CAL POLY

California Cybersecurity Institute

Computer Forensics CCIC Training

Chapter 12: Hidden Data

Lauren Pixley, Cassidy Elwell, and James Poirier

May 2019 (Version 2)

12

Hidden Data

Introduction

There are several ways suspects will try to hide their data. A very common way suspects will usually hide their data is through encryption and passwords. However, they might also try to change file extensions or create hidden files and folders. Not every suspect will try to hide their data through these methods, but it is still important for you to be aware of the different data hiding methods. Once you know how some suspects might try to hide their data, you can learn how to better detect and handle passwords or renamed file extensions.

Passwords and Encryption

Towards the beginning of your analysis, you came across a password protected word document and ZIP file on Craig's desktop. Both file names had the word coupon in it, which means that these files could contain data relevant to your case. You are going to attempt to decrypt these files.

A good method to use when attempting to break a file's password is to follow a phased approach (see Figure 12-1). With a phased approach, you first try the easiest method, which is usually trying any known passwords, such as the user's login password. You can also try to run an English dictionary attack. If those both fail, you can index the suspect's drive and create a word list from the indexed words. The word list can then be used as a dictionary attack. Many criminals are lazy and will save passwords in places on their computer, reuse passwords for multiple accounts, and have physical versions like sticky notes and papers with passwords written on them. Be sure to check all of these sources when doing a real investigation before resorting to the brute force method. If the password is still not broken after all of these attempts, you can try a brute force attack.

The brute force attack method is usually a last resort. It goes through every possible combination of uppercase letters, lowercase letters, numbers, and symbols. A brute force attack will work eventually, but it takes a great deal of time and resources.

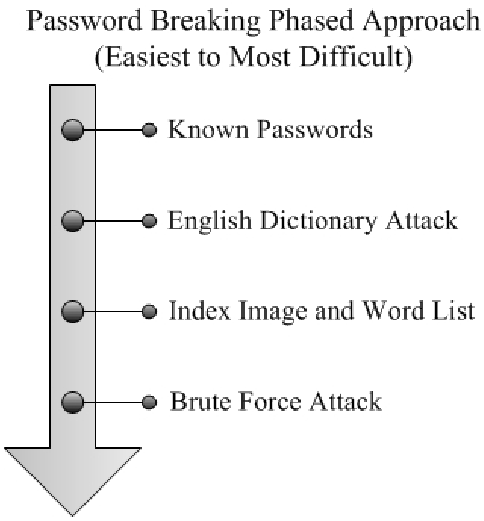
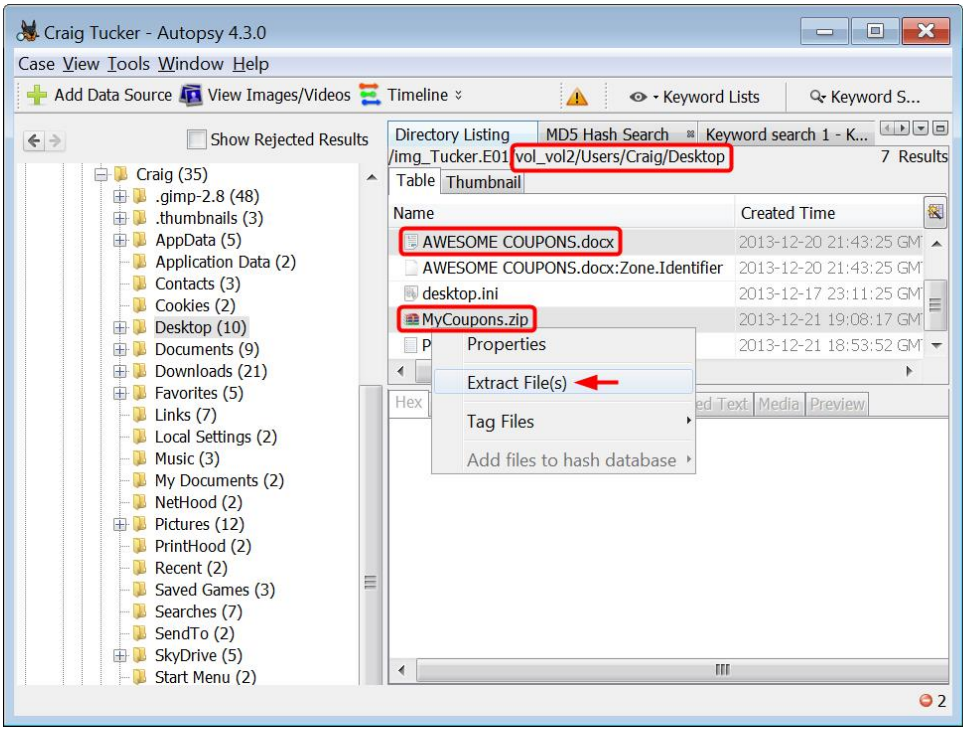

Password Breaking Phased Approach
(Easiest to Most Difficult)
Known Passwords
English Dictionary Attack
Index Image and Word List
Brute Force Attack

Figure 12-1 - Phased Approach to Breaking Passwords

First, you are going to try to use Craig's login password on the MyCoupons.zip and AWESOME COUPONS.docx. Export both these files out of Autopsy and save them into your Export folder.



Name	Created Time
AWESOME COUPONS.docx	2013-12-20 21:43:25 GM
AWESOME COUPONS.docx:Zone.Identifier	2013-12-20 21:43:25 GM
desktop.ini	2013-12-17 23:11:25 GM
MyCoupons.zip	2013-12-21 19:08:17 GM
P	2013-12-21 18:53:52 GM

Figure 12-2 – Extract MyCoupons.zip and AWESOME COUPONS.docx in Craig's Desktop Folder

Once MyCoupons.zip is in your case Export folder, you are going to need a tool to open it with, such as 7-Zip or WinRAR. You can download 7-Zip at:

```
http://www.7-zip.org/download.html
```

Once you have the 7-Zip tool, right-click MyCoupons.zip and click 7-Zip►Open Archive (see Figure 12-3).

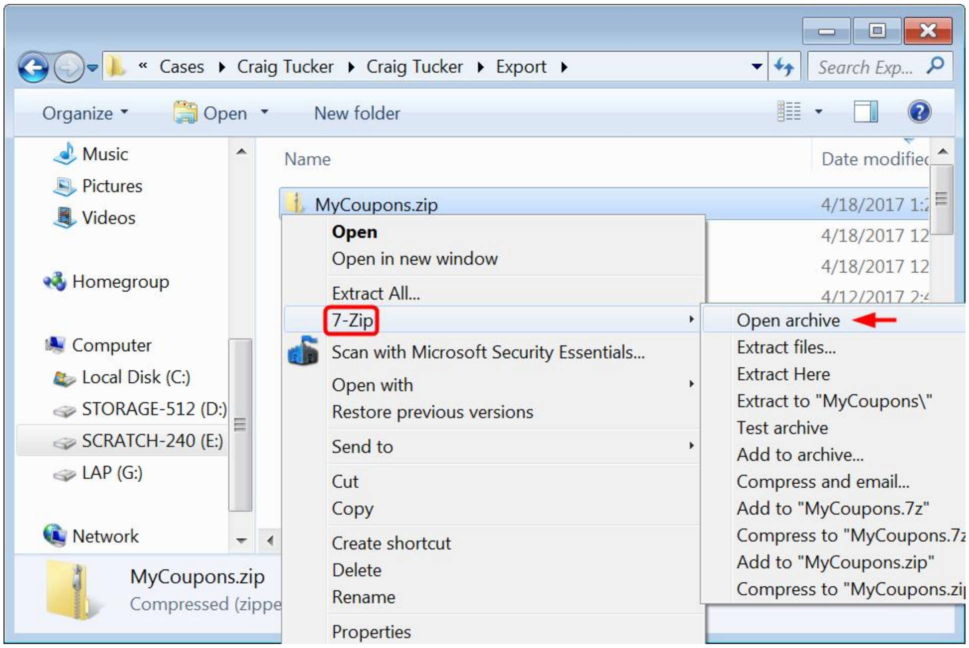
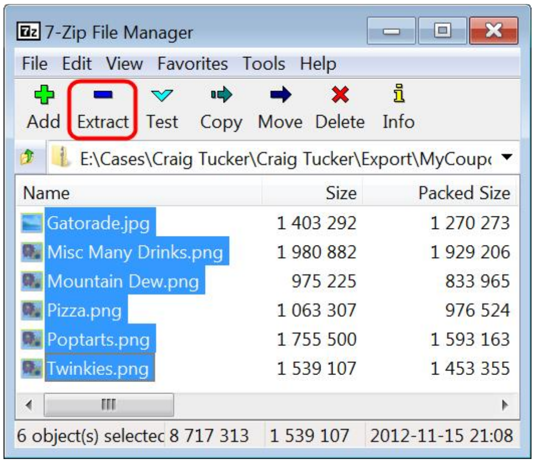


Figure 12-3 – Right-Click MyCoupons.zip and Select Open Archive under 7-Zip

When 7-Zip opens, highlight all the files and click the Extract button in the top bar.

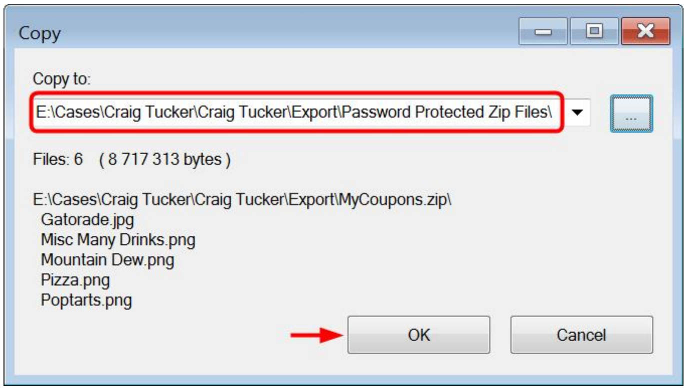

7-Zip File Manager
File Edit View Favorites Tools Help
Add
Extract
Test
Copy
Move
Delete
Info
E:\Cases\Craig Tucker\Craig Tucker\Export\MyCoup

Name	Size	Packed Size
Gatorade.jpg	1 403 292	1 270 273
Misc Many Drinks.png	1 980 882	1 929 206
Mountain Dew.png	975 225	833 965
Pizza.png	1 063 307	976 524
Poptarts.png	1 755 500	1 593 163
Twinkies.png	1 539 107	1 453 355

6 object(s) selectec 8 717 313 1 539 107 2012-11-15 21:08

Figure 12-4 – Highlight All Files and Click Extract Button

7-Zip will prompt you with a Copy window. Create a subfolder under your case Export folder and then click OK (see Figure 12-5).



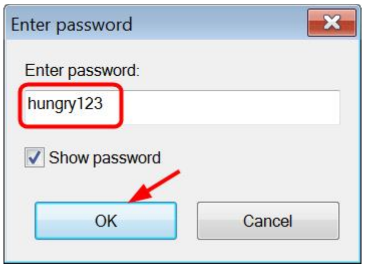
Copy to:
E:\Cases\Craig Tucker\Craig Tucker\Export\Password Protected Zip Files\

Files: 6 (8 717 313 bytes)

E:\Cases\Craig Tucker\Craig Tucker\Export\MyCoupons.zip\
Gatorade.jpg
Misc Many Drinks.png
Mountain Dew.png
Pizza.png
Poptarts.png

Figure 12-5 – Create Subfolder under Case Export Folder and Click OK

Next, you need to type in Craig’s login password “hungry123” and then click OK.

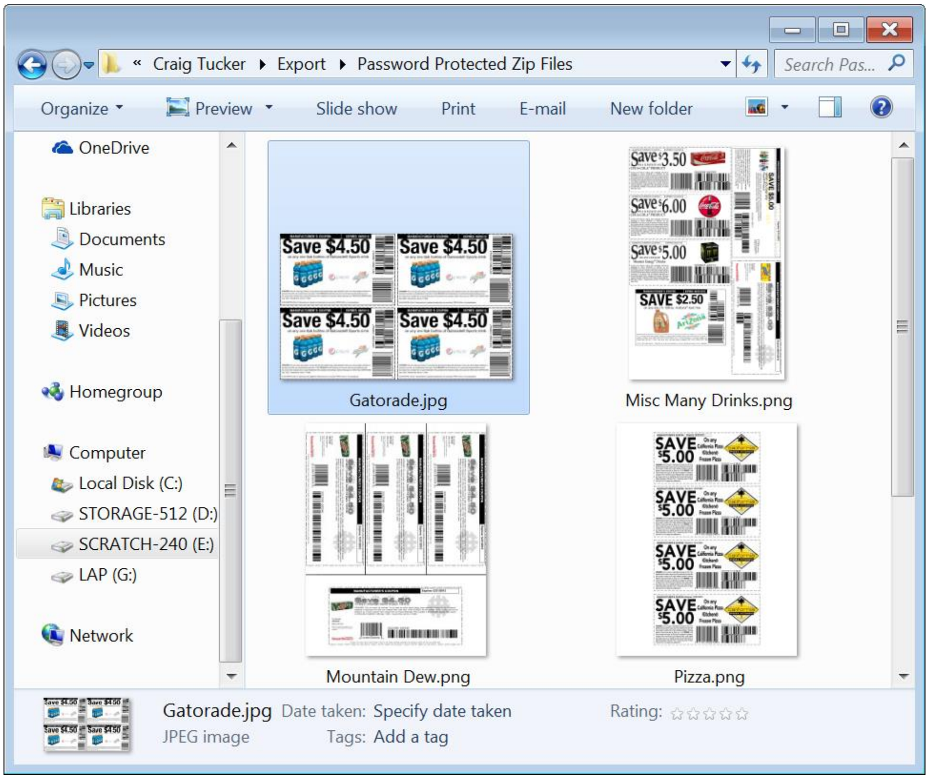


Enter password:
hungry123

☑ Show password

Figure 12-6 – Type in Craig’s Login Password and Click OK

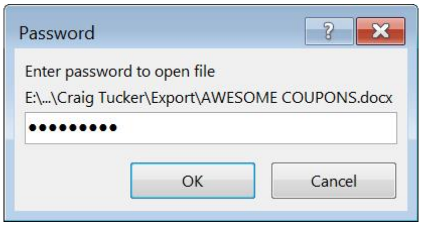
Now you can view the decrypted password protected zip files.



Gatorade.jpg Date taken: Specify date taken Rating: ☆☆☆☆☆
JPEG image Tags: Add a tag

Figure 12-7 – Password Protected Zip Files Decrypted

Now you should try to open Craig's AWESOME COUPONS.docx file with the same login password. When you open the file, you should be prompted with a Password window. Type in hungry123 and click OK.



Enter password to open file

E:\...\Craig Tucker\Export\AWESOME COUPONS.docx

●●●●●●●●●

Figure 12-8 – Type hungry123 for AWESOME COUPONS.docx

The program should tell you that the password is incorrect. If you were to try and find the password for this Word document, you would then want to look if the suspect wrote any passwords down or had sent any in emails or chat. You could also then follow the phased approach and next try an English Dictionary Attack with password breaking software.

Signature Analysis

Sometimes users have suspicious data they do not want deleted, but rather hidden. Users can accomplish storing data yet hiding its true nature by simply changing the file's name and/or extension. For example, a user could change naughty_photo.jpg to water_bill.txt in order to revert attention to the file. The change in file extension, however, is detectable through signature analysis. Each file has a signature embedded in itself which can then be compared to what the file claims to be. When these two pieces of information do not match, the investigators know a more detailed analysis of the file is required. The following are the "magic numbers" for common file types:

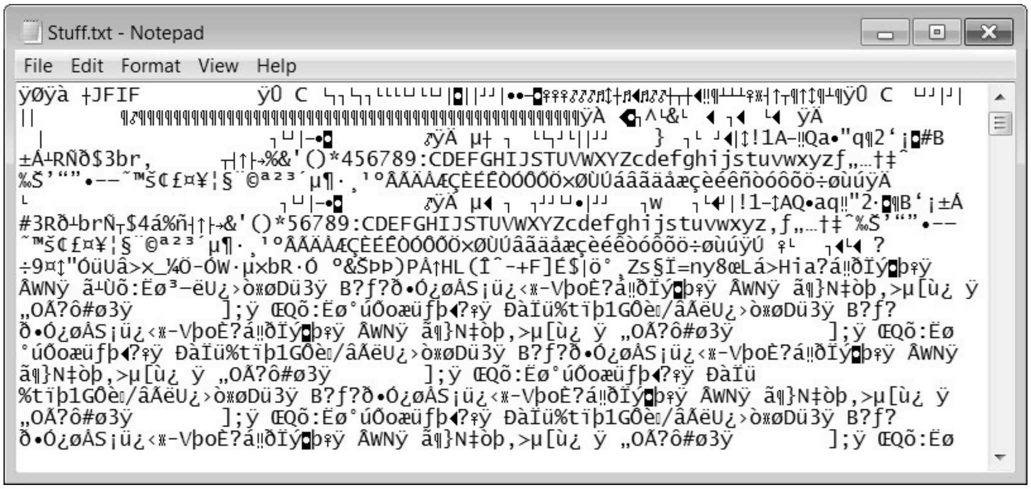
File Type	Extension	Magic Number / Hex Value
JPEG Graphic	.jpg	FF D8
PNG Graphic	.png	89 50 4E 47 0D 0A 1A 0A
MP3 Audio	.mp3	49 44 33
AVI Video	.avi	52 49 46 46
MOV Video	.mov	6D 6F 6F 76
Windows Video File	.wmv	30 26 B2 75 8E 66 CF
PDF	.pdf	25 50 44 46
Rich Text Document	.rtf	7B 5C 72 74 66 31
Word / Excel / PowerPoint Document	.doc / .xls / .ppt	D0 CF 11 E0 A1 B1 1A E1

These specific hex values (also known as "magic numbers") are at the beginning of each file and identify the file's type based on its content, not its file extension.

For additional signatures, you can use a File Extension Seeker or File Signature Table such as:

```
http://file-extension.net/seeker/
https://en.wikipedia.org/wiki/List_of_file_signatures
```

Windows uses a file's extension to determine what program to use to open or execute a file. For example, if there is a picture called Stuff.jpg, it will be opened with Windows Photo Viewer or another picture viewing program by default. However, if the file is renamed to Stuff.txt, it will be opened with Notepad or Word by default and it will look like Figure 12-9.

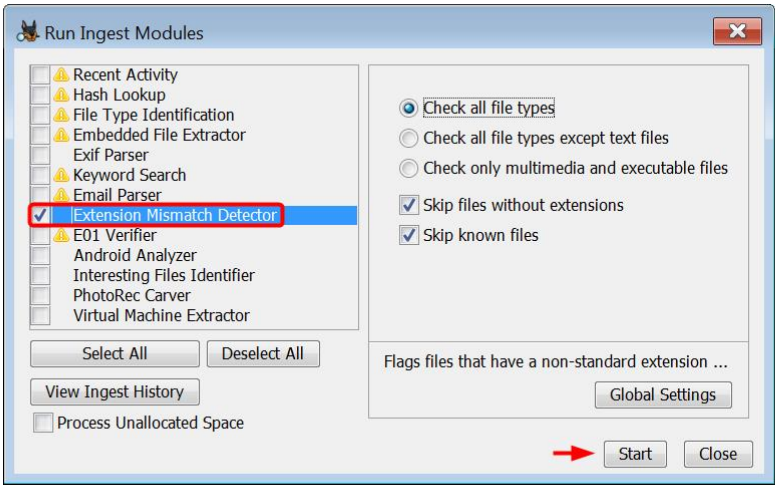


```
ÿØÿà ┼JFIF          ÿÛ C ┐┐┐┐└└┘└┘|◘||┘┘|••─◘♀♀♀♪♪♪♫↕┼♫◄♫♪♪┼┬┼◄‼¶┴┴┴♀¤┤↑┬¶↑↕¶┴¶ÿÛ C  ┘┘|┘|
||        ¶♪¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶¶ÿÀ ◘┐^└&└ ◄┐◄ └◄ ÿÄ
 |                       ┐┘|─•◘          ♪ÿÄ µ┼ ┐  └┐┘└||┘┘    } ┐└ ┘◄|↕!1A─‼Qa•"q¶2'¡◘#B
±Á┴RÑð$3br,       ┬┤↑├→%&'()*456789:CDEFGHIJSTUVWXYZcdefghijstuvwxyzƒ„…†‡ˆ
‰Š'""•–—˜™š¢£¤¥¦§¨©ª²³´µ¶·¸¹ºÂÃÄÅÆÇÈÉÊÒÓÔÕÖ×ØÙÚáâãäåæçèéêñòóôõö÷øùúÿÄ
└                        ┐┘|─•◘          ♪ÿÄ µ◄ ┐  ┐┘┘┘•|┘┘    ┐w    ┐└◄|!1─↕AQ•aq‼"2·◘¶B'¡±Á
#3Rð┴brÑ┬$4á%ñ┤↑├→&'()*56789:CDEFGHIJSTUVWXYZcdefghijstuvwxyz,ƒ„…†‡ˆ‰Š'""•–—
˜™š¢£¤¥¦§¨©ª²³´µ¶·¸¹ºÂÃÄÅÆÇÈÉÊÒÓÔÕÖ×ØÙÚâãäåæçèéêòóôõö÷øùúÿÚ ♀└   ┐◄└◄ ?
÷9¤↕"ÓüUâ>x_¾Ö-ÓW·µxbR·Ó °&ŠÞÞ)PÅ↑HL(Î˜-+F]É$|ö°¸Zs§Ï=ny8œLá>Hia?á‼ðÏý◘þ♀ÿ
ÂWNÿ ã┴Ùõ:Ëø³–ëU¿›ò¤øDü3ÿ B?ƒ?ð•Ó¿øÅS¡ü¿‹¤-VþoÈ?á‼ðÏý◘þ♀ÿ ÂWNÿ ã¶}N‡òþ,>µ[ù¿ ÿ
„OÅ?ô#ø3ÿ        ];ÿ ŒQõ:Ëø°úÕoæüƒþ◄?♀ÿ ÐàÏü%tïþ1GÕè▯/âÃëU¿›ò¤øDü3ÿ B?ƒ?
ð•Ó¿øÅS¡ü¿‹¤-VþoÈ?á‼ðÏý◘þ♀ÿ ÂWNÿ ã¶}N‡òþ,>µ[ù¿ ÿ „OÅ?ô#ø3ÿ        ];ÿ ŒQõ:Ëø
°úÕoæüƒþ◄?♀ÿ ÐàÏü%tïþ1GÕè▯/âÃëU¿›ò¤øDü3ÿ B?ƒ?ð•Ó¿øÅS¡ü¿‹¤-VþoÈ?á‼ðÏý◘þ♀ÿ ÂWNÿ
ã¶}N‡òþ,>µ[ù¿ ÿ „OÅ?ô#ø3ÿ        ];ÿ ŒQõ:Ëø°úÕoæüƒþ◄?♀ÿ ÐàÏü
%tïþ1GÕè▯/âÃëU¿›ò¤øDü3ÿ B?ƒ?ð•Ó¿øÅS¡ü¿‹¤-VþoÈ?á‼ðÏý◘þ♀ÿ ÂWNÿ ã¶}N‡òþ,>µ[ù¿ ÿ
„OÅ?ô#ø3ÿ        ];ÿ ŒQõ:Ëø°úÕoæüƒþ◄?♀ÿ ÐàÏü%tïþ1GÕè▯/âÃëU¿›ò¤øDü3ÿ B?ƒ?
ð•Ó¿øÅS¡ü¿‹¤-VþoÈ?á‼ðÏý◘þ♀ÿ ÂWNÿ ã¶}N‡òþ,>µ[ù¿ ÿ „OÅ?ô#ø3ÿ        ];ÿ ŒQõ:Ëø
```

Figure 12-9 - Picture Renamed to Text File, Windows Opens Picture in Notepad by Default

Extension Mismatch Detector

One way to view the files that are suspicious and possible candidates of a user-modified file extension is to use Autopsy's Extension Mismatch Detector plugin. Click on Tools►Run Ingest Modules►Tucker.E01. When the Run Ingest Modules window opens, check Extension Mismatch Detector and then click Start.



- ☐ Recent Activity
- ☐ Hash Lookup
- ☐ File Type Identification
- ☐ Embedded File Extractor
- ☐ Exif Parser
- ☐ Keyword Search
- ☐ Email Parser
- ☑ Extension Mismatch Detector
- ☐ E01 Verifier
- ☐ Android Analyzer
- ☐ Interesting Files Identifier
- ☐ PhotoRec Carver
- ☐ Virtual Machine Extractor

☐ Process Unallocated Space

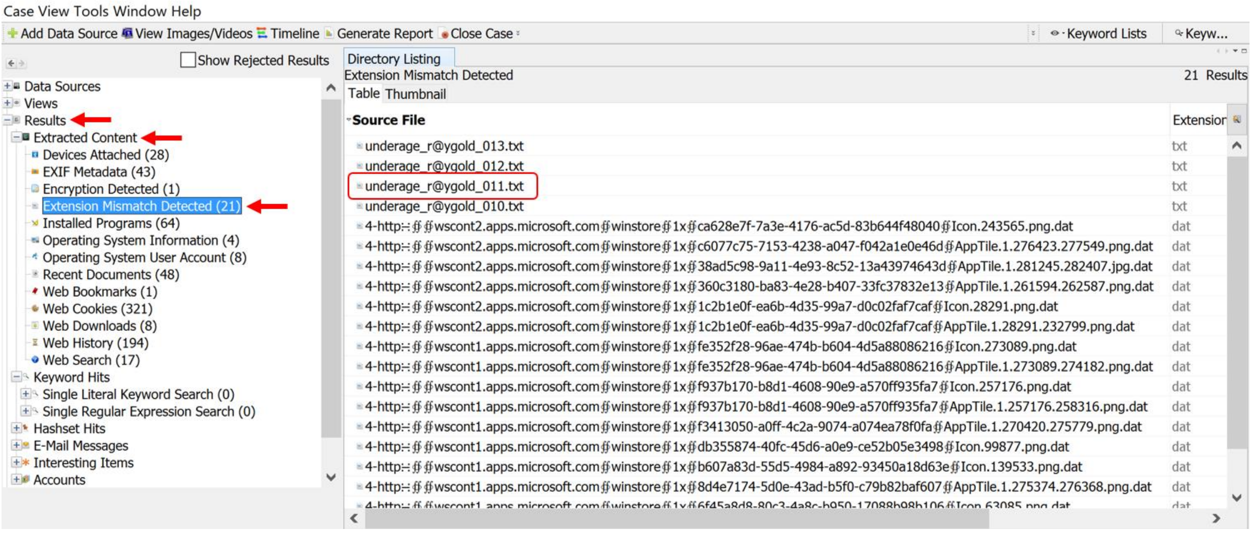
● Check all file types
○ Check all file types except text files
○ Check only multimedia and executable files

☑ Skip files without extensions
☑ Skip known files

Flags files that have a non-standard extension ...

Figure 12-10 – Check Extension Mismatch Detector and Click Start

When Autopsy finishes running, you can view the results in Results\Extracted Content\Extension Mismatch Detected window. This window is where Autopsy places any files that it finds during analysis whose signature possibly does not match its defined extension (see Figure 12-11).

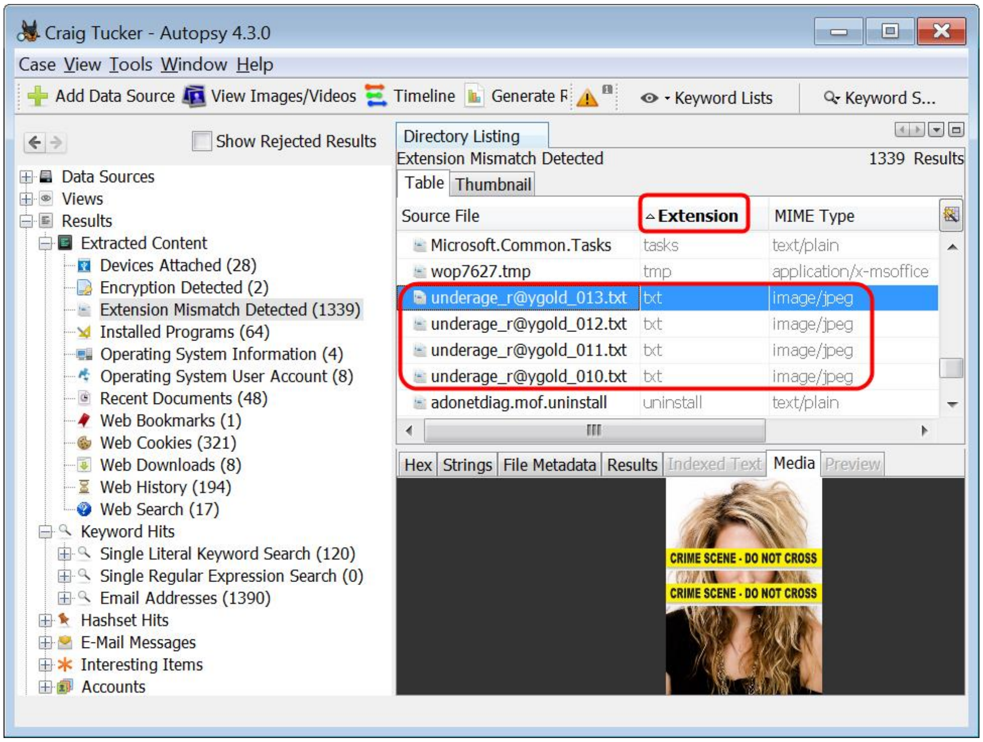


Extension Mismatch Detected — 21 Results

Source File	Extension
underage_r@ygold_013.txt	txt
underage_r@ygold_012.txt	txt
underage_r@ygold_011.txt	txt
underage_r@ygold_010.txt	txt
4-http꞉∯∯wscont2.apps.microsoft.com∯winstore∯1x∯ca628e7f-7a3e-4176-ac5d-83b644f48040∯Icon.243565.png.dat	dat
4-http꞉∯∯wscont2.apps.microsoft.com∯winstore∯1x∯c6077c75-7153-4238-a047-f042a1e0e46d∯AppTile.1.276423.277549.png.dat	dat
4-http꞉∯∯wscont2.apps.microsoft.com∯winstore∯1x∯38ad5c98-9a11-4e93-8c52-13a43974643d∯AppTile.1.281245.282407.jpg.dat	dat
4-http꞉∯∯wscont2.apps.microsoft.com∯winstore∯1x∯360c3180-ba83-4e28-b407-33fc37832e13∯AppTile.1.261594.262587.png.dat	dat
4-http꞉∯∯wscont2.apps.microsoft.com∯winstore∯1x∯1c2b1e0f-ea6b-4d35-99a7-d0c02faf7caf∯Icon.28291.png.dat	dat
4-http꞉∯∯wscont2.apps.microsoft.com∯winstore∯1x∯1c2b1e0f-ea6b-4d35-99a7-d0c02faf7caf∯AppTile.1.28291.232799.png.dat	dat
4-http꞉∯∯wscont1.apps.microsoft.com∯winstore∯1x∯fe352f28-96ae-474b-b604-4d5a88086216∯Icon.273089.png.dat	dat
4-http꞉∯∯wscont1.apps.microsoft.com∯winstore∯1x∯fe352f28-96ae-474b-b604-4d5a88086216∯AppTile.1.273089.274182.png.dat	dat
4-http꞉∯∯wscont1.apps.microsoft.com∯winstore∯1x∯f937b170-b8d1-4608-90e9-a570ff935fa7∯Icon.257176.png.dat	dat
4-http꞉∯∯wscont1.apps.microsoft.com∯winstore∯1x∯f937b170-b8d1-4608-90e9-a570ff935fa7∯AppTile.1.257176.258316.png.dat	dat
4-http꞉∯∯wscont1.apps.microsoft.com∯winstore∯1x∯f3413050-a0ff-4c2a-9074-a074ea78f0fa∯AppTile.1.270420.275779.png.dat	dat
4-http꞉∯∯wscont1.apps.microsoft.com∯winstore∯1x∯db355874-40fc-45d6-a0e9-ce52b05e3498∯Icon.99877.png.dat	dat
4-http꞉∯∯wscont1.apps.microsoft.com∯winstore∯1x∯b607a83d-55d5-4984-a892-93450a18d63e∯Icon.139533.png.dat	dat
4-http꞉∯∯wscont1.apps.microsoft.com∯winstore∯1x∯8d4e7174-5d0e-43ad-b5f0-c79b82baf607∯AppTile.1.275374.276368.png.dat	dat
4-http꞉∯∯wscont1.apps.microsoft.com∯winstore∯1x∯6f45a8d8-80c3-4a8c-b950-17088b98b106∯Icon.63085.png.dat	dat

Figure 12-11 – Results for Extension Mismatch Detector Plugin

Click the Extension column in the Table pane, and scroll down to the entries with txt. As you can see there are four text files of interest all with the names underage_r@ygold. Autopsy has determined based on the hex header of these files that they are actually jpeg images and not text files.



Extension Mismatch Detected — 1339 Results

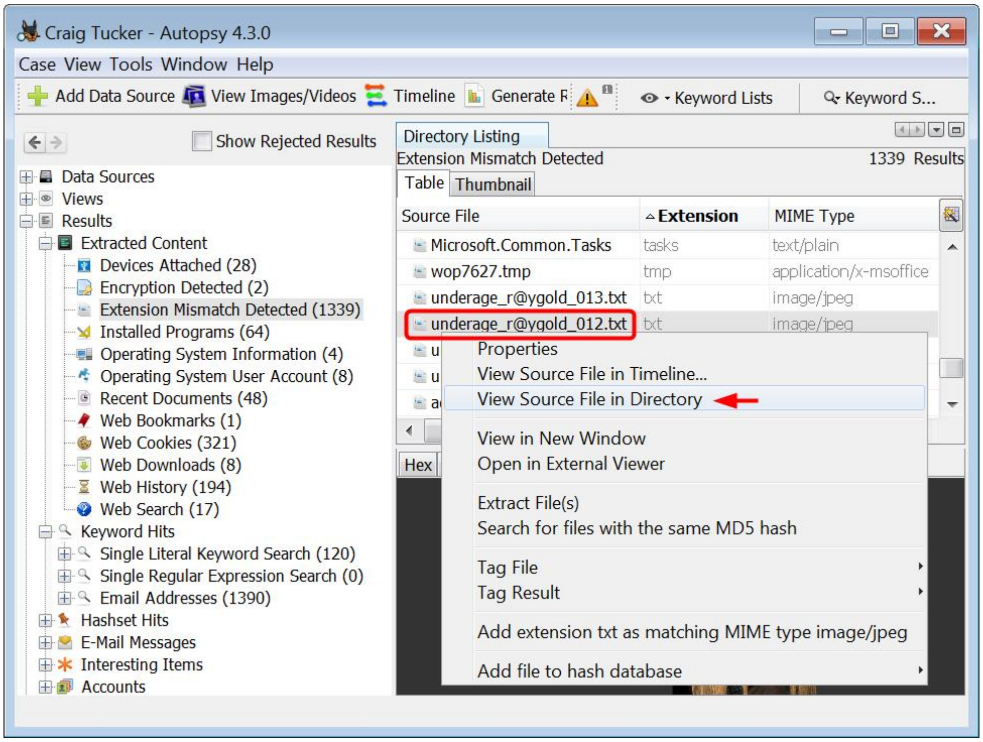
Source File	Extension	MIME Type
Microsoft.Common.Tasks	tasks	text/plain
wop7627.tmp	tmp	application/x-msoffice
underage_r@ygold_013.txt	txt	image/jpeg
underage_r@ygold_012.txt	txt	image/jpeg
underage_r@ygold_011.txt	txt	image/jpeg
underage_r@ygold_010.txt	txt	image/jpeg
adonetdiag.mof.uninstall	uninstall	text/plain

Figure 12-12 – Sort by Extension Column and View Text Files that are Images

Note: The problem with this category is that it comes back with many results that are not useful. It's true that these files' extensions don't match their headers, but that does not mean the suspect intentionally renamed the extensions. There are several Windows system files and programs files that have mismatched headers and extensions.

Hex View Example

In Autopsy, navigate to the location of the file that you deem as suspicious and open it in Hex view. Open the file “underage_r@ygold_011.txt” to check its contents, as it seems suspicious.

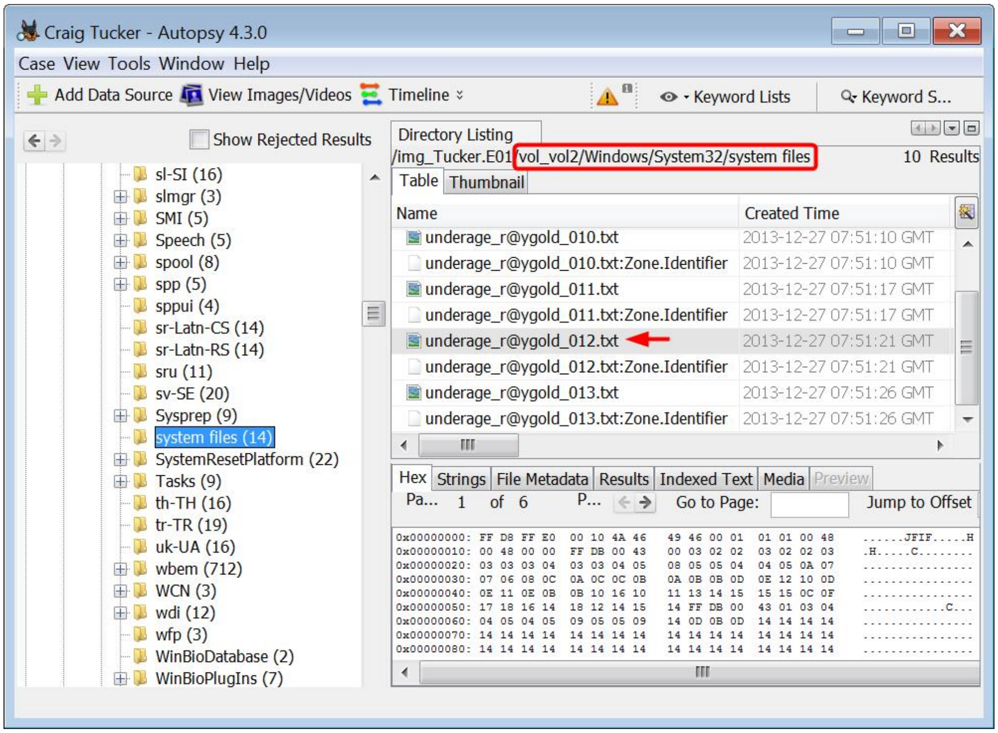


Source File	Extension	MIME Type
Microsoft.Common.Tasks	tasks	text/plain
wop7627.tmp	tmp	application/x-msoffice
underage_r@ygold_013.txt	txt	image/jpeg
underage_r@ygold_012.txt	txt	image/jpeg
u		
u		
a		

Figure 12-13 – Right-Click One of the Text Files and Click View Source File in Directory

Autopsy will take you to the folder where these pictures are being stored, which is the following path (see Figure 12-14):

```
C:\Windows\System32\system files
```



sl-SI (16)
slmgr (3)
SMI (5)
Speech (5)
spool (8)
spp (5)
sppui (4)
sr-Latn-CS (14)
sr-Latn-RS (14)
sru (11)
sv-SE (20)
Sysprep (9)
system files (14)
SystemResetPlatform (22)
Tasks (9)
th-TH (16)
tr-TR (19)
uk-UA (16)
wbem (712)
WCN (3)
wdi (12)
wfp (3)
WinBioDatabase (2)
WinBioPlugIns (7)

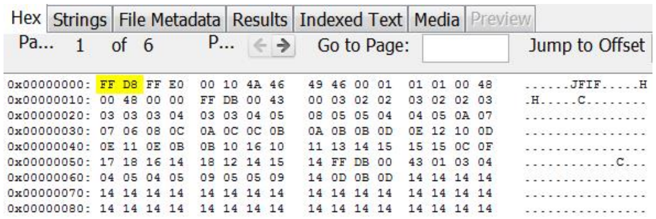
/img_Tucker.E01/vol_vol2/Windows/System32/system files 10 Results

Name	Created Time
underage_r@ygold_010.txt	2013-12-27 07:51:10 GMT
underage_r@ygold_010.txt:Zone.Identifier	2013-12-27 07:51:10 GMT
underage_r@ygold_011.txt	2013-12-27 07:51:17 GMT
underage_r@ygold_011.txt:Zone.Identifier	2013-12-27 07:51:17 GMT
underage_r@ygold_012.txt	2013-12-27 07:51:21 GMT
underage_r@ygold_012.txt:Zone.Identifier	2013-12-27 07:51:21 GMT
underage_r@ygold_013.txt	2013-12-27 07:51:26 GMT
underage_r@ygold_013.txt:Zone.Identifier	2013-12-27 07:51:26 GMT

```
0x00000000: FF D8 FF E0  00 10 4A 46  49 46 00 01  01 01 00 48   ......JFIF.....H
0x00000010: 00 48 00 00  FF DB 00 43  00 03 02 02  03 02 02 03   .H.....C........
0x00000020: 03 03 03 04  03 03 04 05  08 05 05 04  04 05 0A 07   ................
0x00000030: 07 06 08 0C  0A 0C 0C 0B  0A 0B 0B 0D  0E 12 10 0D   ................
0x00000040: 0E 11 0E 0B  0B 10 16 10  11 13 14 15  15 15 0C 0F   ................
0x00000050: 17 18 16 14  18 12 14 15  14 FF DB 00  43 01 03 04   .............C...
0x00000060: 04 05 04 05  09 05 05 09  14 0D 0B 0D  14 14 14 14   ................
0x00000070: 14 14 14 14  14 14 14 14  14 14 14 14  14 14 14 14   ................
0x00000080: 14 14 14 14  14 14 14 14  14 14 14 14  14 14 14 14   ................
```

Figure 12-14 – Renamed Picture Files Stored in system files Subfolder

View one of these picture files in hex view and look at the header. As you can see it has a header of a JPEG Graphic file.



```
0x00000000: FF D8 FF E0  00 10 4A 46  49 46 00 01  01 01 00 48   ......JFIF.....H
0x00000010: 00 48 00 00  FF DB 00 43  00 03 02 02  03 02 02 03   .H.....C........
0x00000020: 03 03 03 04  03 03 04 05  08 05 05 04  04 05 0A 07   ................
0x00000030: 07 06 08 0C  0A 0C 0C 0B  0A 0B 0B 0D  0E 12 10 0D   ................
0x00000040: 0E 11 0E 0B  0B 10 16 10  11 13 14 15  15 15 0C 0F   ................
0x00000050: 17 18 16 14  18 12 14 15  14 FF DB 00  43 01 03 04   .............C...
0x00000060: 04 05 04 05  09 05 05 09  14 0D 0B 0D  14 14 14 14   ................
0x00000070: 14 14 14 14  14 14 14 14  14 14 14 14  14 14 14 14   ................
0x00000080: 14 14 14 14  14 14 14 14  14 14 14 14  14 14 14 14   ................
```

Figure 12-15 – JPEG Graphic File Header on Renamed Text File

When you compare the extension of “underage_r@ygold_011” with the file type found from the signature, you should have a mismatch. The signature matches a JPEG Graphic file with “FF D8”, rather than that of a Plain Text File (.txt). Therefore, you as an investigator should do more analysis on the file since the user was attempting to keep its contents hidden.