COMP-5350

Digital Forensics

FROM: Jacob Howard & Philip Julian

TO: Dr. Jason Cuneo & Jiaxiang Ren

DUE DATE: 10/15/21

Project 1

Overview

This project was designed to help us understand digital forensics and data recovery. We went through multiple senearious on how to recover deleted/modified files on different drive formats.

Part 1: Technical Analysis

Firstly, our goal was to find out how many partitions were on the drive given to us and what the partition types were. We can easily find this out by going to our fold containing the disk image through terminal and typing in the command "*fdisk -l Project1.dd*". This command will list all the partitons and types on the image.

After using the command, it was found that there were 4 partitons; 1 FAT16 partion, 1 FAT32 partiton, and 2 NTSF Partitions. Once we knew the types of drive paritions, we could hexdump at specific partition locations to read certain data off the drives.

For the first FAT16 partition, we saw that there had been 2 deleted files. This information was found in the file allocation table. By collecting data from the FAT, as seen below in Table 1, we were able to find the root directory location and data area location where the deleted files were stored. We were then able ultimately find the File locations of the 2 deleted files and use the "dd" command to copy the data. Two password protected zip files were found containing .jpg files. The images can be seen below in Image 1 and Image 2 while Table 2 shows all relative data for the partiton. The password used to unzip the files was found in a file in the first NTFS partition.

A	B	C	D	E
Description	Value	Structure	Start Location	Size
Sectors Before Partition	2048	Boot Sector	0x1C	4
Bytes/Sec	512	Boot Sector	0xB	2
Sec/Cluster	4	Boot Sector	0xD	1
Reserved Sectors	4	Boot Sector	0xE	2
Sec/FAT	200	Boot Sector	0x16	2
Root Directory Sectors	32	Root Directory		
Data Area Buffer	12	FAT		

Table 1

Filename	Ext	Status	Cluster Start (Hex)	Cluster Start (Dec)	# Clusters	# Sectors	File Size	File Size (Sectors)
CA256	ZIP	Deleted	5	5	23	92	45826	90
FC187	ZIP	Deleted	1C	28	-28	-112	38197	75
								0

	Allocated (Sectors)	Start	File Length (Sectors)
Sectors to Partition	2048	0	
Reserved Sectors	4	2048	
FAT #1 Length	200	2052	
FAT #2 Length	200	2252	
Root Directory Length	32	2452	
Data Area Buffer	12	2484	
File #1	92	2496	90
File #2	-112	2588	75
File #3	0	2476	0

Skip	Count	Confirmation Command
1277952	46080	hexdump -C -s $((2452*512)) -n $((32*512)) Project1.dd
1325056	38400	hexdump -C -s $((2452*512)) -n $((32*512)) Project1.dd
1267712	0	

Recovery Command
dd if=Project1.dd of=CA256.zip bs=512 skip=2496 count=90
dd if=Project1.dd of=FC187.zip bs=512 skip=2588 count=75

Table 2

Image 1

Image 2

Next, we dove into the first NTFS partition. This partition contained 2 deleted files. Using Active Disk Editor, we were able to pull the information we needed about where the files started and ended, so that we could copy them into our vm. The tables below show all the data collected, the Filenames and data, and the commands we used in the terminal to retrieve these files. The files contained an email with a password for unzipping the previously recovered zip files and an Encoding pdf file for encoding and decoding messages. The pdf was used to decode messages in from files in the second NTFS partition.

General NTFS Values				
Description	Value	Structure	Start Location	Size
Bytes/Sec	512	MBR	0xB	2
Sec/Cluster	8	MBR	0xC	1
Reserved Sectors	0	MBR	0xD	2
Sectors Before Partition	178176	MBR	0x1C	4
$MFT Cluster Start	4	MBR	0x30	8
$MFTMirr Cluster Start	14079	MBR	0x38	8
# System $MFT Records	43	MFT		
$MFT Record Size	2	MFT		

General NTFS Values

NTFS Data Stucture Locations		
	Allocated (Sectors)	Start
Sectors to Partition	178176	0
$MFTMirr Start	112632	290808
$MFT Cluster Start	32	
$MFT System Records	86	178208
File #1 $MFT Record	2	178294
File #2 $MFT Record	2	178296

NTFS Data Structure Locations

Info For files: Top row is Email.docx info and bottom row is for Encoding.pdf

					NTFS $MFT Record I
Filename	Ext	Attributes	In Use (Header)	Non-Resident (0x10)	Allocated Size (x30)
Email	docx	$Standard Information x10 $Filename x30 $Filename x30 $Data x80	Yes	0	20480
Encoding	pdf	$Standard Information x10 $Filename x30 $Data x80	Yes	0 (No)	106496

Real Size (x80)	1st Cluster (x80 - 2)	1st Sector	1st Sector + Disk Offset	# Clusters (x80)	# Sectors	First VCN (x80)	Last VCN (x80)
16503	4958	39664	217840	5	32.23242	0	4
104632	4963	39704	217880	26	204.3594	0	25

Confirmation Command
hexdump ntfs.dd -s $((178294 *512)) -n $((1*512))
hexdump ntfs.dd -s $((178296*512)) -n $((1*512))
Recovery Command
dd if=Project1.dd of=Email.docx bs=512 skip=217840 count=33
dd if=Project1.dd of=Encoding.pdf bs=512 skip=217880 count=205

Commands used to Recover Files

Next, we went to the location of the FAT32 and reading data using active disk editor, we were able to find that there were 3 deleted files. All files were incripted in .gpg formant with a password. The data tables are shown below with all values for the FAT32 partition. We were able to use the dd command to recover these files. For file 1, the command was “dd if=Project1.dd of=CEB27.zip.gpg bs=512 skip=407431 count=101”. We just make the skip=Start and the count=File Length for all files on the drive. The Itinerary file contained the dates and times of a team for a heist and can be seen below.

Description	Value	Structure	Start Location	Size
Sectors Before Partition	405503	Boot Sector	0x1C	4
Bytes/Sec	512	Boot Sector	0xB	2
Sec/Cluster	1	Boot Sector	0xD	1
Reserved Sectors	32	Boot Sector	0xE	2
Sec/FAT	946	Boot Sector	0x16	2
Root Directory Sectors	1	Root Directory		
Data Area Buffer	3	FAT		

Filename	Ext	Status	Cluster Start (Hex)	Cluster Start (Dec)	# Clusters	# Sectors	File Size	File Size (Sectors)
CEB27.zip	gpg	Deleted	0x6	6	101	101	51481	101
Intructions.docx.gpg	gpg	Deleted	0x6B	107	FALSE	0	12493	25
Itinerary.xls.gpg	gpg	Deleted	0x84	132			7591	15

	Allocated (Sectors)	Start	File Length (Sectors)			
Sectors to Partition	405503	0				
Reserved Sectors	32	405503				
FAT #1 Length	946	405535				
FAT #2 Length	946	406481				
Root Directory Length	1	407427				
Data Area Buffer	3	407428		Skip	Count	Confirmation Command
File #1	101	407431	101	208604672	51712	hexdump -C -s $((407427*512)) -n $((1*512)) Project1.dd
File #2	25	407532	25	208656384	12800	same as above or using Active Disk editor
File #3	15	407557	15	208669184	7680	same as above or using Active Disk editor

Disk/File Information

	A	B	C	D	E	F
1		Time	Location	Event		
2	10/2/2021	8:00 AM	Paris, France	Meet Up With Team		
3	10/3/2021	8:00 AM - 10:00 PM	Paris, France	Gather Equipment Together		
4	10/4/2021	7:43 AM	Paris, France	Fly to New York		
5	10/4/2021	7:30 AM - 4:00 PM	New York	Drive to Heist Location		
6	10/6/2021	*SECRET*	*SECRET*	Set Up		
7	10/8/2021	*SECRET*	*SECRET*	Pay Day!		

Itinerary

Lastly, we dove into the last NTFS file. This was probaboly the most difficult partition to recover data, because even though we were able to see 3 files on the drive, the last 2 files were actually stored in the MFT, making it more difficult to find location and files size so that one could retreieve the data. We used the ordinary dd command for the first file, but for the other 2 files, we had to slightly alter it so that we could retrieve the files. The data tables and commands are shown below for the second NTFS partition. The Mystery.txt file contrained a hex format, that once decoded, gave the password to decript the previous .gpg files that were recovered. The ECC424 file contained a recon.txt file that had an encoded message in base64. The FC187.zip contained a location.jpg image, which is the assumed location of the heist.

General NTFS Values				
Description	Value	Structure	Start Location	Size
Bytes/Sec	512	MBR	0xB	2
Sec/Cluster	8	MBR	0xC	1
Reserved Sectors	0	MBR	0xD	2
Sectors Before Partition	732183	MBR	0x1C	4
$MFT Cluster Start	4	MBR	0x30	8
$MFTMirr Cluster Start	19262	MBR	0x38	8
# System $MFT Records	43	MFT		
$MFT Record Size	2	MFT		

Generic Values

NTFS Data Stucture Locations		
	Allocated (Sectors)	Start
Sectors to Partition	732183	0
$MFTMirr Start	154096	886279
$MFT Cluster Start	32	
$MFT System Records	86	732215
File #1 $MFT Record	2	732301
File #2 $MFT Record	2	732303
File #3 $MFT Record	2	732305
File #4 $MFT Record	2	732307
File #5 $MFT Record	2	732309

NTFS Data Structure Locations

NTFS $MFT Record Information													
Filename	Ext	Attributes	In Use (Header)	Non-Resident (0x10)	Allocated Size (x30)	Real Size (x80)	1st Cluster (x80 - 2)	1st Sector	1st Sector + Disk Offset	# Clusters (x80)	# Sectors	First VCN (x80)	Last VCN (x80)
DFA738	zip	$Standard Information x10 $Filename x30$ Data x80	No (0)	Yes	188416	185902	6255	50040	782223	5	363.0898	0	45
ECC424	zip	$Standard Information x10 $Filename x30 $Data x80	No (0)	Yes	358								
Mystery	txt	$10, $30, $80	No (0)	Yes	166								

NTFS $MFT Record Information

Confirmation Command
hexdump Project1.dd -s $((732301*512)) -n $((2*512))
hexdump ntfs.dd -s $((732303*512)) -n $((2*512))
hexdump ntfs.dd -s $((732305*512)) -n $((2*512))

Recovery Command
dd if=Project1.dd of=DFA738.zip bs=512 skip=782223 count=368
dd if=Project1.dd of=ECC424.zip bs=1 skip=374939424 count=358 iflag=skip_bytes,count_bytes
dd if=Project1.dd of=Mystery.txt bs=1 skip=374940448 count=166 iflag=skip_bytes,count_bytes

Commands

Location (from FC187.zip)

Part 2: Operational Analysis

In part 2, we were asked questions about the contents of the drives. The first question asked what methods were used to hide the files on the disk. The method for all files besides the last 2 files recovered in the second NTFS partion was file deletion. Since the files were marked as deleted but not overwriten, we were able to recover all those files. The second method, which the last two files in the NTFS partition used, was file manipulation to make it more difficult to recover the data.

The tools that were use to hide the data once recovered were encryption and password protection. The users seemed to want these files protected because their objective seemed to be a heist. They seemed to be interested in stealing from a specific location. One of the Base64 encoded messages, once decoded, gave a location as well. The location seemed to be the Smithsoniam Museum. The file this location was found, was from the Recon.txt file recovered and the exact decoded message is ”https://www.google.com/maps/d/u/0/viewer?msa=0&ie=UTF8&t=h&ll=38.89028798326893%2C-77.0293097175214&mid=1JEkfH9bJtMKrVCMHrKGPP_QmMys&z=17”