

# Introducing the Autopsy Logical Imager

Ann Priestman  
Basis Technology

# Introduction

---

- What is Logical Imager:
  - Command line program that runs on target computer and collects:
    - Files of interest
    - System information (such as users)
  - Results can be easily imported into Autopsy as a data source
- Use Cases:
  - Search warrant at an organization with a large number of computers. Use to collect basic information from each computer and then decide which to fully image first.
  - Collect only a subset of folders (such as for a specific user) that you have consent for.
  - Time-sensitive situation where you want to prioritize what files get collected.

# Benefits Over Other Tools

---

- Parses raw drive data using The Sleuth Kit:
  - Can access locked files and bypass rootkits that hide files
  - Does not update time stamps during search
  - Can access dual boot volumes (Linux for example)
- Can create a full image if you keep it plugged in long enough.
- Tightly integrated with Autopsy to configure and review the results.

# Requirements

---

- You need to configure logical imager from a Windows computer.
- The target machine must be running Windows.
- You need to be able to run with administrator rights on the target machine.
- You must have an NTFS or ExFAT external drive to run from.

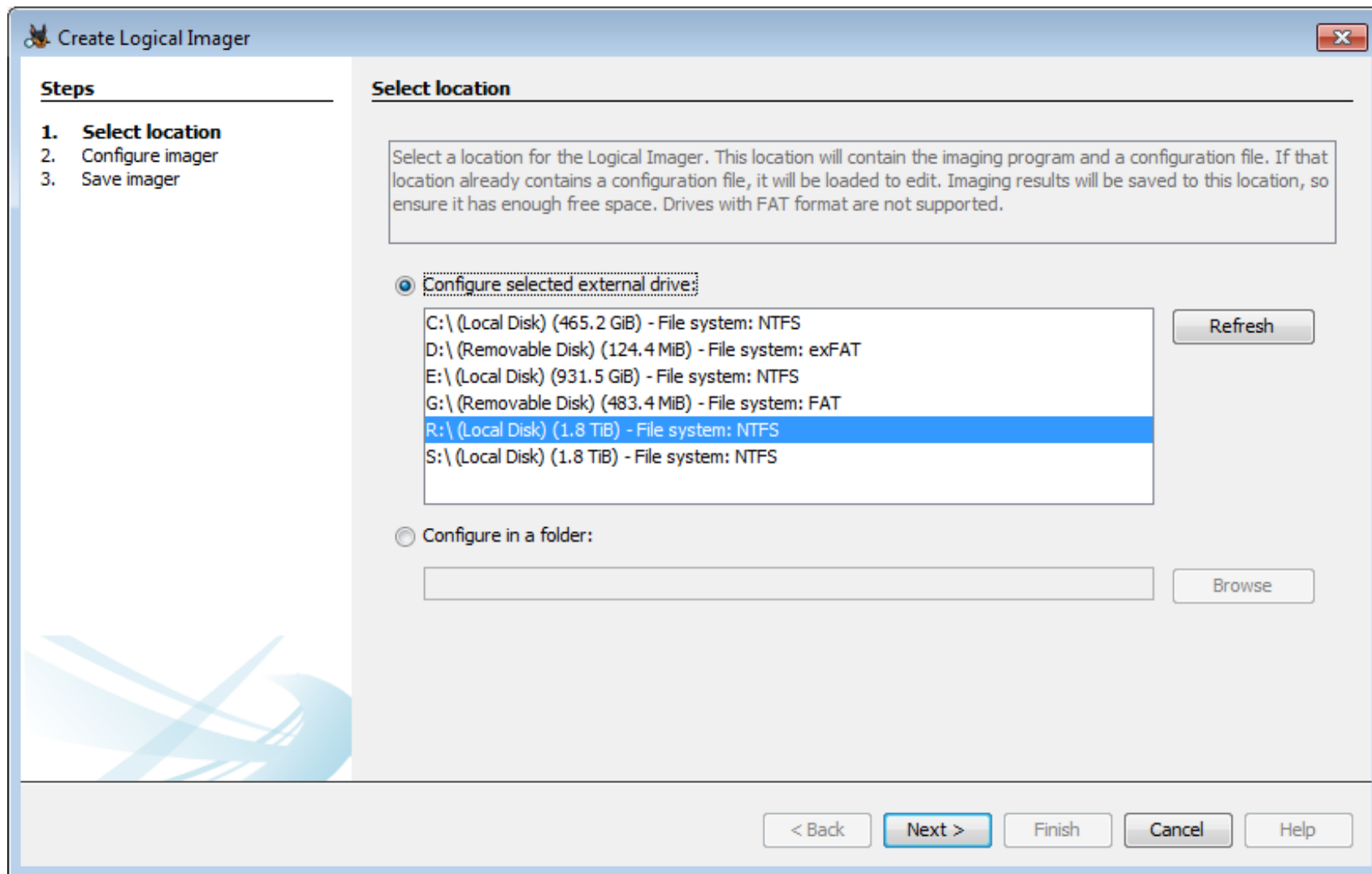
# Configuring Logical Imager

# Overview

---

- General configuration process:
  - Go to Tools->Create Logical Imager
  - Pick a USB drive to configure
  - Create a set of rules
  - Choose your global settings
  - Save your configuration and the logical imager executable to the drive selected

# Selecting a Drive



# Main Configuration Panel

**Steps**

1. Select location
2. **Configure imager**
3. Save imager

**Configure imager**

Configuration rule file: R:\logical-imager-config.json

Rule Name	Description
Downloaded archives	Archive files in the user downlo...
Encryption Programs	Find encryption programs
Large recent files	Files over 500 MB that were cha...

Rule name: Downloaded archives

Description: \rchive files in the user downloads folder

Extensions: zip,rar

File names:

Folder names: [USER\_FOLDER]/Downloads

Full paths:

File size in bytes: Minimum: Maximum:

Modified Within: day(s)

Extract file

Alert in imager console

Alert if encryption programs are found

Create VHD

Continue imaging after searches are performed

Prompt before exiting imager

New Rule Edit Rule Delete Rule

< Back Next > Finish Cancel Help



# Main Configuration Panel

**Steps**

1. Select location
2. **Configure imager**
3. Save imager

**Configure imager**

Configuration rule file: R:\logical-imager-config.json

Rule Name	Description
Downloaded archives	Archive files in the user downlo...
Encryption Programs	Find encryption programs
Large recent files	Files over 500 MB that were cha...

**Rule Details**

Rule name: Downloaded archives  
Description: Archive files in the user downloads folder  
Extensions: zip,rar  
File names:  
Folder names: [USER\_FOLDER]/Downloads  
Full paths:  
File size in bytes: Minimum: Maximum:  
Modified Within: day(s)  
 Extract file  
 Alert in imager console

**Settings**

Alert if encryption programs are found  
 Create VHD  
 Continue imaging after searches are performed  
 Prompt before exiting imager

New Rule Edit Rule Delete Rule

< Back Next > Finish Cancel Help

# Making Rules

- Click on “New Rule” to create a new rule.
- You can either make a rule based on file attributes or to make a rule based on the full path to the file

New Rule

Choose the type of rule  
Attribute Search for files based on one or more attributes or metadata fields.

Rule name: Downloaded archives

Description (Optional): Archive files in the user downloads folder

Extensions: zip,rar  
Extensions are case insensitive.

File names: Example: filename.txt, readme.txt  
File names are case insensitive.

Folder names: [USER\_FOLDER]/Downloads  
Starting a folder name with the token [USER\_FOLDER] will allow matches of all user folders in the file system. Folder name matches are case insensitive and occur anywhere in a path.

Minimum size: Bytes

Maximum size: Bytes

Modified within: day(s)

If file is found:  
 Extract file  
 Alert in imager console

OK Cancel

# Attribute Rules

- Similar to Interesting File rule sets – you can match by extension, file name, path, size, and last modified date.
- Select “Attribute” in the combo box at the top to make an attribute rule
- Enter a rule name and optional description

Choose the type of rule

Attribute Search for files based on one or more attributes or metadata fields.

Rule name: Downloaded archives

Description (Optional): Archive files in the user downloads folder

# Attribute Rules – Extension and Name

- You can enter any number of extensions or file names.
- Note that each file name should include its extension, so you can not specify both exceptions and file names.

Extensions:

**i** Extensions are case insensitive.

File names:

**i** File names are case insensitive.

# Attribute Rules – Folder Name

- You can enter any number of folder names.
- The folder names can appear anywhere in the path
- You can use “[USER\_FOLDER]” to match Windows or Linux user folders.

Folder names:

**i** Starting a folder name with the token [USER\_FOLDER] will allow matches of all user folders in the file system.  
Folder name matches are case insensitive and occur anywhere in a path.

# Attribute Rules – File Size and Date

- You can specify a minimum and/or maximum size
- You can also require that the file was modified in the last X days

<input checked="" type="checkbox"/> Minimum size:	<input type="text" value="1,000"/>	<input type="text" value="Bytes"/>
<input checked="" type="checkbox"/> Maximum size:	<input type="text" value="50"/>	<input type="text" value="Megabytes"/>
<input checked="" type="checkbox"/> Modified within:	<input type="text" value="7"/>	day(s)

# Full Path Rules

- File must exactly match the name and path given
- Multiple paths can be entered on separate lines

New rule

Choose the type of rule

Full Path Search for files based on full exact match path.

Rule name: Notepad

Description (Optional):

Full paths: /Windows/System32/notepad.exe

If file is found:

Extract file

Alert in imager console

OK Cancel

# All Rules – Choose Action for Match

---

- Extract the contents of the file
- Write an alert to the imager console that a match was found
  - Best for cases where few matches are expected

If file is found:

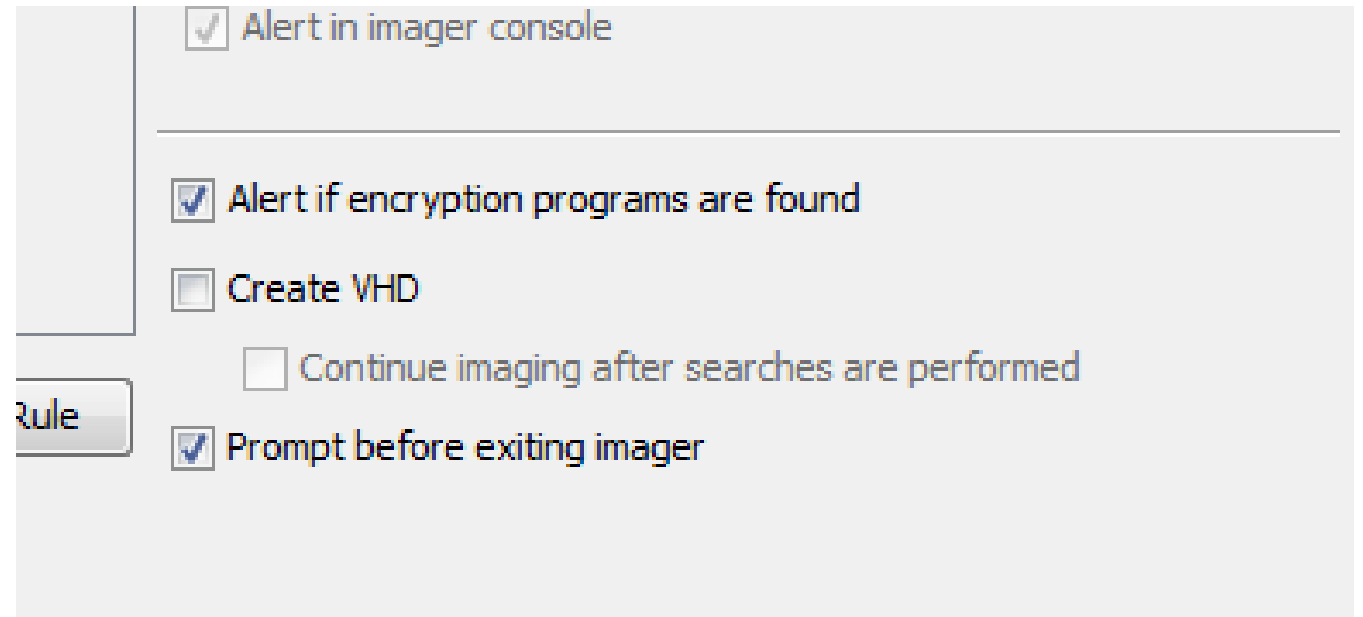
Extract file

Alert in imager console



# Settings – Encryption and Console Setting

- Alert if encryption programs are found
  - Preset rule that looks for “truecrypt.exe”, “VeraCrypt.exe”, etc.
- Prompt before exiting imager
  - Keeps the console window open so you can quickly see any alerts or error messages.



# Settings - Output Format

---

- Default – Each matching file will be saved to a folder
- Create VHD – All blocks read by logical imager will be written to a sparse VHD, which will include the full contents of all matching files
  - Continue imaging after searches are performed – When the search is done, continue to acquire blocks until the program is terminated.

Create VHD

Continue imaging after searches are performed

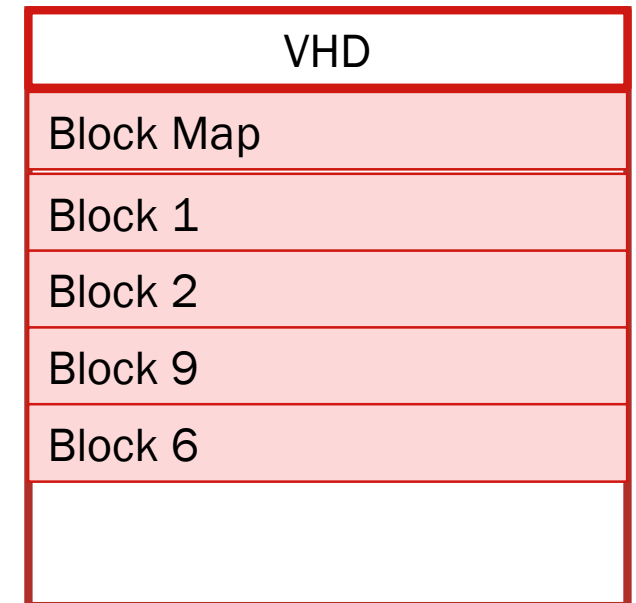
# Sparse VHD

---

- A Virtual Hard Disk (VHD) is a file representing a hard disk.
- A Sparse VHD is a variation where:
  - Blocks may occur in any order (the header maps original offsets to offsets in the VHD)
  - Any blocks that aren't present are interpreted as all null bytes
- Since unused blocks are not included, sparse VHDs can be much smaller than the hard drive they represent.

# Sparse VHD

- Each block read by logical imager will be written to the VHD.
- Since logical imager will read the metadata for every file on the system, metadata will also be present in the VHD.
  - The full partition tables and master file table(s) will be included
- Logical imager will read the complete data for any matching files, so they will be copied to the VHD in full.
- Will optionally continue filling in any missing blocks after the search is complete.
- In testing, the sparse VHDs created were typically around 10% of the full image size after the search completed.



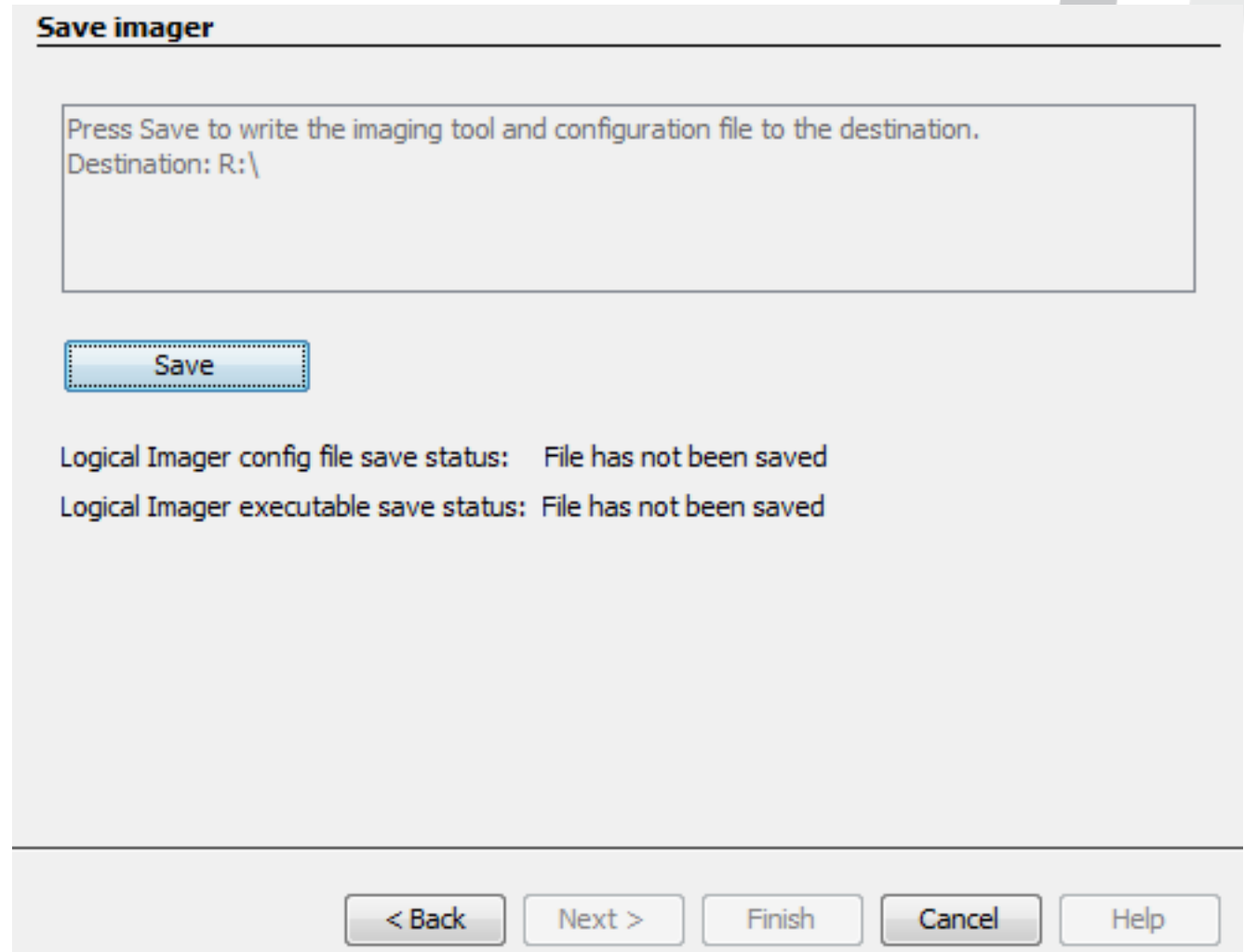
# Extracting Files vs. Using a VHD

---

- File mode
  - Pros: Typically much faster and will use significantly less disk space
  - Cons: No data about the file system or non-matching files is saved. Not all file metadata is preserved.
- VHD mode
  - Pros: Can be used to make a full copy of the drive. Has metadata about all files, which enables for more post-processing analytics (such as prioritization).
  - Cons: Typically much slower and uses more disk space. Can also be more confusing in Autopsy since files that were not copied will appear in the tree

# Saving

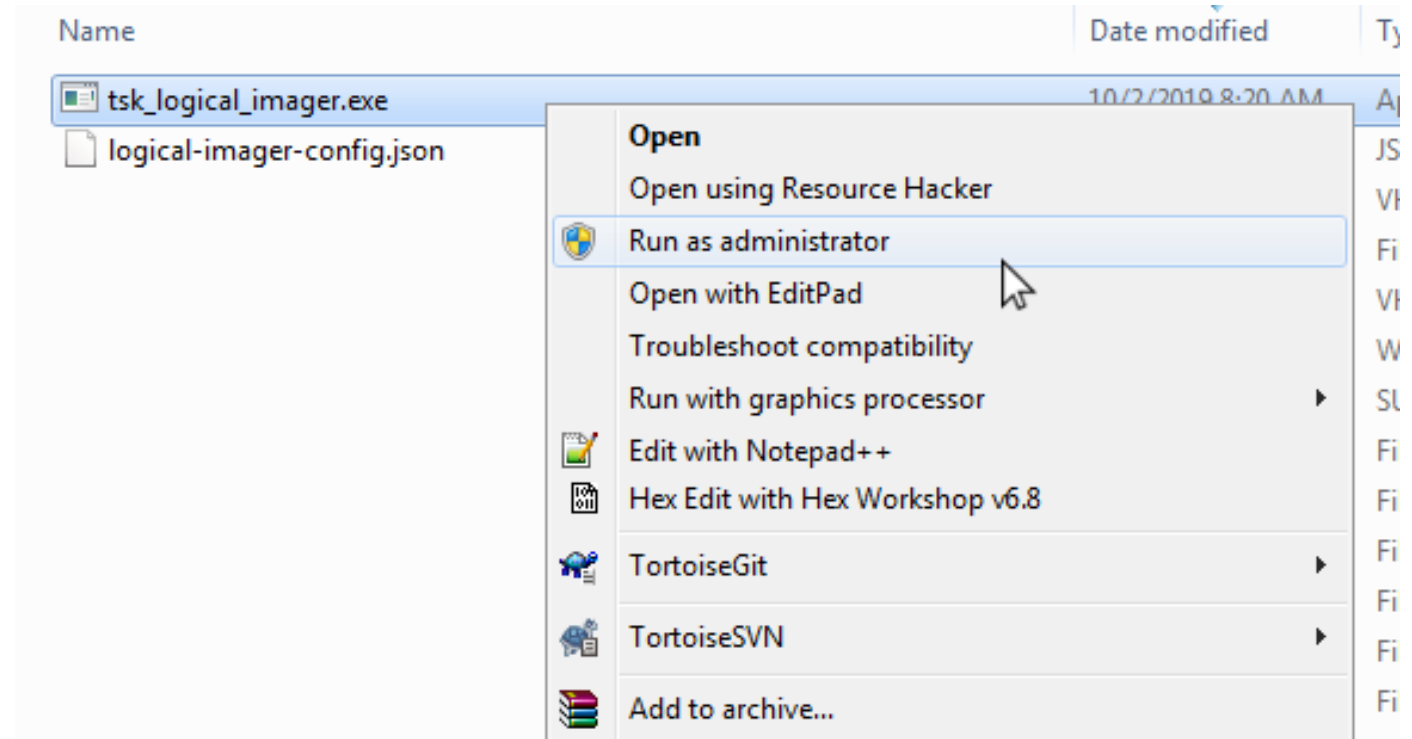
- Executable:
  - tsk\_logical\_imager.exe
- Configuration file:
  - logical\_imager\_config.json (default)



# Running Logical Imager

# Launching Logical Imager

- Insert the drive you configured into the target computer
- Right-click to run `tsk_logical_imager.exe` as Administrator
- You can also run from an elevated command prompt



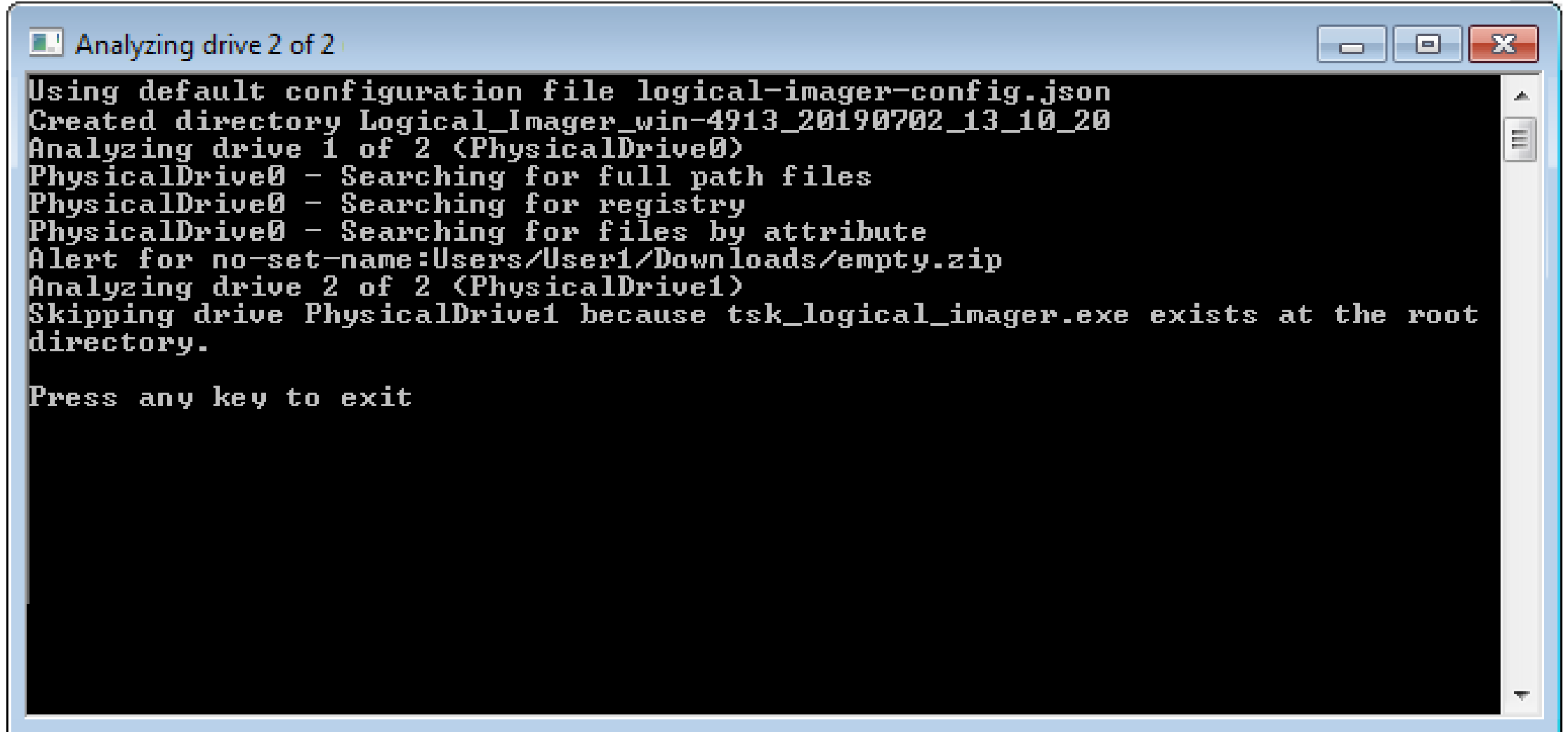


# What Will Happen

---

- All “Physical Drives” will be analyzed to identify file systems.
- If a drive is encrypted, then “Logical Drives” (such as \\.\C:) will be used.
- For each file system:
  - Searches will be conducted for full path-based rules
  - Registry hives will be searched for and processed to identify users
  - All files will be scanned and evaluated against attribute-based rules

# Launching Logical Imager



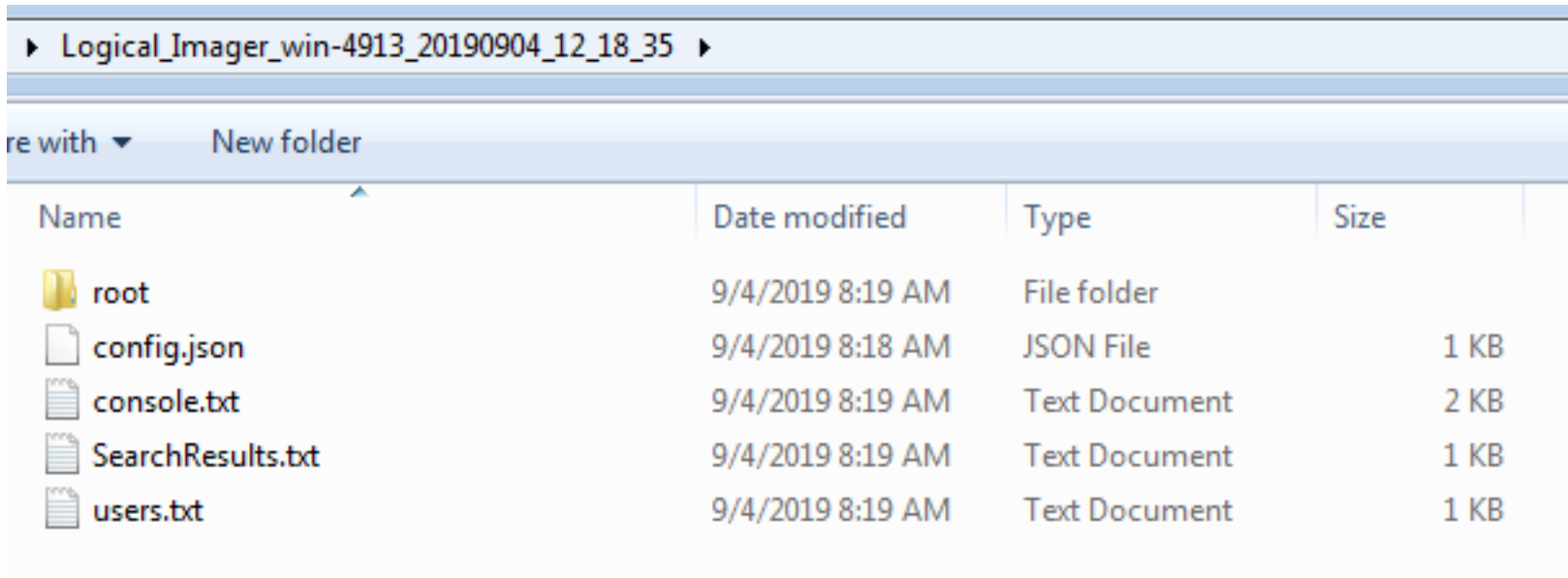
```
Analyzing drive 2 of 2
Using default configuration file logical-imager-config.json
Created directory Logical_Imager_win-4913_20190702_13_10_20
Analyzing drive 1 of 2 (PhysicalDrive0)
PhysicalDrive0 - Searching for full path files
PhysicalDrive0 - Searching for registry
PhysicalDrive0 - Searching for files by attribute
Alert for no-set-name:Users\User1\Downloads\empty.zip
Analyzing drive 2 of 2 (PhysicalDrive1)
Skipping drive PhysicalDrive1 because tsk_logical_imager.exe exists at the root
directory.

Press any key to exit
```

# Viewing Results

# Output Folders

- The results will be in a folder next to the logical imager executable
- Non-VHD runs will contain all exported files under the “root” folder
- VHD runs will contain one or more .vhd images instead of the “root” folder



Name	Date modified	Type	Size
root	9/4/2019 8:19 AM	File folder	
config.json	9/4/2019 8:18 AM	JSON File	1 KB
console.txt	9/4/2019 8:19 AM	Text Document	2 KB
SearchResults.txt	9/4/2019 8:19 AM	Text Document	1 KB
users.txt	9/4/2019 8:19 AM	Text Document	1 KB

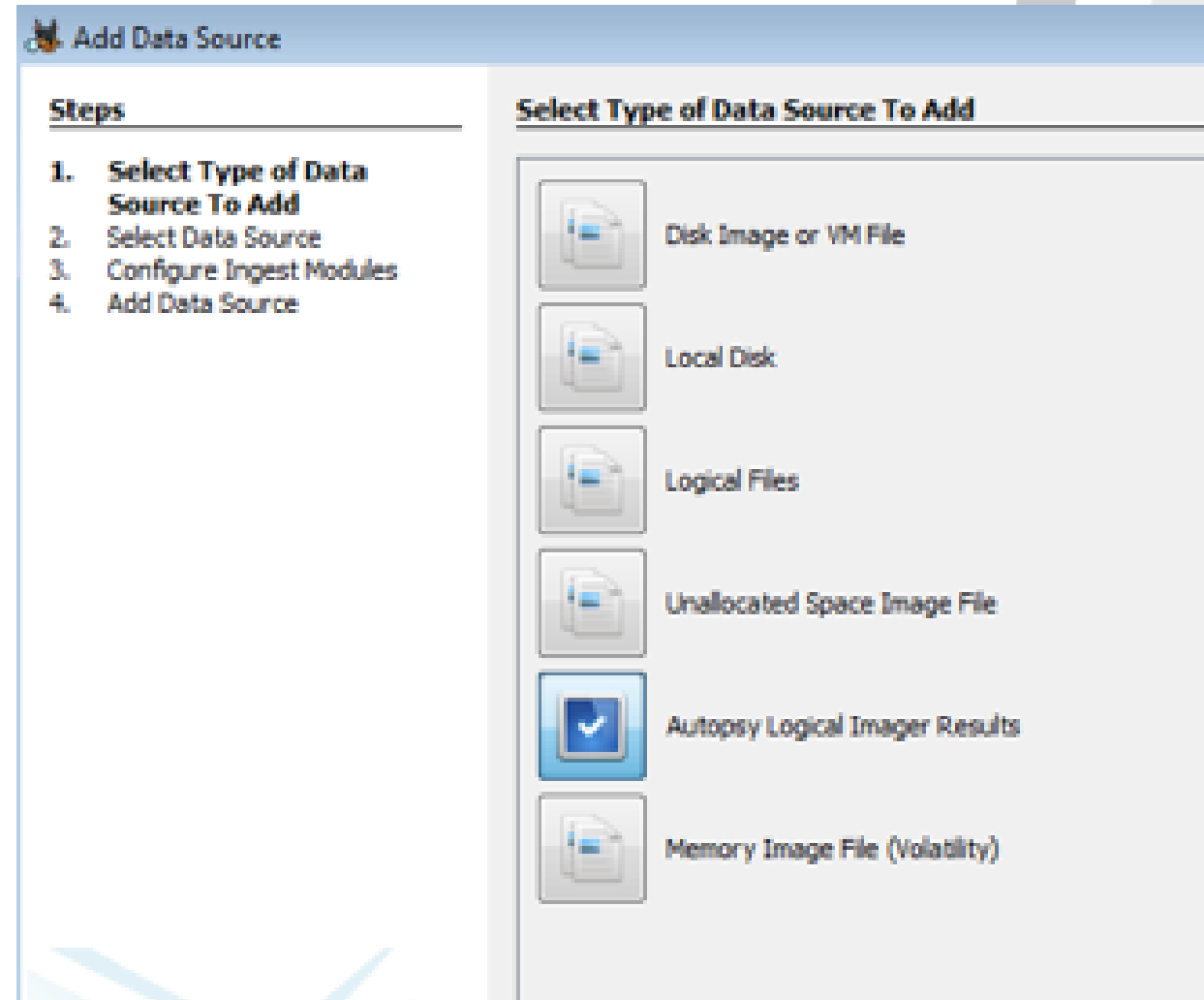
# Users.txt

- Contains user information from the registry

	A	B	C	D	E	F	G	H
1	LOCAL USER ACCOUNTS ONLY							
2								
3	UserName	FullName	UserDomain	HomeDir	AccountType	AdminPriv	DateCreated	LastLoginDate
4	Administrator		local		Regular	Yes	2016-06-06T03:55:48.000000000Z	2016-04-24T14:54:42.0
5	Guest		local		Limited	No	2016-06-06T03:55:48.000000000Z	Unknown
6	DefaultAccount		local		Regular	No	2016-06-06T03:55:48.000000000Z	Unknown
7	WDAGUtilityAccount		local		Regular	No	2017-12-05T23:15:19.000000000Z	Unknown
8	User1		local		Regular	Yes	2016-06-05T12:59:52.000000000Z	2019-08-07T19:37:18.0
9								
10								
11								
12								
13								

# Viewing in Autopsy

- Use the “Autopsy Logical Imager Results” option to add your results to Autopsy.



# Viewing in Autopsy

- Select the logical imager drive and acquisition from the top, or manually browse to the folder.

Import From External Drive

Select Drive

C:\ (Local Disk) (465.2 GiB)  
G:\ (Removable Disk) (483.4 MiB)  
R:\ (Local Disk) (1.8 TiB)

Select acquisition from Drive R:\

Hostname	Extracted Date
win-4913	2019/06/12 13:13:28
win-4913	2019/06/12 13:14:48
win-4913	2019/06/14 11:44:01
win-4913	2019/06/16 23:30:59

Manually Choose Folder

Selected Folder: R:\Logical\_Imager\_win-4913\_20190612\_13\_13\_28

# Interesting Items

- Interesting Item results are created for each matching file.
- Double click on a result (or right-click and select “View Source File in Directory”) to move to the file’s location in the Data Sources section of the tree.

The screenshot shows a forensic analysis tool interface. On the left is a tree view with the following structure:

- Data Sources
  - PhysicalDrive2.vhd
- Views
  - File Types
  - Deleted Files
  - MB File Size
- Results
  - Extracted Content
  - Hashset Hits
  - E-Mail Messages
  - Interesting Items
    - Logical Imager results (2)
      - Interesting Files (2)
      - Interesting Results (0)
  - Accounts
- Tags
- Results

On the right, the 'Logical Imager results' table is displayed with the following data:

Source File	Category	File Path
empty.zip	Downloaded archives	/img_Phys
climbing.zip	Downloaded archives	/img_Phys



# Non-VHD Mode

- Files appear in their original folders.
- All files will be complete.
- No non-matching files or folders that do not contain extracted files will be present.

The screenshot displays a forensic analysis tool interface. On the left, a tree view shows the following structure:

- Data Sources
  - Logical\_Imager\_win--4913\_20191004\_
    - root (1)
      - PhysicalDrive2 (1)
        - Users (1)
          - User1 (1)
            - Downloads (2)

Below the tree view, there are sections for Views (File Types, Deleted Files, MB File Size) and Results (Extracted Content, Keyword Hits, Hashset Hits, E-Mail Messages, Interesting Items, Logical Imager results (2), Interesting Files (2), Interesting Results (0), Accounts).

On the right, a 'Listing' pane shows a table of files:

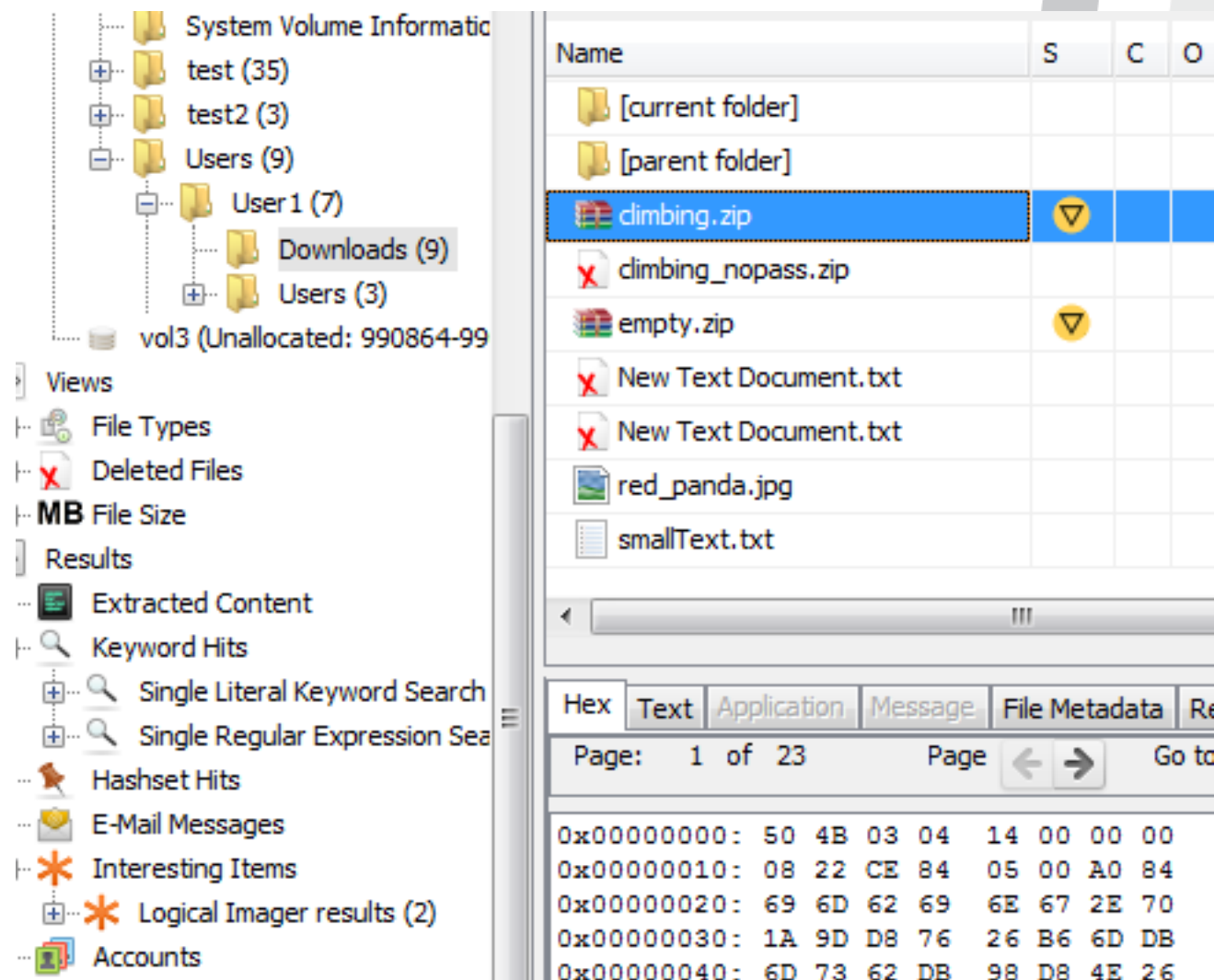
Name	S	C	O	Modified Time
dimbing.zip	▼	0		2018-03-21 08
empty.zip	▼	0		2019-07-01 13

Below the table is a hex view of the selected file:

Hex	Text	Application	Message	File Metadata	Result
Page: 1 of 23	Page	←	→	Go to Pag	
0x00000000:	50 4B 03 04	14 00 00 00	08		
0x00000010:	08 22 CE 84	05 00 A0 84	05		
0x00000020:	69 6D 62 69	6E 67 2E 70	6E		
0x00000030:	1A 9D D8 76	26 B6 6D DB	B6		
0x00000040:	6D 73 62 DB	98 D8 4E 26	B6		
0x00000050:	7A AD 7A AA	EE DA B5 51	FD		
0x00000060:	E2 C2 FE F8	F1 03 41 5A	4A		
0x00000070:	3F 40 6D A1	21 FF B5 2C	42		
0x00000080:	25 C5 7E 7C	FF 77 E6 77	97		

# VHD Mode – Extracted File

- Files appear in their original folders.
- Extracted files will be complete.
- Non-matching files and folders that do not contain extracted files will be present.



# VHD Mode – Non-extracted File

- Files that were not extracted will generally not contain data.
- The timestamps will be present.

The screenshot displays a forensic tool interface with a table of file metadata. The table has four columns: File Name, Date/Time, File Size, and another Date/Time. The 'red\_panda.jpg' row is highlighted in blue. Below the table is a navigation bar with tabs for 'Hex', 'Text', 'Application', 'Message', 'File Metadata', 'Results', 'Annotations', and 'Other Occurrences'. The 'Text' tab is active, showing a hex dump of the file's content, which consists of null bytes (0x00).

File Name	Date/Time	File Size	Another Date/Time
red_panda.jpg	2018-03-26 10:17:52 EDT	0000-00-00 00:00:00	2019-10-03 00:00:00 EDT
smallText.txt	2019-10-03 14:08:46 EDT	0000-00-00 00:00:00	2019-10-03 00:00:00 EDT

Page: 1 of 20    Page      Go to Page:     Jump to Offset     

```
0x00000000: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000010: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000020: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000030: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0x00000040: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
```

- It's best to use the Interesting File results to navigate to the extracted files

**Demo**