with image files as carriers, given that the alteration of a few minor pixels in an image will not be discernable to the unaided eye.

4. In the OpenPuff window, **click** the **Unhide button** to open the Data Unhiding window.



OpenPuff window

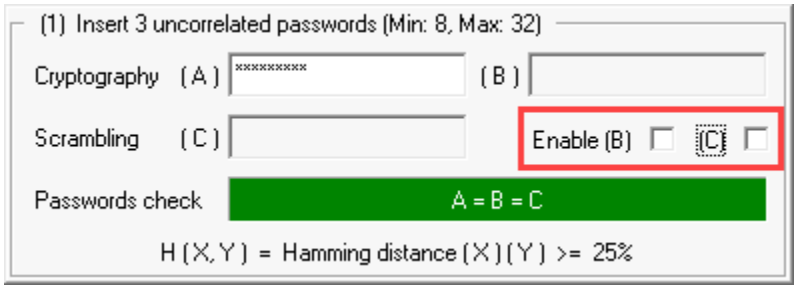
5. In the Cryptography (A) field, **type *the passphrase you located in the Received folder***.



Enter the cipher

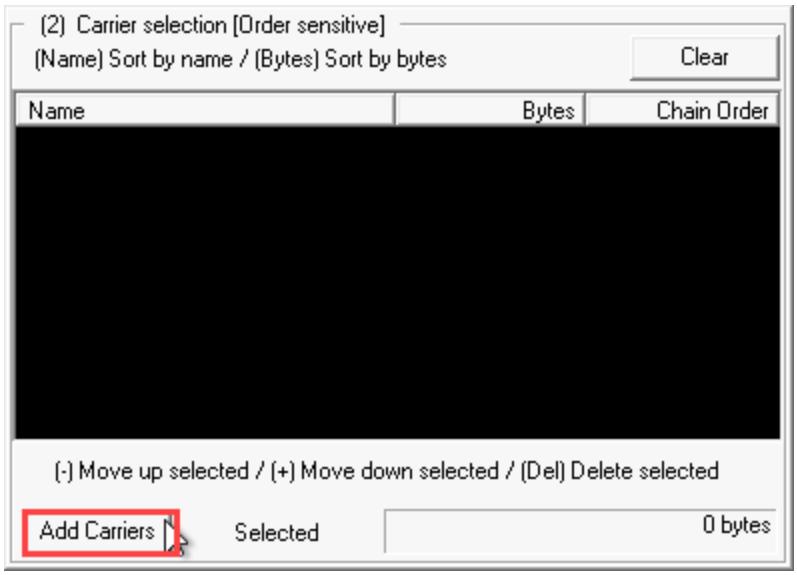
Note: This passphrase is the secret key that will reverse the encryption that has been applied to the hidden data.

6. **Click** the **Enable (B)** and **(C)** checkboxes to remove these options.



Remove the Enable (B) and (C) options

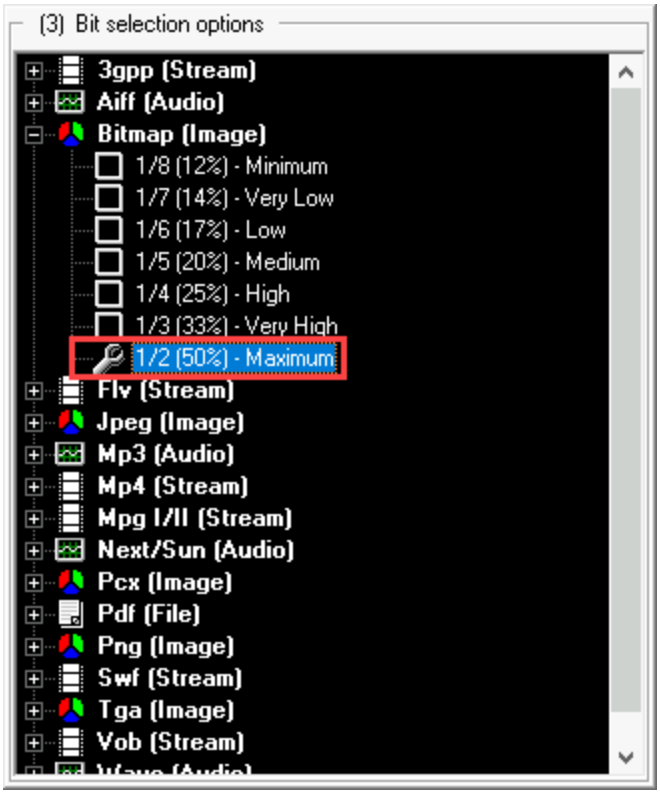
7. In the Carrier selection pane, **click** the **Add Carriers button** to select the file you wish to decrypt.



Carrier selection

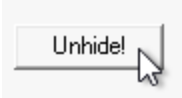
8. In the Open dialog box, **navigate** to **C:\Users\Administrator\Documents\E3 Cases\Beverly Gates evidence\Received** and **select** the ***suspicious file***, then **click Open** to add the image file.

9. In the Bit selection options pane, **expand** the **Bitmap (Image) node** and **select** the **Maximum option** to prepare the suspicious file, or stego image, so the hidden message can be revealed.



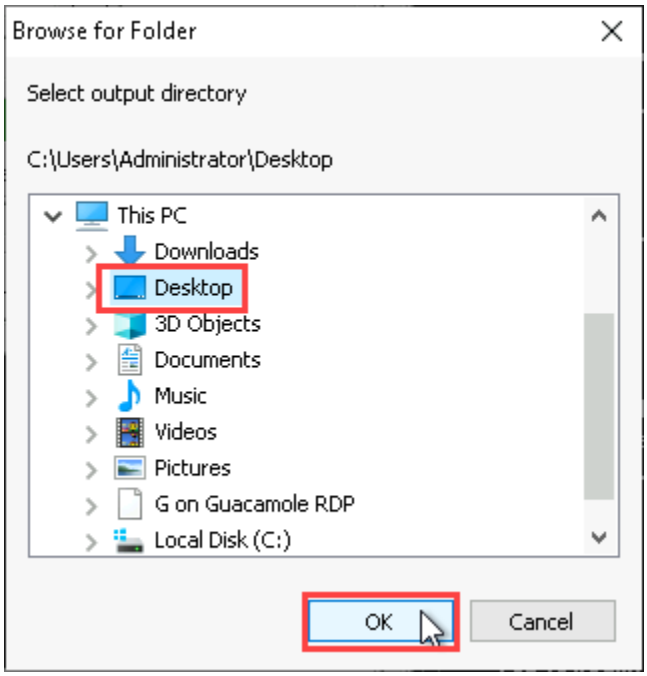
Bit selection options

10. **Click** the **Unhide! Button**.



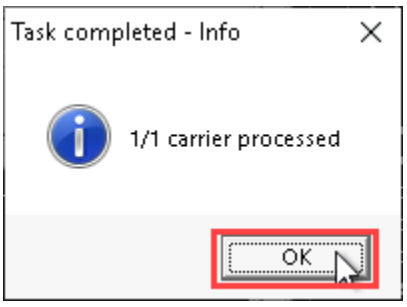
Unhide button

11. When prompted, **select** to **This PC\Desktop** and **click OK** to save the resulting file to the Desktop.



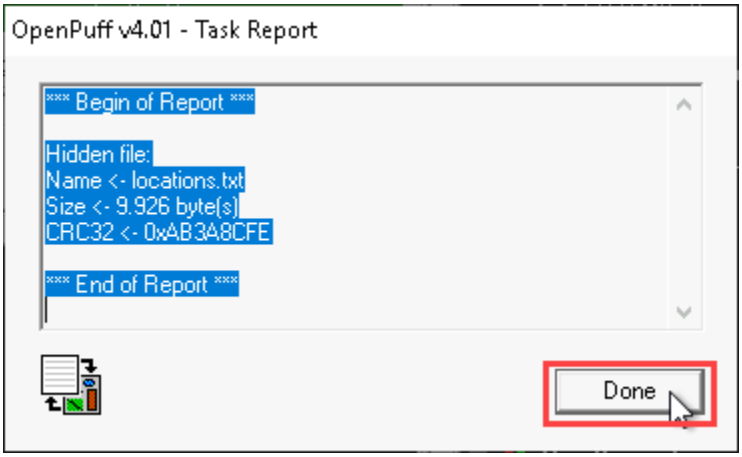
Save the file to the Desktop

12. When prompted, **click OK** to close the Task Completed dialog box.



Task Completed dialog box

13. When prompted, **click Done** to close the Task Report dialog box.



Task Report dialog box

14. **Close** any **open windows**.

15. From the vWorkstation desktop, **open** the **extracted file** and **review** its contents.

16. **Make a screen capture** showing the **contents of the file extracted by OpenPuff**.

17. **Describe** the contents of the hidden file. How might it be relevant to the current investigation?

18. **Close** any **open windows**.

Note: This concludes Section 1 of the lab.

Section 2: Applied Learning

Note: SECTION 2 of this lab allows you to apply what you learned in **SECTION 1** with less guidance and different deliverables, as well as some expanded tasks and alternative methods. You will experiment with different steganography tools and file formats.

Part 1: Detect Steganography Software on a Drive Image

Note: In Section 1 of this lab, you used E3 and a specialized hash database to identify steganography software on a suspect's drive image. You then exported a suspicious collection of images, ran the command line tool StegExpose to identify an image that contains hidden data, and used the same steganography tool identified on the suspect's drive image to extract the hidden data. However, image files are not the only files that can be used as carriers for hidden data. Audio files such as the .WAV format can also be used to conceal data. In this section of the lab, you will repeat many of the same steps you followed in Section 1, but using different steganography tools and file formats. Intruders and malicious insiders are always looking for new and more effective ways to conceal their actions. As a forensic investigator, it is important to understand the many tools and methods that bad actors may employ.

In this part of the lab, you will use a different hash database to detect additional steganography software on the suspect's drive image.

1. **Launch** the **E3 application** and **open** the **Beverly Gates evidence** case file located at C:\Users\Administrator\Documents\E3 Cases\Beverly Gates evidence.

2. In E3, **attach** the **Hash Database 2.pdh file** located at C:\E3\Beverly Gates Evidence.

3. **Open** the **Sorted Files Search** and **add** the **Hash Databases 2.pdh file** to the list of accepted databases, then **run** the search.

Note: If necessary, stop the query that is currently running in the Data Viewer pane.

4. **Open** the **Show Links dialog box** for the first search result and **review** the Description field.

   You should see three results, two of which have the same hash values and are associated with

the same tool.

5. **Make a screen capture** showing the **search result and its description**.

Part 2: Detect Hidden Data in Image and Audio Files

Note: In this part of the lab, you will again employ the StegExpose steganalysis tool to detect steganographic implants. First, you will export evidence – in this case, a series of image files – from an E3 case file. You will then use the StegExpose tool to assess whether any of these image files contains hidden data.

1. From the Case Content tab, **navigate** to **Beverly Gates evidence\BG_evidence\NTFS\Root\Users\Beverly Gates\Dropbox\My PC (DESKTOP-RD1DJLJ\Documents** and **open** the **Sent folder**.

   This folder should contain five GIF files and two WAV files, as well as other system artifacts.

2. **Export** the **Sent folder** to the Beverly Gates evidence folder.

3. **Open** the **Command Prompt**, then **execute** the **command to run the StegExpose tool** on the contents of the Sent folder.

4. **Identify** the image file with concealed data according to the StegExpose steganalysis tool.

5. **Open** and listen to each WAV file using the Windows Media player.

   Based on their contents alone, the files should appear to be the same.

6. **Restore** the **E3 application** and **navigate** to the **Multimedia category** in the Sorted Files

pane, then **locate** the two WAV files you exported from the Sent folder.

Note: Pay attention to the file size and hash values. The fact that both files appear to contain the same audio, but have different file sizes and hash values could potentially indicate that the larger file contains steganographically concealed data.

7. **Make a screen capture** showing the **WAV file sizes and hash values in E3**.

8. **Close** the **E3 application**.

Part 3: Extract Hidden Data from Image and Audio Files

Note: In this part of the lab, you will use the S-Tools and Xiao applications. S-Tools will allow you to extract concealed data in BMP, GIF, and WAV files. S-Tools also allows users to hide multiple files in a single audio file or image, and compresses the data before it is encrypted, which further reduces the chances of detection. Both S-Tools and Xiao use passwords to protect files and hides them in images using LSB methods. As a first step, you will locate the cipher keys needed to reveal the hidden information in the suspect files.

1. In Sent folder, **open** the **key.txt file** and **review** its contents.

Note: The file appears to contain the cipher keys/passphrases and algorithms used to encrypt the image and audio files in the Sent folder. When hiding data, various encryption algorithms can be applied; for example, RC2, RC4, DES, 3DES (triple DES), AE129, and so on. In this case, the key file makes reference to the triple DES and RC2 encryption algorithms. The words to their left can be safely assumed to be cipher keys.

2. From the vWorkstation desktop, **open** the **S-Tools.exe application**.

3. **Drag and drop** the image file that you confirmed to contain concealed data into the S-Tools window.

4. In the S-Tools window, **right-click** the image file and **select Reveal** from the context menu.

5. When prompted, **enter** one of cipher keys from the Keys.txt file as the passphrase and **select** the **corresponding encryption algorithm**, then **click OK** to extract the hidden file.

   If your cipher key / encryption algorithm combination was correct, S-Tools will open a new window containing the name of the hidden file.

7. **Right-click** the **hidden file name** and **select Save as**, then **save** the hidden file to the vWorkstation desktop.

8. **Open** the **hidden file**.

9. **Make a screen capture** showing the **contents of the hidden file extracted by S-Tools**.

10. From the vWorkstation desktop, **open** the **Xiao application**.

11. In the Xiao window, **click Extract Files**, then **select Load Source File**, **navigate** to the **location** of the audio file you suspect to contain concealed data, and **click Next**.

12. In the password field, **type** the correct password for this file and **click Extract File**.

    By process of elimination, you should be able to guess the correct password.

13. When prompted, **enter** `WAV_extract` as the file name and **save** the hidden file to the vWorkstation desktop.

14. From the vWorkstation desktop, **open** the **WAV_extract file** and **review** its contents.

15. **Make a screen capture** showing the **contents of the hidden file extracted by Xiao**.

16. **Describe** the contents of the two hidden files. How might they be relevant to the current investigation?

17. **Close** any **open windows**.

Note: This concludes Section 2 of the lab.

Section 3: Challenge and Analysis

Note: The following exercises are provided to allow independent, unguided work using the skills you learned earlier in this lab - similar to what you would encounter in a real-world situation.

Part 1: Detect More Hidden Data

Another forensic analyst working on the same case has heard that you've successfully identified hidden evidence in some of Beverly's files. The analyst has extracted files from Beverly's hard drive that they believe may also contain hidden evidence, but unfortunately they lack the tools and experience to retrieve it, so they've asked you to take a look.

Using StegExpose, determine if any of the files in the C:\Steganography Evidence\Section 3 folder contain hidden data.

Record the names of the files that contain concealed data.

Part 2: Extract More Hidden Data

Your initial scan of the new evidence files has returned some promising results - two new files that may contain hidden data, to be exact.

Based on the StegExpose results in Part 1, extract the hidden data from the image files using OpenStego. You will find the cipher keys saved in the C:\Steganography Evidence\Section 3 folder.

Make a screen capture showing the **first file extracted by OpenStego**.

Make a screen capture showing the **second file extracted by OpenStego**.

Note: This concludes Section 3 of the lab.