Introduction

Data recovery techniques are the principal ancestor to modern digital forensics. When computers were much simpler and not connected to the Internet, digital forensic investigations were largely a matter of finding and recovering deleted data from a single hard drive. While today's forensic investigations can span computers, networks, smartphones, and the cloud, data recovery is still a critical component.

When a user deletes a file, the file is typically moved to the Trash (Mac OS) or Recycle Bin (Windows). However, moving a file to the Trash or Recycle Bin does not actually delete the file. It simply changes the file's location and designates the file as planned for deletion. Just like the trash can in your own home, you can easily reach in and pull out an item if you change your mind about throwing it out. A file will only be considered "deleted" after the user runs the Empty Trash or Empty Recycle Bin function. However, emptying the Trash or Recycle Bin still does not actually remove the data. The only thing that is removed is the reference in the master file table that tells the operating system where the file was located. From the operating system’s perspective, you are only removing the map to the data, not the data itself. Removing the map gives the operating system permission to mark that area of the hard drive as "free" and overwrite it should new data need to be written there, but until an overwrite occurs, the data remains intact.

Of course, there are many other ways of deleting data beyond moving a file to the Recycle Bin and emptying it. Some more effective data deletion methods include delete with overwriting and physical destruction. A delete with overwriting means that the data is deleted, and then new random data is written to the drive several times, making the recovery of the original files very difficult or impossible. Physical destruction is just that – the destruction of the actual hard drive along with all the data stored on it. This can be accomplished through various methods, including shredding, melting, or crushing the hard drive.

In this lab, you will assume the role of a digital forensics specialist who has been assigned to a case involving intellectual property theft. After obtaining a search warrant, the local authorities have seized multiple computers from the suspects and transferred images of their hard drives to the digital forensics team. So far, no incriminating evidence has been found on the drive images, but your boss, the lead investigator, believes the suspects may have deleted incriminating evidence. As a digital forensics specialist, you will use professional data recovery tools and techniques to recover deleted data from different operating systems and file systems.

Lab Overview

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

1. In the first part of the lab, you will recover data from a Windows NTFS drive image using E3.

2. In the second part of the lab, you will recover data from a Linux Ext4 drive image using Autopsy.

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 pause your investigation to learn more about recovering deleted data directly from live Windows and Linux systems. You will also learn how partially deleted data can be recovered using data carving techniques.

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

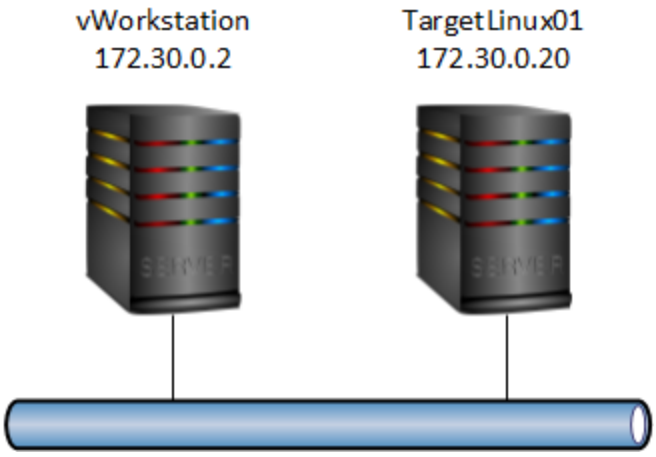
Upon completing this lab, you will be able to:

1. Understand how files are deleted and recovered.
2. Recover deleted files from NTFS, Ext4, FAT, and APFS drive images.
3. Recover deleted files using data carving techniques.
4. Recover deleted files from a live Windows system.
5. Recover deleted files from a live Linux system.

Topology

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

- vWorkstation (Windows: Server 2019)
- TargetLinux01 (Linux: Ubuntu)



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.

- E3
- Autopsy
- PhotoRec

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:

- List of recovered files in the E3 Trash folder
- Patent file in the File Viewer
- Recovered files in the File Explorer
- Contents of the list of deleted files in Autopsy
- Recovered patent file

2. Any additional information as directed by the lab:

- None

SECTION 2

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

- Contents of the RAR archive in the /mnt/media/home/ash directory
- Failed mount attempt on the /dev/sdb5 device.
- Compressed files recovered by PhotoRec
- Backup files recovered from the RAR archive

2. Any additional information as directed by the lab:

- None

SECTION 3

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

- Patent file recovered from the FAT32 drive image within E3
- Patent file recovered from the APFS drive image within Autopsy

2. Any additional information as directed by the lab:

- None

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: Recover Deleted Files from an NTFS Drive Image with E3

Note: In this part of the lab, you will recover deleted files from a Windows drive image. Windows is the most commonly used desktop operating system among home and business users, which means that it is equally common as a source of evidence in forensic investigations. Starting with the release of Windows NT in 1993, the default file system for the Windows operating system has been the proprietary NTFS (New Technology File System). Since its release, Microsoft has released four updated versions of NTFS, the most recent being 3.1.

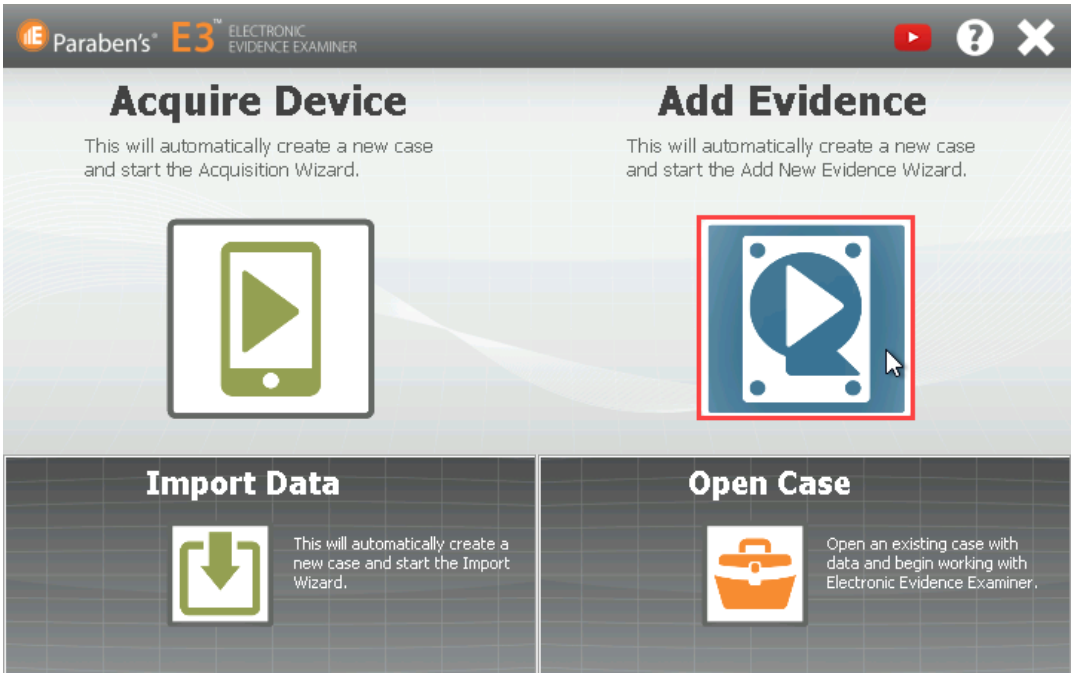
In the next steps, you use E3, a professional-grade digital forensics tool, to import an NTFS-formatted drive image as evidence.

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

E3 icon

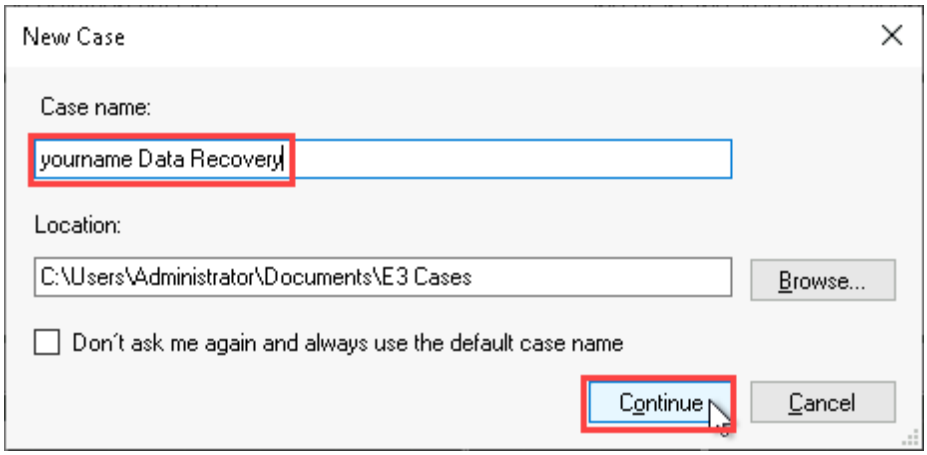
Note: E3 may take several minutes to load. The E3 welcome screen opens with shortcuts to the most common activities that forensic investigators will perform within the tool.

2. At the Welcome screen, **click** the **Add Evidence button** to open the New Case dialog box.



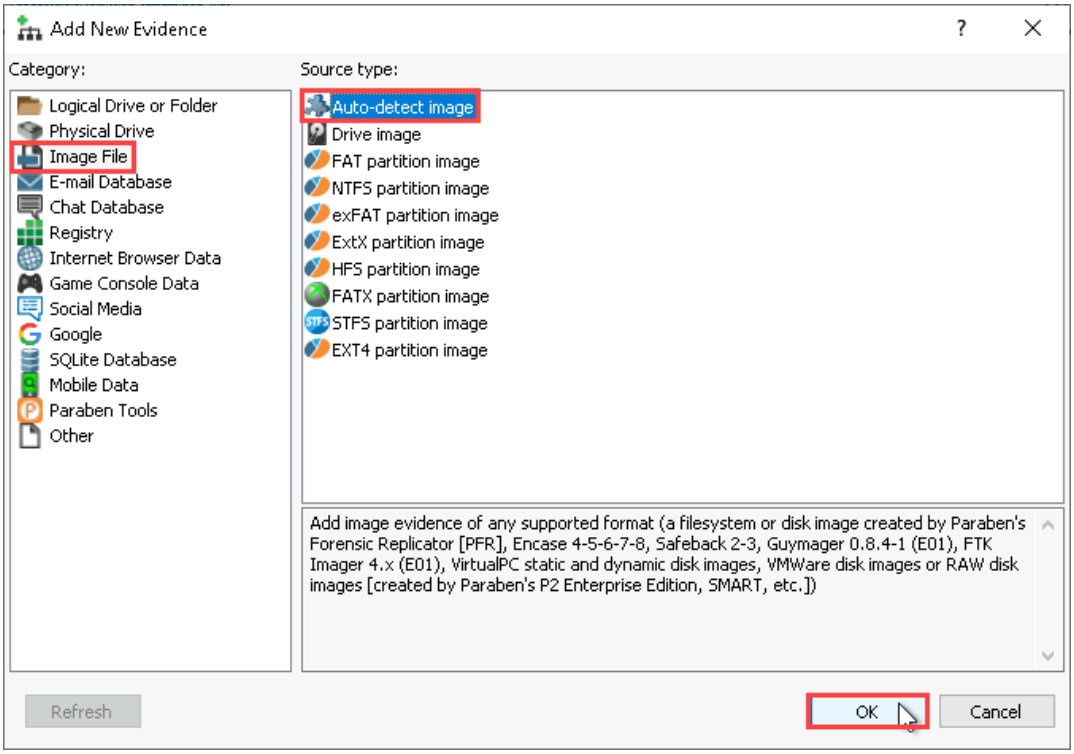
Welcome page - Add Evidence

3. In the New Case dialog box, **type** ***yourname*** **`Data Recovery`** in the Case name field, replacing *yourname* with your own name, then **click Continue** to create your new case file and open the Add New Evidence dialog box.



New Case dialog box

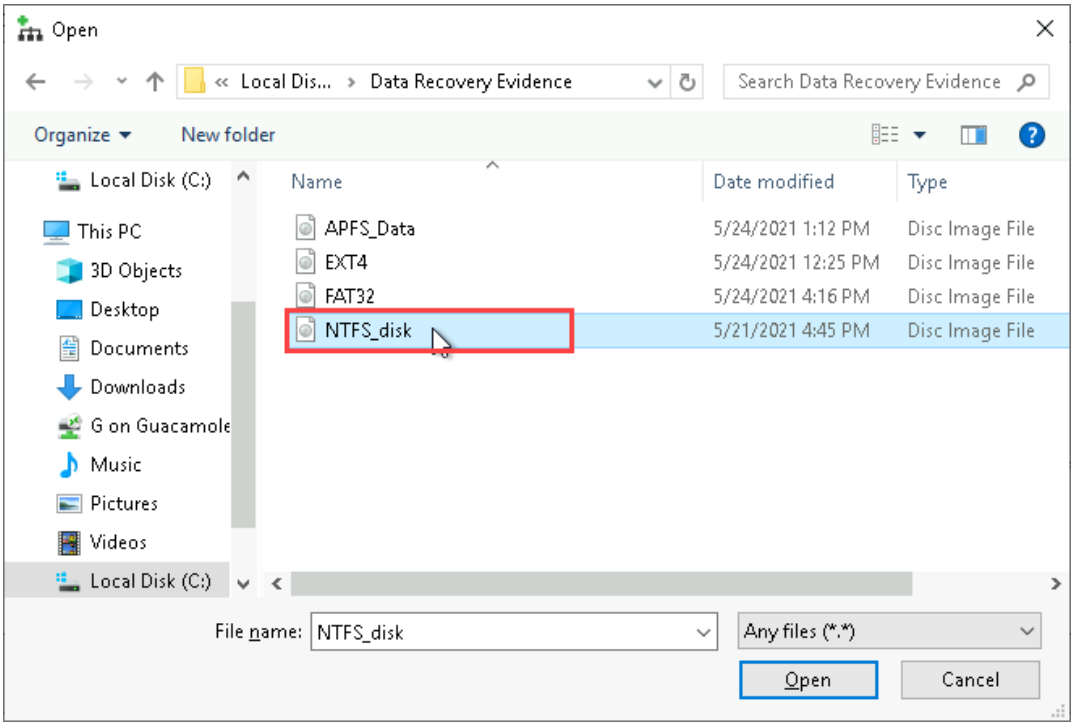
4. In the Add New Evidence dialog box, **click** the **Image File category**, then **select** the **Auto-detect image Source type** and **click OK** to continue.



Add New Evidence - Auto-detect image

Note: When importing drive images as evidence in E3, you can manually define the type of file system used on the drive image if you know it. However, E3 can quickly determine this on its own using the Auto-detect image function.

5. In the Open dialog box, **navigate** to the **Data Recovery Evidence folder** (This PC > Local Disk (C:) > Data Recovery Evidence) and **double-click** the **NTFS_disk.img** to select the digital drive image for this lab.



Open dialog box

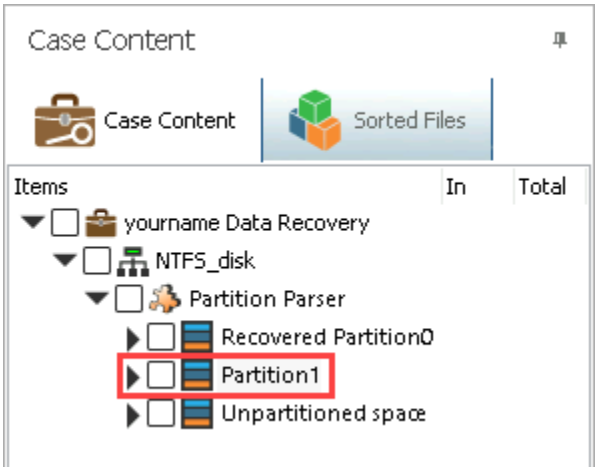
6. When prompted, **click OK** to accept the default name for the drive image and add the drive image to the Case Content pane.

Evidence name

Note: The *yourname* Data Recovery 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.

7. In the Case Content pane, **navigate** to ***yourname* Data Recovery\NTFS_disk\Partition Parser\Partition1** to open the NTFS Settings dialog box for Partition1.



Partition1

Note: E3 is capable of automatically restoring deleted files and folders when drive images are imported as evidence. However, for NTFS file systems, this process can take a great deal of time. For this reason, users are presented with the NTFS Settings dialog box when they access NTFS evidence. The NTFS Settings will allow the user to determine if deleted data should be restored by E3, and if so, to what extent.

When E3 and similar tools retrieve deleted data from NTFS file systems, they do so by referencing NTFS metafiles. In the context of digital forensics and data recovery, there are two critical metafiles in the NTFS file system: the Master File Table ($MFT) and the bitmap ($BitMap). The Master File Table serves as the file system's principal record of metadata for all files and folders in the system. This includes details like file name, location, creation date, permissions, size, and the clusters where the data is stored. The bitmap file is a record of which clusters on the disk are allocated and unallocated. When a file is deleted by emptying the Recycle Bin in an NTFS file system, the data is still not actually deleted. The corresponding record in the Master File Table is simply updated to indicate that file has been deleted. The record will remain intact, as will the file data, at least until the corresponding clusters – now marked as unallocated space – are overwritten.

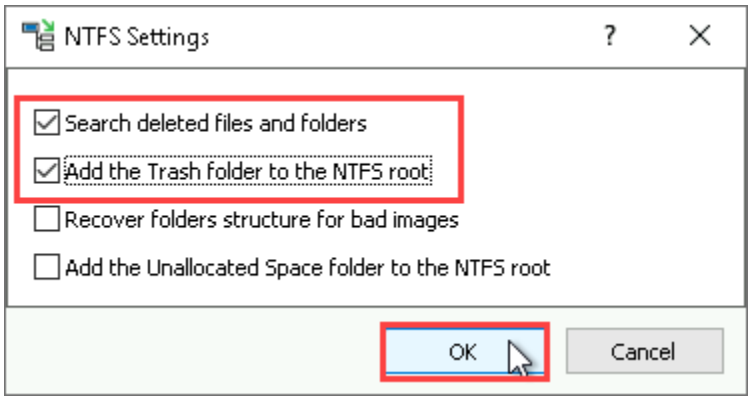
Using this information, E3 and similar tools are able to quickly identify deleted files and recover them if the data has not been overwritten.

8. In the NTFS Settings dialog box, **click** the **Search deleted files and folders checkbox** to direct E3 to search for deleted files and folders in the unallocated space in Partion1.

9. **Click** the **Add the Trash folder to the NTFS root checkbox** to direct E3 to add the Trash folder to Partition1 in the Case Content browser.

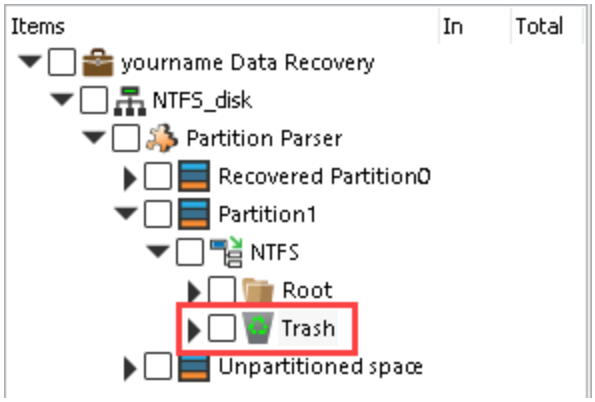
Note: In E3, the Trash refers to the folder where E3 collects any deleted files and folders that it was able to recover from the unallocated space on the drive. It should not be confused with the file system's Trash or Recycle Bin directory, which is still available within the drive image's normal file structure.

10. **Click OK** to close the NTFS Settings dialog box.



NTFS Settings

11. In the Case Content pane, **navigate** to **Partion1/NTFS/Trash** to display the contents of the Trash folder in the Data Viewer.



Trash folder

Note: In E3, when a file is identified as deleted in the file system, a red X will appear next to the file name. E3 will attempt to recover the file if possible.

12. In the Data Viewer, **review** the files recovered by E3.

Note: Recall that the setting for this lab is an investigation of intellectual property theft. As you select each recovered file in the Trash folder, E3 will display the file in different viewers in the Viewers pane. You may need to expand this pane to see the name of each viewer tab, which includes:

- Properties
- Thumbnails View
- File View
- Document View
- Text View
- Extracted Text View
- Hex View
- File Slack: Text View
- File Slack: Hex View

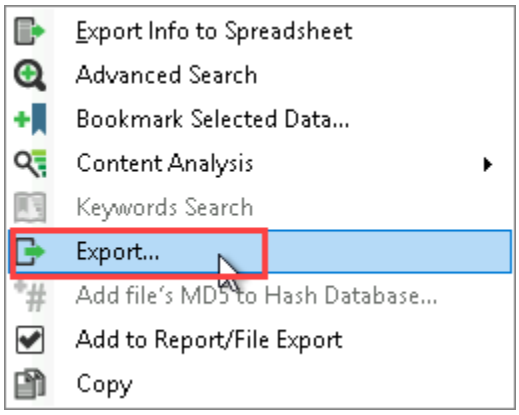
Using the File Viewer tab, you should see that several of the deleted files contain copies of patents and product designs. Given the scope of the investigation, these files all appear to be potentially incriminating evidence.

13. **Make a screen capture** showing the **list of recovered files and folders in the E3 Trash folder**.

14. In the Data Viewer, **click** the **alarmhead.jpg file**, then **click** the **ww_2020_ 6_845.jpg file** while holding the Shift key to select all of the files.

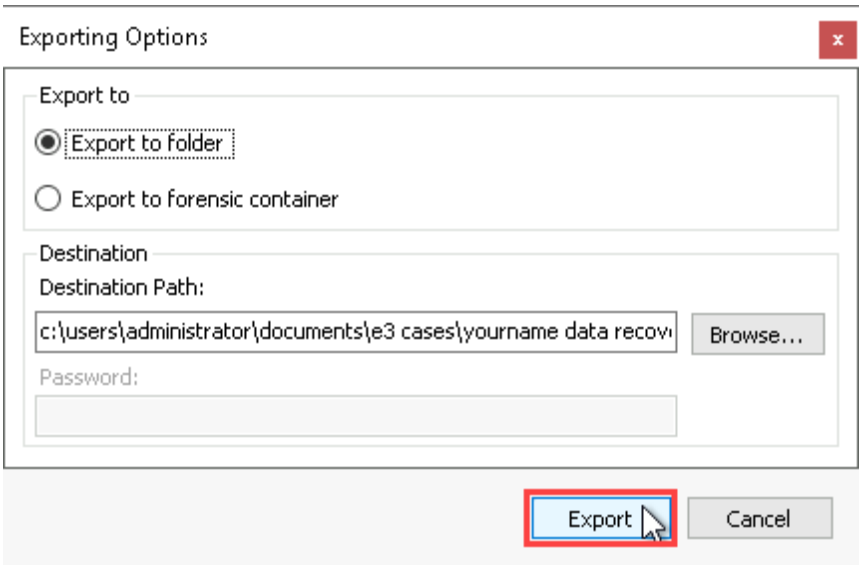
Select the files

15. **Right-click** any of the **selected files**, then **select Export** from the context menu to open the Export Options dialog box.



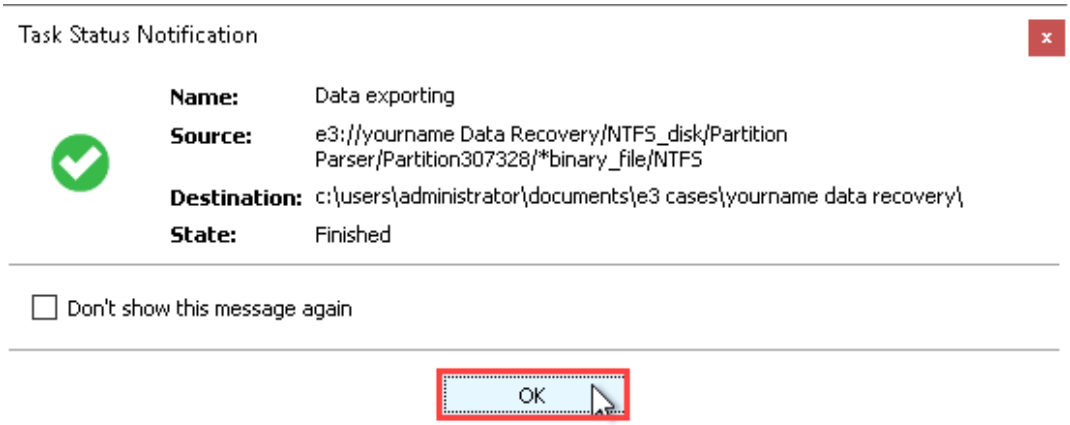
Context menu

16. In the Export Options dialog box, **click Export** to save the recovered files to the case file folder.



Export Options dialog box

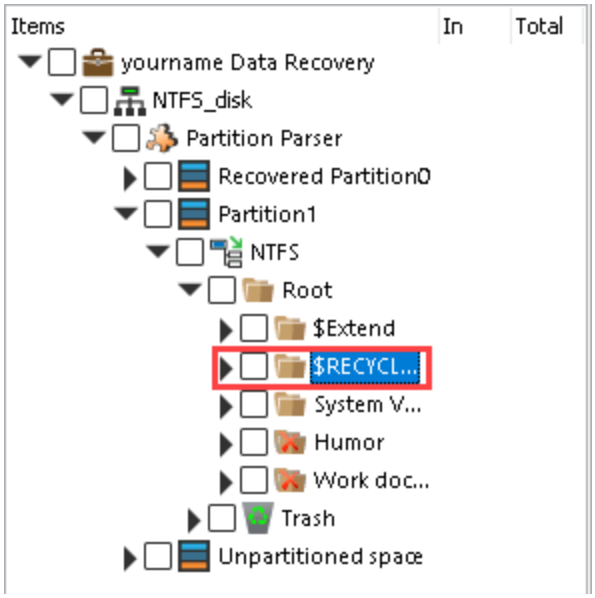
17. When prompted, **click OK** to close the Task Status Notification dialog box.



Task Status Notification dialog box

Note: In the next steps, you will extend your investigation to the Windows drive image's Recycle Bin. While the contents of the Recycle Bin are even further from true deletion than the deleted files you just recovered, many users may not realize this or have simply forgotten to empty the Recycle Bin.

18. In the Case Content pane, **expand** the **Root node**, then **select** the **$RECYCLE.BIN folder** to open it in the Data Viewer.



&RECYCLE.BIN folder

Note: When files are moved to the Recycle Bin in Windows 10, two files are created in the Recycle Bin. The first file is assigned a random alpha-numeric name beginning with $I. This file contains the original location of the deleted file. The second file is assigned the same random name, but beginning with $R. The second file is the actual deleted file.

19. **Review** the contents of the $RECYCLE.BIN folder and **locate** a patent file related to sharing location information.

20. **Make a screen capture** showing the **patent file in the File Viewer**.

21. **Repeat steps 15-17** to export the patent file to the case file.

22. **Close** the **E3 window**.

23. From the taskbar, **click** the **File Explorer icon** to open a new File Explorer window.

File Explorer icon

24. In the File Explorer window, **navigate** to the **C:\Users\Administrator\Documents\E3 Cases\yourname Data Recovery folder**.

25. **Make a screen capture** showing the **recovered files in the File Explorer**.

26. **Close** the **File Explorer**.

Part 2: Recover Deleted Files from an Ext4 Drive Image with Autopsy

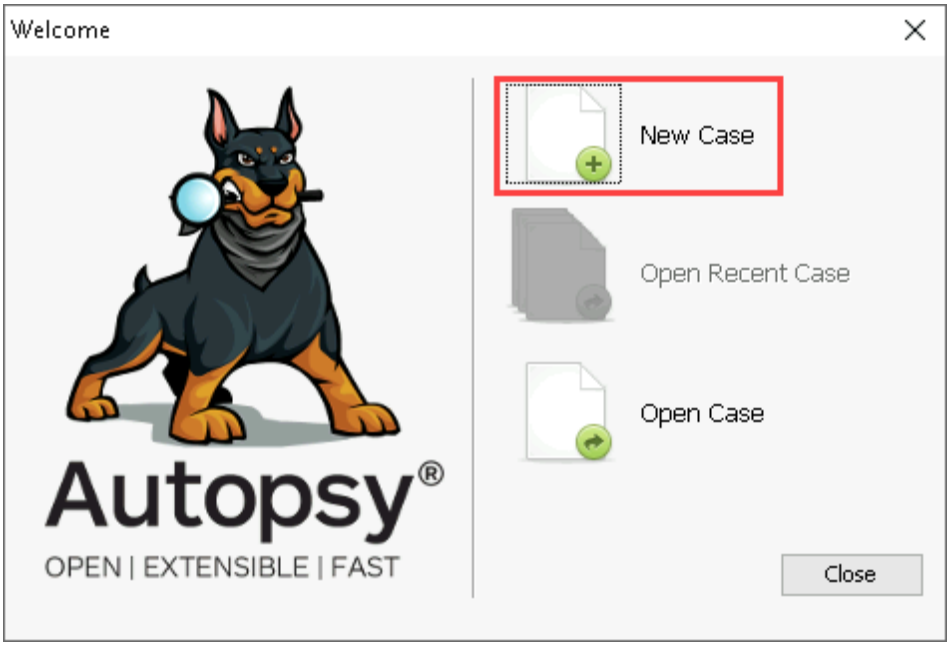
Note: In this part of the lab, you will recover deleted files from a Linux drive image. While Linux is not commonly used as a desktop operating system among casual users, the fact that Linux is free and open-source makes it an ideal underlying operating system for servers, routers, Android smartphones, and IOT-enabled devices. Although Linux is available in several different distributions and famously customizable, the most commonly used file system for Linux is Ext4 (Extended File System version 4).

In the next steps, you will use Autopsy, a popular open-source digital forensics tool, to acquire an Ext4-formatted drive image as evidence.

1. On the vWorkstation desktop, **double-click** the **Autopsy shortcut** to launch the Autopsy application.

Autopsy icon

2. In the Autopsy Welcome window, **click** the **New Case option** to create a new case file.

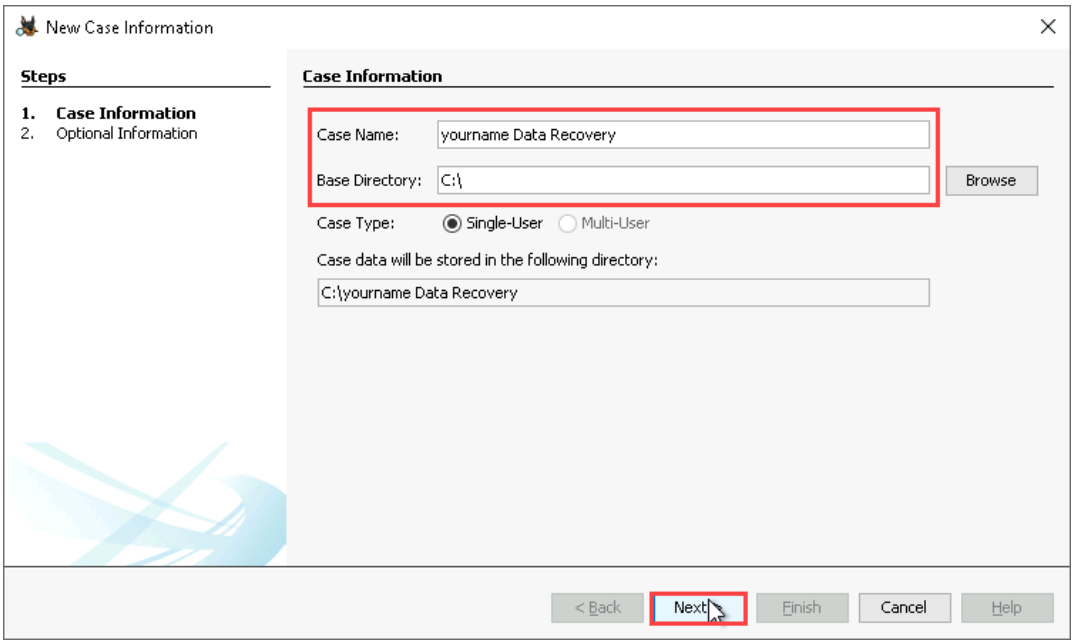


Welcome window

3. On the Case Information page, **type** ***yourname*** **`Data Recovery`** in the Case Name field.

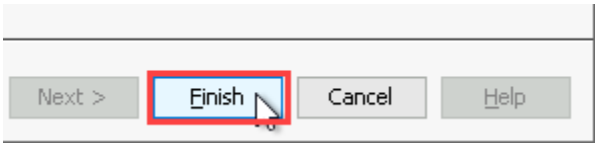
4. **Type** **`C:\`** in the Base Directory field to define the location of the Autopsy case file.

5. **Click Next** to continue to the Optional information page.



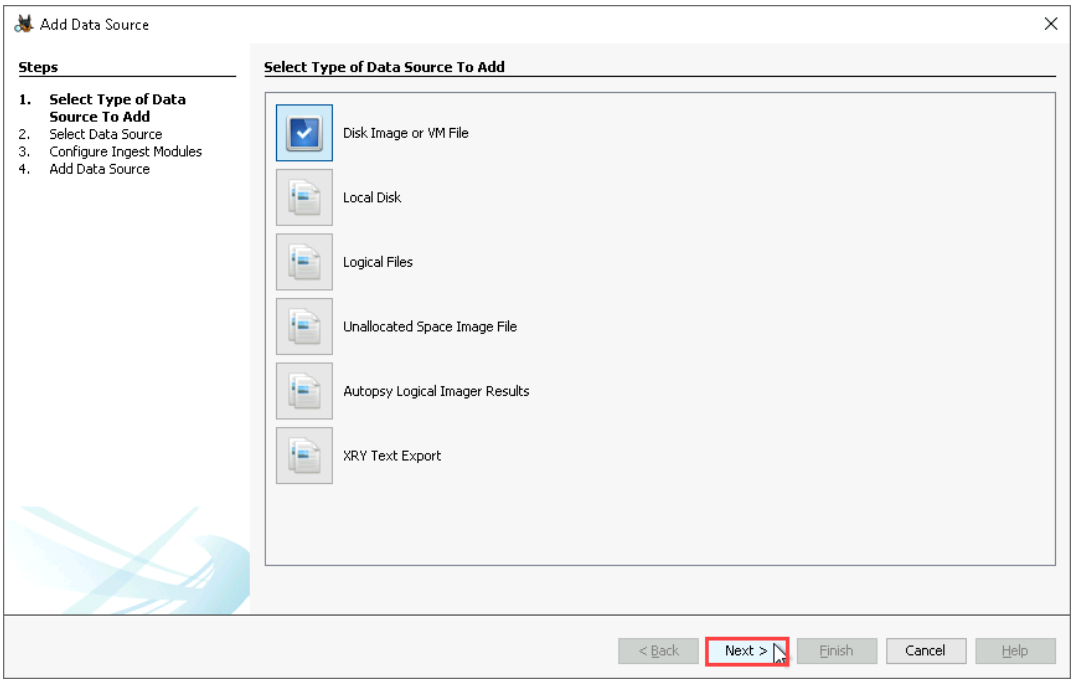
Case Information

6. On the Optional Information page, **click Finish** to save your case file and open the Add Data Source Wizard.



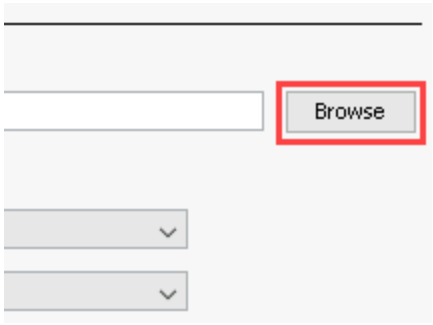
Optional Information

7. On the Select Type of Data Source To Add page, **click Next** to accept the default (Disk Image or VM File) and continue.



Select Type of Data Source to Add

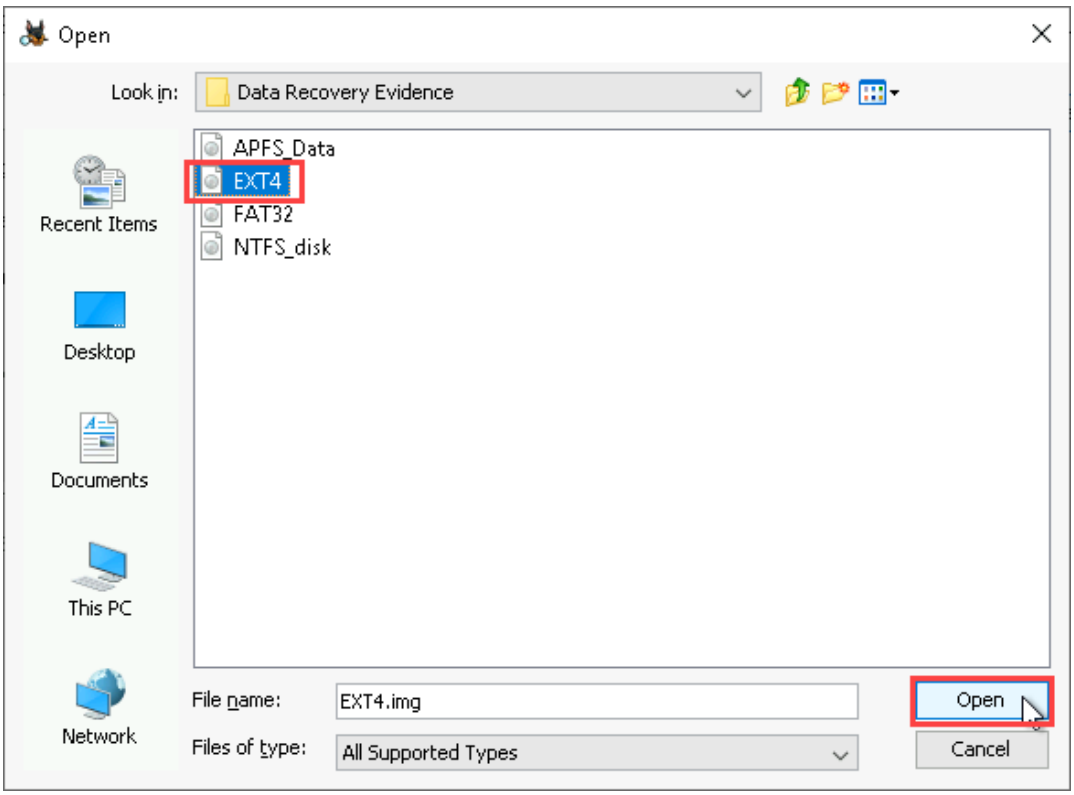
8. On the Select Data Source page, **click Browse** to open the Browse dialog box.



Browse button

9. In the Browse dialog box, **navigate** to **This PC > Local Disk (C:) > Data Recovery**

Evidence and **select** the **EXT4.img file**, then **click Open**.

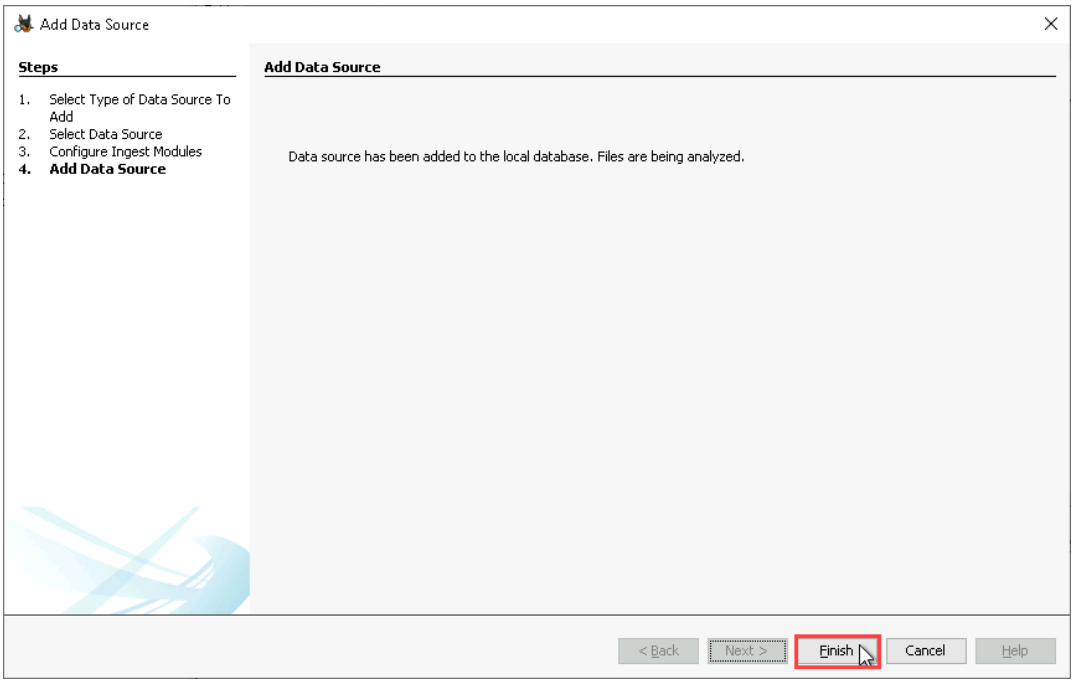


Browse

10. On the Select Data Source page, **click Next** to continue.

11. On the Configure Ingest Modules page, **click Next** to accept the default options and open the Add Data Source page.

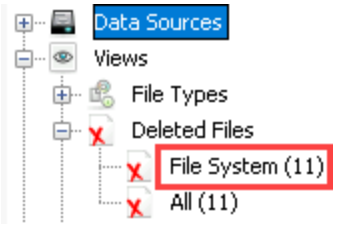
12. On the Add Data Source page, **click Finish** to close the wizard.



Add Data Source

Note: Like E3 and NTFS, Autopsy allows you to access deleted data from unallocated space on a drive image. When a drive image is acquired as evidence, Autopsy will automatically scan it for deleted files and attempt to recover them. Any deleted files that Autopsy is able to recover will be moved to the Deleted Files folder in the Tree Viewer, similar to the Trash folder in E3.

13. In the Tree Viewer, **expand** the **Deleted Files node**, then **select** the **File System node** to display the contents in the Result Viewer.

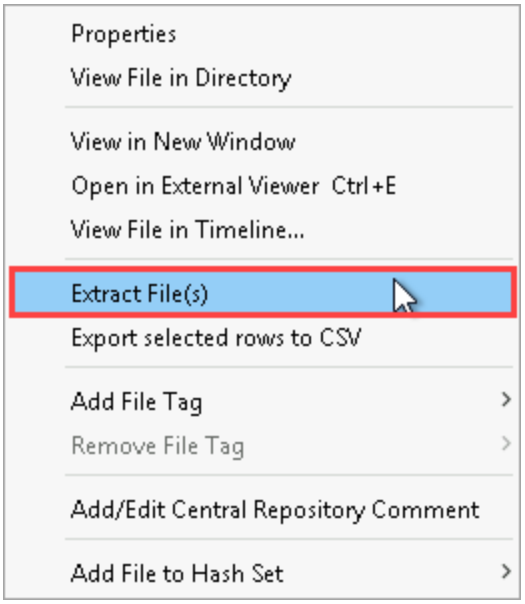


Tree Viewer

14. **Make a screen capture** showing the **contents of the list of deleted files in Autopsy**.

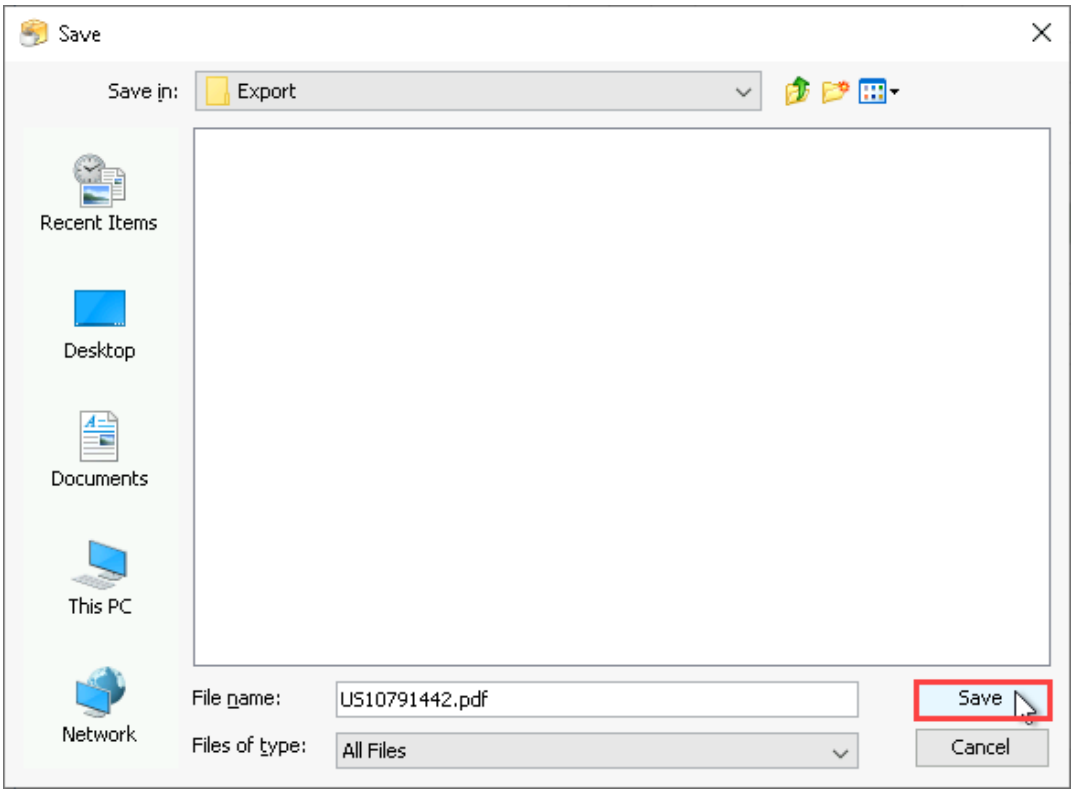
15. **Review** the recovered files in the Result Viewer to locate a patent file.

16. In the Result Viewer, **right-click** the **patent file** and **select Extract File(s)** from the context menu to open the Save window.



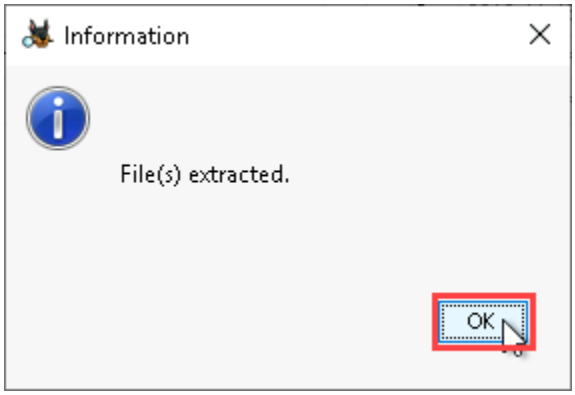
Extract File(s)

17. In the Save dialog box, **click Save** to save the patent file to the C:\yourname Data Recovery\Export folder.



Save window

17. When prompted, **click OK** to dismiss the Information dialog box.



Information dialog box

18. **Close** the **Autopsy window**.

19. From the taskbar, **click** the **File Explorer icon** to open a new File Explorer window.

20. In the File Explorer window, **navigate** to the **C:\yourname Data Recovery\Export folder**.

21. In the Export folder, **double-click** the **patent file** to open it in Adobe Reader.

22. **Make a screen capture** showing the **recovered patent file**.

23. **Close** any **open windows**.

Note: This concludes Section 1 of the lab.

Section 2: Applied Learning

Part 1: Recover Deleted Files in Linux with PhotoRec

Note: *Data carving* refers to the process of reassembling files from raw data on the disk, without the guidance of any file system metadata; that is, it is the removal of organized information from undifferentiated data. When a file system is corrupted, it may become inaccessible through the usual methods. This is often due to damage to the metadata stored about the file system, which contains information about its structure and the location of the various files within the directory tree. Without this information, the data on the drive is just a large jumble of raw unintelligible data. But there are ways to identify the organized chunks of data (files) in these jumbles, so long as you have the right tool.

One common approach to data carving is searching for certain file signatures, such as common headers and footers/trailers for specific filetypes, that indicate the beginning and ending of a file, which can then be used as cut (or 'carve') lines. For example, if you were to inspect any JPEG file with a hex editor (a program that allows you to represent any file in base-16, aka hexadecimal), you would see the characters FFD8FF at the beginning (header) and the characters FFD9 at the ending (footer/trailer), which may as well say "cut here." Such strings are commonly referred to as "magic numbers," and provide a way to uniquely identify specific file formats within raw data.

In this part of the lab, you will use PhotoRec, a popular data carving utility, to recover data from a corrupted hard drive. You will first access a Linux file system located on an external drive. You will then delete sensitive data on the drive and perform an intentional corruption of it. After confirming that the device is no longer mountable, you will then use PhotoRec to carve out the data you deleted from the now crippled file system.

1. On the Lab View toolbar, **select TargetLinux01** from the Virtual Machine menu to connect to the TargetLinux01 system.

2. From the TargetLinux01 toolbar, **open** a new **Terminal window**.

3. At the command prompt, **execute** `lsblk -p` to view information about the storage devices.

Note: The `lsblk` command lists information about all attached block (storage) devices. Adding the `-p` option will display the full path to each device discovered, such as /dev/sda1 instead of sda1. The /dev folder is a unique directory that consists mostly of block device files, which store data, and of character devices, which typically transfer data. This highlights a unique feature of Linux – everything is considered a file, including attached devices.

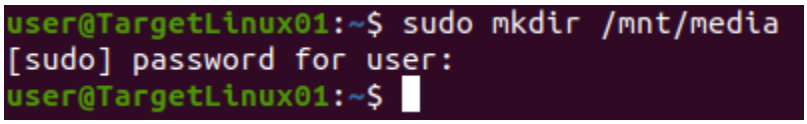
Linux follows a convention of labelling the first hard disk *sda*, the second *sdb*, and so on, adding numeral suffixes to indicate partitions. You can see the /dev/sda disk has two partitions mounted (made accessible from a place in your own file tree): the boot partition is mounted at /boot/efi, and the

file system is mounted at /. The second disk contains your target ext4 file system, which lives on /dev/sdb2. You will mount this partition in the following steps.

The loop devices can be ignored – these are special files used to mount images in Snap, a package deployment system found in Ubuntu distributions, the mechanics of which are beyond the scope of this lab.

4. At the command prompt, **execute** `sudo mkdir /mnt/media` to create a folder called media within the /mnt folder.

   When prompted, **type** `password` and **press Enter** to escalate your user privileges.



Create a folder

Note: The /mnt directory is commonly used to temporarily mount file systems. You are creating a directory within called *media*, which is where you will mount the target ext4 file system.

5. At the command prompt, **execute** `sudo mount /dev/sdb2 /mnt/media` to mount the file system.

Note: Mounting is made possible so long as that file system's metadata is intact. This file system metadata consists of characteristics such as the size, usage, and the location of files and directories on the system. In Linux, this record is known as the superblock.

A successful mount suggests there is no file system corruption, so you should now be able to access the target ext4 system from within your own file tree.

6. At the command prompt, **execute** `cd /mnt/media/home/ash` to access the user's home directory on the mounted ext4 file system.

7. At the command prompt, **execute** `ls` to display files in the user's directory.

Note: You should see a resume and a backup archive. The latter contains some 2FA recovery keys and a server image, as you will find in the next step.

8. At the command prompt, **execute** `rar l backups.rar` to view the contents of the RAR archive.

```
user@TargetLinux01:/mnt/media/home/ash$ rar l backups.rar

RAR 5.50   Copyright (c) 1993-2017 Alexander Roshal   11 Aug 2017
Trial version             Type 'rar -?' for help

Archive: backups.rar
Details: RAR 5

 Attributes      Size     Date    Time   Name
----------- ---------  ---------- -----  ----
 -rw-------         0  2021-08-20 14:19  backups/.recoveryKeys.txt.swp
 -rw-------         0  2021-08-20 14:20  backups/.recoveryKeys.txt.swo
 -rw-r--r--  20000000  2021-08-20 14:22  backups/serverImage.dd
 -rw-r--r--       257  2021-08-20 14:21  backups/recoveryKeys.txt
----------- ---------  ---------- -----  ----
             20000257                    4

user@TargetLinux01:/mnt/media/home/ash$
```

View backups.rar contents

9. **Make a screen capture** showing the **contents of the RAR archive in the /mnt/media/home/ash directory**.

10. At the command prompt, **execute** `rm -f *` to force (`-f`) removal of all files (`*`) in the current directory.

11. At the command prompt, **execute** `cd ~` to return to your home directory, so that you are no longer in the mounted ext4 file system.

Note: In the next steps, you will deliberately corrupt the file system by writing random raw data to it. This simulates a scenario where the drive has been damaged, such that the superblock is in no shape to facilitate any mounting. Being unable to mount the directory, and without the guidance from the file system metadata contained in the superblock, you will need to resort to data carving to recover your files.

12. At the command prompt, **execute** `sudo umount /dev/sdb2` to unmount the drive.

13. At the command prompt, **execute** `sudo dd if=/dev/urandom of=/dev/sdb2 bs=1k seek=200 count=4k` to corrupt the file system.

```
user@TargetLinux01:~$ sudo dd if=/dev/urandom of=/dev/sdb2 bs=1k seek=200 count
=4k
4096+0 records in
4096+0 records out
4194304 bytes (4.2 MB, 4.0 MiB) copied, 0.248133 s, 16.9 MB/s
user@TargetLinux01:~$
```

dd command

Note: This command uses a versatile utility called dd, or disk destroyer, to write random data to a section of the file system. The file used to generate input (**`if=`**) is /dev/urandom, a character device in the /dev/folder. This urandom device uses the noise produced by operations in the OS to generate pseudo-random numbers, and in your case, is being used to make a complete mess of the partition, which you've designated as the output file (**`of=/dev/sdb2`**). You have also specified to write in 1kB blocks (**`bs=`**), and to write 4000 (**`count=4k`**) of these blocks. Finally, you have specified where to start writing, which is immediately following the first 200 blocks (**`seek=200`**).

14. At the command prompt, **execute** `sudo mount /dev/sdb2 /mnt/media` to remount the drive.

Note: You should receive an error message indicating the superblock is damaged ("bad superblock").

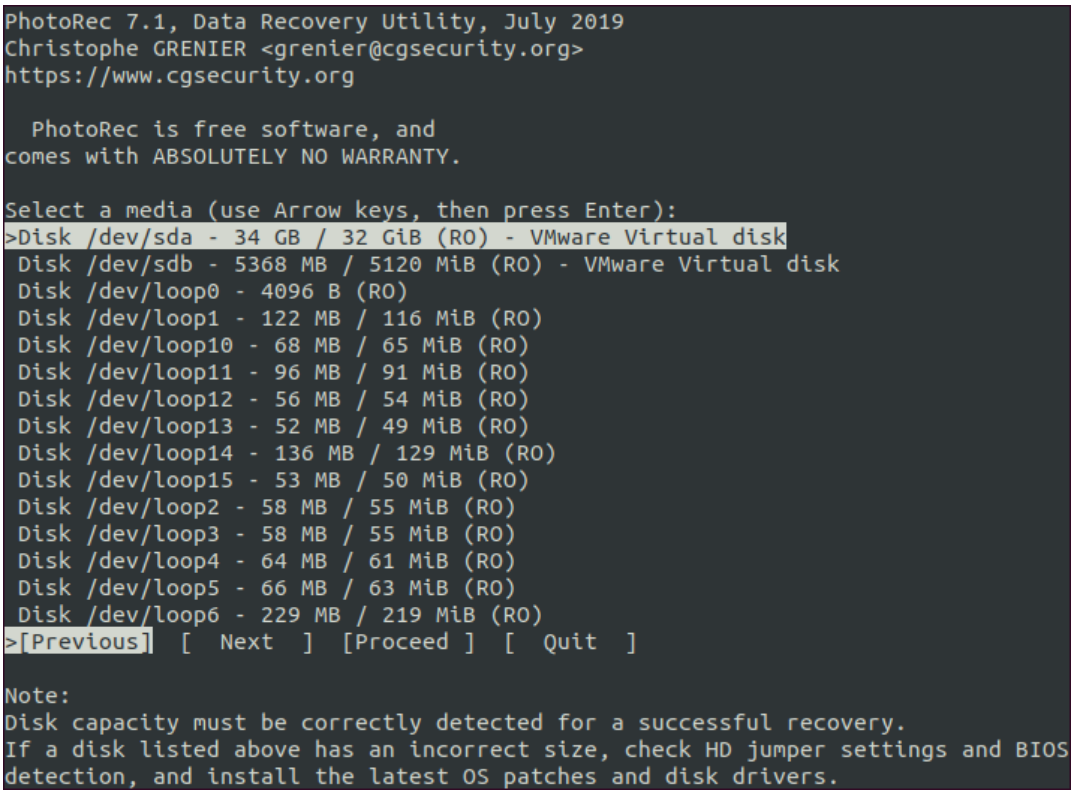
The file system metadata cannot be read, so you may either attempt to repair the file system, or to perform a recovery of certain data on it. Here it is presumed a file system repair was unsuccessful, so carving is now necessary.

15. **Make a screen capture** showing the **failed mount attempt on the /dev/sdb2 device**.

16. At the command prompt, **execute** `sudo apt-get install testdisk` to download and install the TestDisk utility suite.

Note: The TestDisk package contains two tools, one of which is TestDisk itself. TestDisk can recover both damaged/deleted storage partitions and specific files (so long as that file system is still intact). However, in the event the file system metadata is corrupted, and TestDisk is unable to repair the damage itself, its companion utility, PhotoRec, can be invoked to carve the device without such guidance. PhotoRec is a signature-based data-carving tool that ignores the file system and instead reaches for the underlying data directly. This means it is able to recover files even when the file system is severely damaged.

17. At the command prompt, **execute** `sudo photorec` to launch the data carving application contained within the TestDisk package.



PhotoRec interface

Note: Navigating the PhotoRec interface is easy once you understand the layout. All items for the current page are displayed vertically, and can be selected using the up/down arrow keys. The bottom menu is traversable using the left/right arrow keys, and acts much like the buttons in a typical graphical install wizard (Back, Next, Abort/Cancel, maybe More Options). In this context, the Quit button behaves more like a back button in a web browser, returning you to the previous page in the process, unless you are on the first stage (as you are presently), in which case it will exit and terminate the program.

You are greeted with an array of drives – optionally, you could have specified one with the initial command execution.

18. At the Select a media menu, **select** the **/dev/sdb disk**, then **press Enter** to continue.

19. From the Partition menu, **select** the **Linux partition**.

20. From the bottom menu, **select File Opt**, then **press Enter** to continue.

21. At the filetype selection page, **press s** to deselect all file families.

Note: PhotoRec can search for around 500 different filetypes. All are selected by default, but you can select any combination to perform a more targeted search, as you will do in the next steps. Because you are in search of a resume created in Open Office (.odt) and some backups in a RAR archive (.rar), you will select only the file families that contain these filetypes.

22. Use the arrow keys and spacebar to **select** the **rar** and **zip** file families from the list.

Note: Open Office documents are actually compressed files that fall within the zip file family category in PhotoRec.

Once both file families are selected, proceed to the next step.

23. **Press b** to save your selections, then **press Enter** to continue.

24. **Press q** to return to the Partition menu.

25. From the bottom menu, **select Search**, then **press Enter** to continue.

Note: The next menu requires you to specify the file system on which your targeted files are stored. While you know the file system on the sdb2 partition is ext4, it is also the most common file system used for Linux, so it is typically a safe bet. Infrequently, you may also see XFS or ReiserFS, in which case you would select the Other option.

26. At the Filesystem menu, **press Enter** to accept the default (ext2/ex3) and continue.

27. At the Space to be analysed menu, **press Enter** to accept the default (Free) and continue.

Note: Electing to scan only the free space is adequate for your purposes. You are only interested in deleted files, so can expedite your approach by ignoring all allocated space on disk.
The next menu requires you specify a place to deposit the files. It also admonishes you to not deposit your recovered files to the same media from which you recovered them, which is good advice. To this end, you will use the /home/users/Documents/ directory on your live file system to store your rescues. PhotoRec places you in the same directory you ran the program from, so you will presently find yourself in /home/user, indicated by the Directory line immediately above the file list.

You will now select the Documents directory and begin the recovery process.

28. From the Destination menu, **select** the **Documents folder** and **press Enter**, then **press c** to begin carving the file system.

Note: As the program processes blocks of data, it will report in real-time the type and quantity of files it has discovered.
This process should take around 3 minutes. Once it has finished, you will see the message *Recovery completed*, after which you may continue to the next step. You will also notice the directory where the recovered data has been saved. You will navigate there in the following steps in search of your compressed files: the .odt resume and the .rar backup.

29. At the Recovery completed page, **press q twice** to exit the PhotoRec program and return to the command prompt.

30. At the command prompt, **execute** `cd Documents/recup_dir.1` to navigate to the folder containing all data recovered by PhotoRec.

31. At the command prompt, **execute** `ls` to display all recovered files.

32. **Make a screen capture** showing the **compressed files recovered by PhotoRec**.

Note: Although the names have not been recovered (one of many casualties of the corruption), it appears the target files have. For some reassurance, you will extract the data from the RAR archive and list the contents in the console to finish out this section.

33. At the command prompt, **execute** `sudo rar e nameofrar` to extract the contents of the RAR archive.

34. At the command prompt, **execute** `ls` to display the recovered backup files.

Note: During a real-world recovery operation, you would want to conduct further verification of the files to completely ensure their integrity. For example, you could compare these recovered files against metrics taken from the originals, such as their length or maybe a hash generated beforehand. However, for the purposes of this lab, you can consider this mission successful.

35. **Make a screen capture** showing the **backup files recovered from the RAR archive**.

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: Recover Deleted Files from a FAT Drive Image

Now that you've had some practice with recovering deleted files directly from live systems, you will resume the investigation of intellectual property theft that you began in Section 1. In the first phase of your investigation, you discovered multiple deleted files and potentially incriminating evidence on two drive images. This morning, the other two drive images were transferred to you. Your supervisor has asked you to apply what you have learned so far to discover further incriminating evidence on the two remaining drive images.

The first drive image is from a Windows system, but uses the older FAT32 file system. FAT is short for File Allocation Table, and traces its roots back to Windows 95 when it was introduced as a replacement to the even older FAT16. While FAT32 has long since been replaced by NTFS as the default file system on Windows desktops, servers, and laptops — due in no small part to a weak permissions systems — it is still commonly used as the default file system on flash drives because of its compatibility with a wide range of devices.

Using E3, import the FAT32.img drive image located in the Data Recovery Evidence folder and attempt to recover at least one additional patent file.

Make a screen capture showing the **patent file recovered from the FAT32 drive image within E3**.

Part 2: Recover Deleted Files from a APFS Drive Image

The second additional drive image was taken from a computer running Mac OS. Beginning with the release of OS X in 2001, Apple's Mac OS operating system has been based on the UNIX operating system, which means that Mac OS shares a great deal of common DNA with Linux. However, while Mac OS contains some open source components, much of the source code is closed source and proprietary, including the file system.

From 1985 through 1998, new Apple computers used a proprietary file system called HFS (Hierarchical File System). With the release of Mac OS 8.1 in 1998, Apple began transitioning to a new version of HFS called HFS Plus (or HFS Extended). HFS Plus was finally replaced in 2017 when Apple introduced APFS (Apple File System) with the release of Mac OS 10.12.4 (High Sierra). Unlike HFS and HFS Plus, which were originally developed for the era of floppy disks, APFS is optimized for solid-state drive storage and also supports encryption, snapshots, and increased data integrity.

Using Autopsy, import the APFS.img drive image located in the Data Recovery Evidence folder and attempt to recover at least one additional patent file.

Make a screen capture showing the **patent file recovered from the APFS drive image within Autopsy**.

Note: This concludes Section 3 of the lab.