Windows 8 Forensic Guide

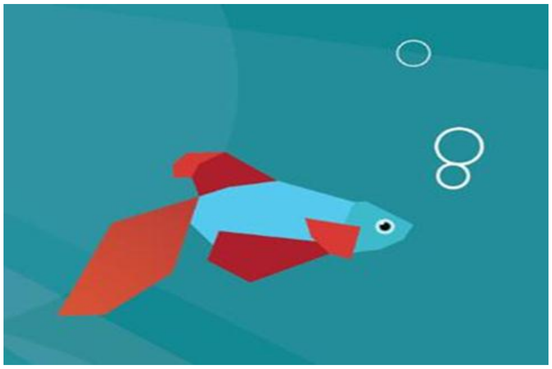
Amanda C. F. Thomson, M.F.S. Candidate

Advised by Eva Vincze, PhD

The George Washington University, Washington, D.C.

Windows® 8 ™

Consumer Preview



Windows 8 Forensic Guide

Amanda C. F. Thomson
The George Washington University
Washington, D.C.

Contents

About This Guide

With a new operating system, come new forensic challenges. Microsoft's Windows 8 is connected to everything – wherever you sign in, it's connected. E-mail is connected to Facebook is connected to contacts is connected to Internet Explorer is connected to … you get the point.

Windows 8 is an operating system "reimagined and reinvented from a solid core of Windows 7 speed and reliability"[i]. While I can neither confirm nor deny this statement, there are certainly many forensically interesting spots we are familiar with from Windows 7 and Vista, which is good for us because it means this operating system is not completely reinvented. With Windows 8, you will still find that Windows is Windows – it keeps track of everything. App Data and its Local and Roaming folders are still present. The Registry has the same structure we've been familiar with for quite some time. And Windows still has the same standard programs. Some things in Windows 8, however, are different.

Gone are the days when we could just sit, or read a book, or, dare I suggest it? - talk to the person next to us! – while waiting for an appointment or riding the train. Everywhere we go, we see people staring intently into their tablet or cell phone reading the latest celebrity gossip, updating Facebook, calling in sick to work, and shopping online, all while texting and driving. Hopefully not, but you get the point. And so does Microsoft. Windows 8 is an operating system geared toward mobile devices, and that is definitely evident with the new interface.

i could be doing a million awesome things right now if this stupid computer didn't make me look at it!

When I registered for an independent research project in my program at The George Washington University, I wanted to do something that would contribute to the computer forensic community. So I decided to take on Windows 8. And by "take on", I mean, it consumed my life for nearly four months. No more Facebook. No more Netflix. It was just me and Windows 8 every night after work. Friday nights. Weekends. Thankfully, Windows 8 did not care that I was turning into a pasty basement-dwelling nerd subsisting off of caffeine and over-processed food.

While I am very well aware of this and other operating systems' existence, I somehow failed to realize, despite my forensic experience and everything I have learned since I entered the industry, that I would be researching an *entire operating system*. Wait… what? That doesn't make sense? Let me explain - I had this lofty goal of creating a user manual with charts and cheat sheets and compiling everything that could ever be possibly useful to a forensic examiner. While I did create a user manual with charts and cheat sheets, this is not a comprehensive guide. In fact, I would not be surprised if I did not scratch the surface of Windows 8, because while much of it is forensically similar to Windows 7, there is so much more that is completely different.

For those wondering what my research methodology was, here's what I did:
Originally I started this project with *Windows 8 Developer Preview*, but when *Consumer* Preview came out at the end of February, I started over. I downloaded *Windows 8 Consumer Preview 32-bit Edition* from Microsoft and installed it in a virtual machine using VMWare Workstation 8[ii]. I used it for nearly two weeks and every couple of days I made an image using FTK Imager v3.0.1[iii]. I then used Guidance Software's EnCase Forensic v6.17 for my examination and analysis and a variety of written resources (which have been given credit)[iv].

So, I have done my best to find forensically interesting artifacts and information in Windows 8. When I did find something, I pointed it out, attempted to figure out what was going on, and offer an explanation. When I couldn't figure it out, I stated so, because my hope is that this user guide will be a "living" document. I want to keep it updated and as I discover new things in Windows 8, or revalidate what we already know from 7 and Vista, I will add to this. If *you* find something new or confirm an existing fact, please let me know and you will be credited accordingly. I have tried to keep the language of this guide easy to read, but if there is something that is unclear or I am wrong, let me know that, too.

In this guide, you will find a section on Windows Artifacts, a section devoted to the Communications App, and the last section on the Windows Registry. Boiling down this research project to just those three items doesn't sound like much, but I think I packed a lot of information into those three sections. I learned a lot conducting this research and actually did have *some* fun, but what I really hope to get out of this is that *you* found this guide useful and it made your job as a forensic examiner a bit easier. If you have any comments or suggestions, please shoot me an e-mail at *propellerheadforensics@gmail.com*. For updates, visit my website at *http://propellerheadforensics.com* or follow me on Twitter @propellerhead23.

Windows 8 User Interface

Nearly everything that is new about this OS is geared toward touch screen devices; you can sign-in by swiping your finger on the screen in a pre-set pattern, you can read a document by "flipping" through the pages, and you can zoom in on an object by expanding the screen with two fingers.

While it is still possible to access the old interface, we can begin to get ideas for figuring out where data of forensic interest might reside by spending some time with the new one. I wanted to go over the Windows 8 UI because I also think it can help us get an idea of what the user's experience was like. During our forensic examinations, we are usually able to determine what was important to the user, such as their documents, pictures, Internet favorites, etc., because we know where to look. A majority of us have used Windows enough to know common locations we are likely to store our data and generally look there first. We may also be able to visualize what this looked like from the user's perspective (unless you're lucky enough to get an image of their hard drive to operate in a VM). Regardless of your method, it gives us better awareness of where to look for forensic artifacts and other useful data.

Figure 1 shows the user's login/lock screen will display their calendar, e-mail notifications, and Facebook notifications, if they have enabled this feature.

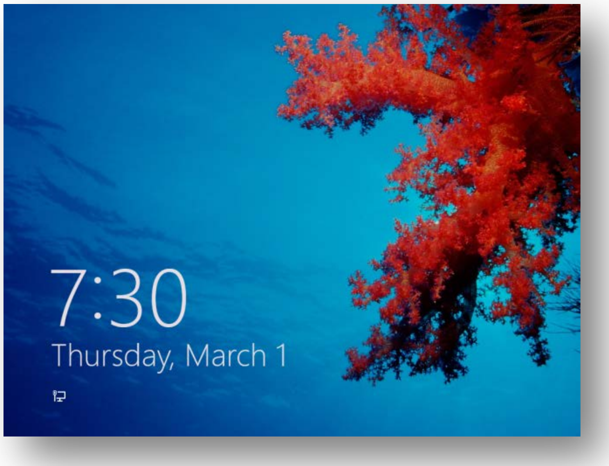
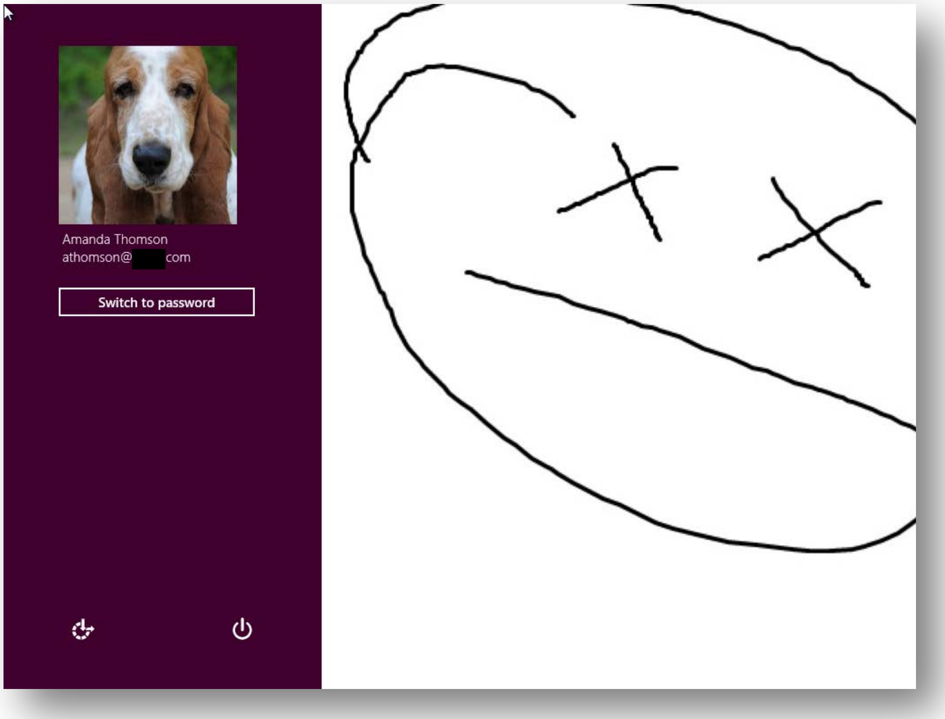


Figure 1 Windows 8 Login Screen

There are three options to sign-in to Windows 8 – traditional sign-in, picture sign-in, and PIN sign-in. Picture sign-in allows you to draw a pattern to sign-in to your computer (*Figure **2***), and PIN sign-in is just that – using a PIN to sign-in.



Amanda Thomson
athomson@[redacted]com

Figure 2 Windows 8 Picture Sign-In

The new Start Menu, which also appears to be the Desktop, is much different than the traditional Windows Start Menu we are accustomed to, and will probably garner a lot of attention (or complaints). *Figure 3* shows that the Start Menu is made up of Tiles, which consists of Metro Apps, which seems to be Microsoft's new term for "programs" in Windows 8. The default Start Menu includes an app for the Windows Store, Internet Explorer 10, a variety of communications apps, a Map App, and a Weather App. Several apps are available for the user to download from the Windows Store.

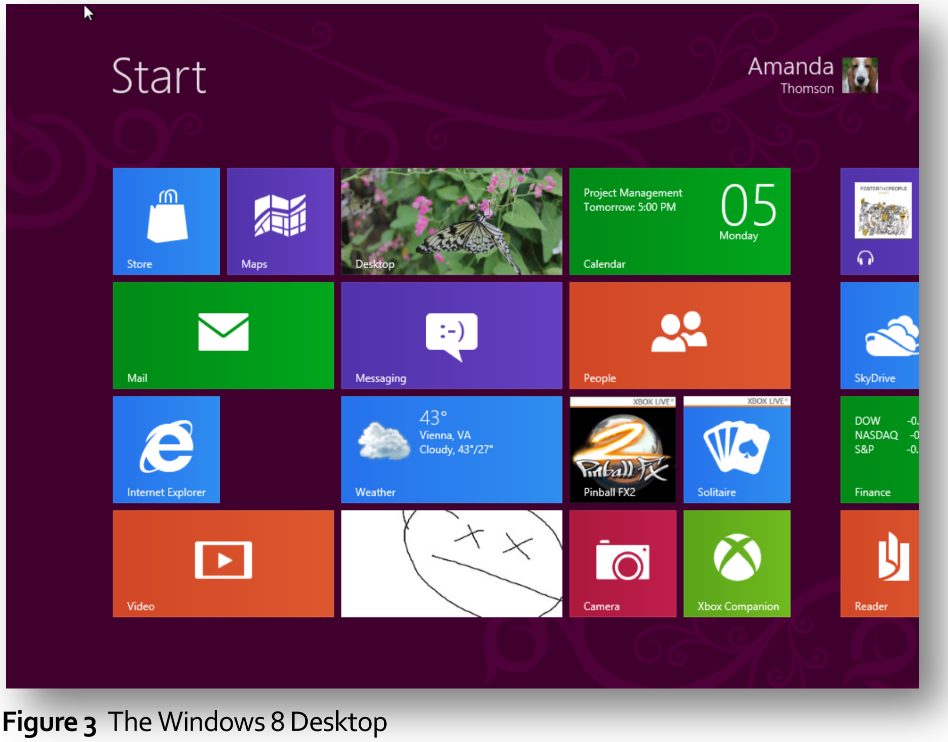


Figure 3 The Windows 8 Desktop

The Windows 8 Desktop has "Charms", which basically allow you to quickly access Windows features, such as Search, Share, Devices, and Settings (*Figure 4*).

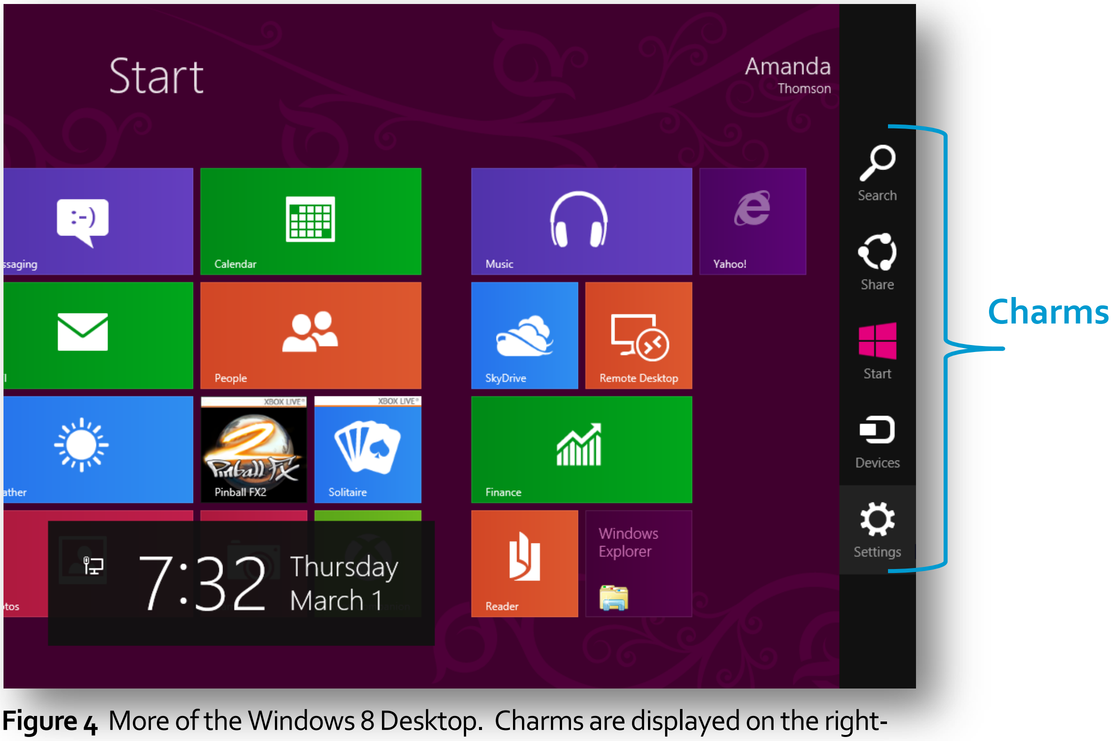
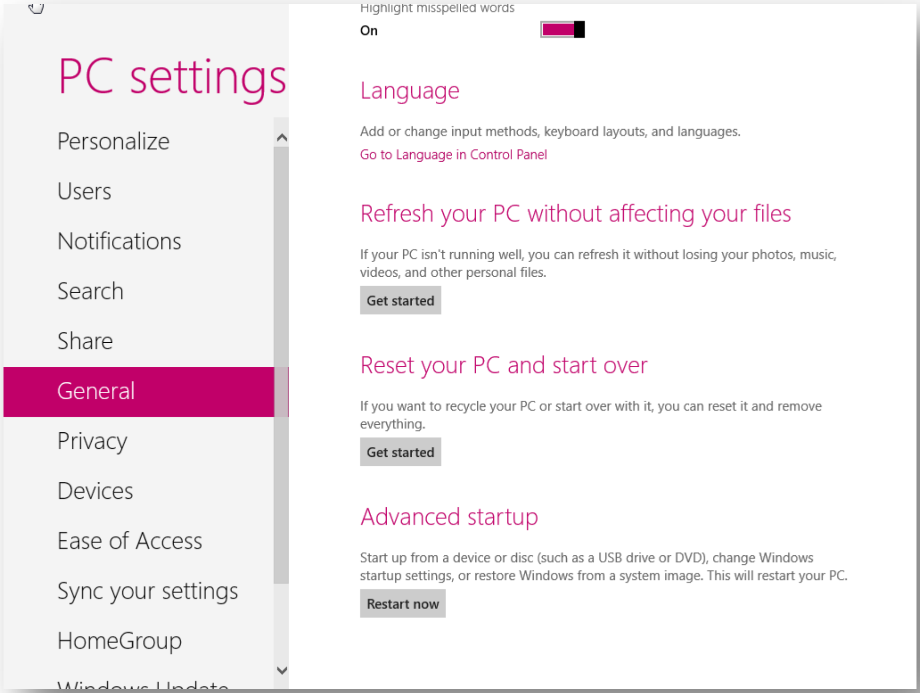


Figure 4 More of the Windows 8 Desktop. Charms are displayed on the right-hand side.

From Charms, you can access PC Settings. Many of these settings were inaccessible in *Consumer Preview* (but should be accessible when Windows 8 is officially released), but there were a couple of noticeable settings that are "new" to Windows 8. These may not necessarily be new features, but Microsoft has definitely made Windows 8 more user-friendly in terms of being able to understand what you are doing to your computer.



Highlight misspelled words

On

Language

Add or change input methods, keyboard layouts, and languages.

Refresh your PC without affecting your files

If your PC isn't running well, you can refresh it without losing your photos, music, videos, and other personal files.

Reset your PC and start over

If you want to recycle your PC or start over with it, you can reset it and remove everything.

Advanced startup

Start up from a device or disc (such as a USB drive or DVD), change Windows startup settings, or restore Windows from a system image. This will restart your PC.

Figure 5 PC Settings

Figure 5 shows that under "General", you have two System Restore-like options – "Refresh your PC without affecting your files" and "Reset your PC and start over".

Here's what happens when you refresh your PC:

- Your files and personalization settings won't change
- Your PC settings will be changed back to their default
- Apps from Windows Store will be kept
- Apps you installed from discs or websites will be removed
- A list of removed apps will be saved on your desktop

Resetting your PC does this:

- All your personal files and apps will be removed
- Your PC settings will be changed back to their defaults

Figure 6, *Figure 7*, and *Figure 8* show a couple of other apps you might see:

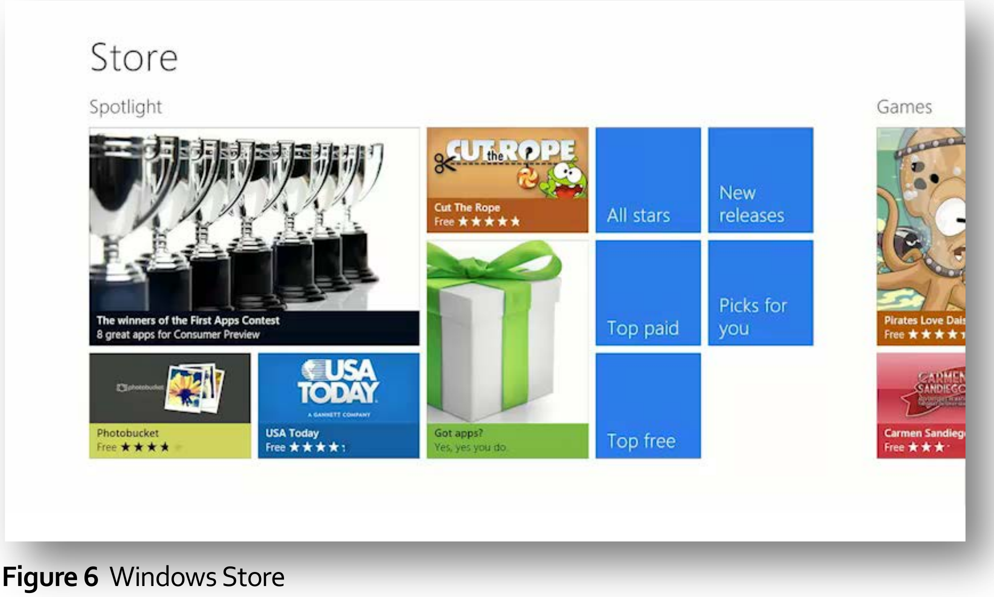
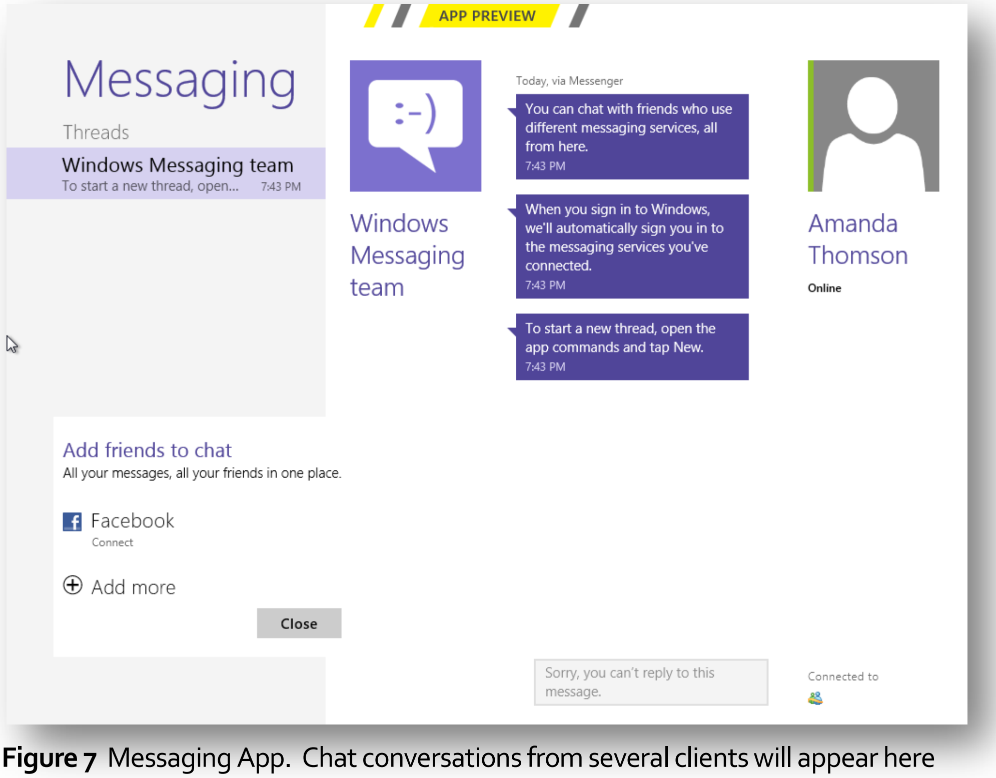


Figure 6 Windows Store



Today, via Messenger

You can chat with friends who use different messaging services, all from here.
7:43 PM

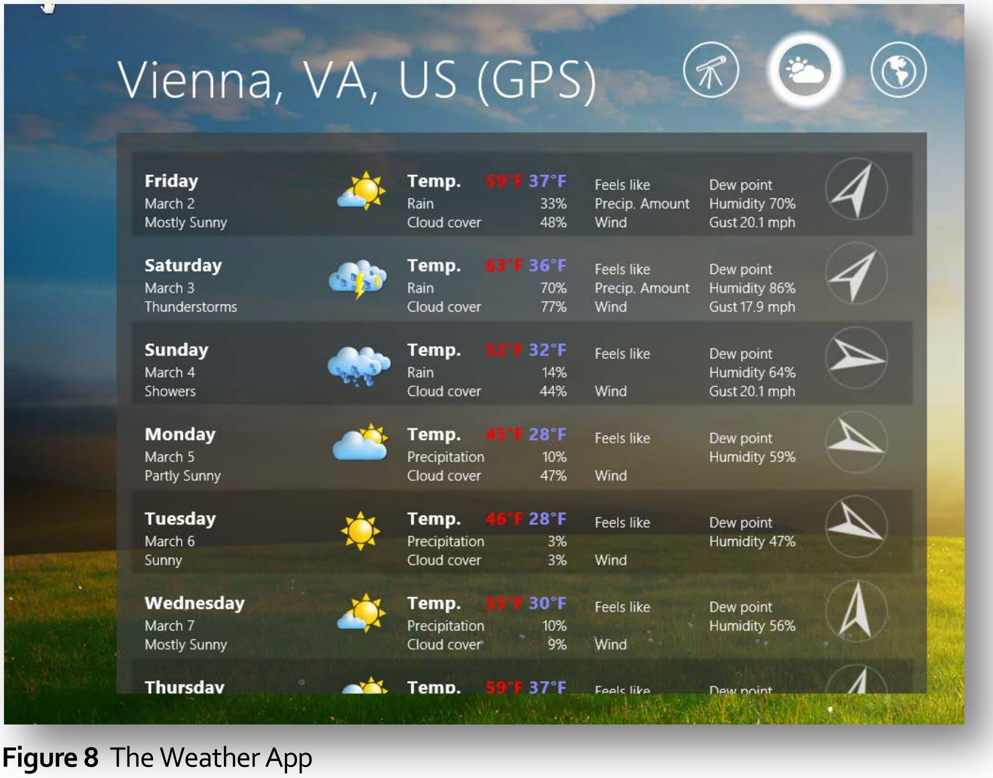
When you sign in to Windows, we'll automatically sign you in to the messaging services you've connected.
7:43 PM

To start a new thread, open the app commands and tap New.
7:43 PM

Add friends to chat
All your messages, all your friends in one place.

Sorry, you can't reply to this message.

Figure 7 Messaging App. Chat conversations from several clients will appear here

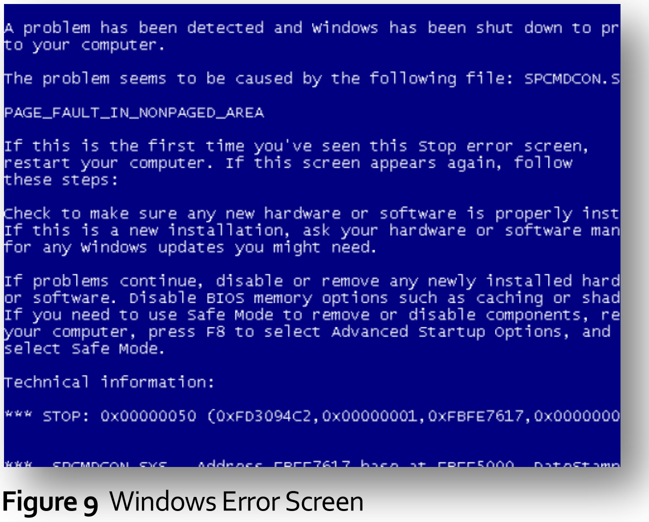


Vienna, VA, US (GPS)

Friday March 2 Mostly Sunny	**Temp.** 59°F 37°F Rain 33% Cloud cover 48%	Feels like Precip. Amount Wind	Dew point Humidity 70% Gust 20.1 mph
Saturday March 3 Thunderstorms	**Temp.** 63°F 36°F Rain 70% Cloud cover 77%	Feels like Precip. Amount Wind	Dew point Humidity 86% Gust 17.9 mph
Sunday March 4 Showers	**Temp.** 52°F 32°F Rain 14% Cloud cover 44%	Feels like Wind	Dew point Humidity 64% Gust 20.1 mph
Monday March 5 Partly Sunny	**Temp.** 45°F 28°F Precipitation 10% Cloud cover 47%	Feels like Wind	Dew point Humidity 59%
Tuesday March 6 Sunny	**Temp.** 46°F 28°F Precipitation 3% Cloud cover 3%	Feels like Wind	Dew point Humidity 47%
Wednesday March 7 Mostly Sunny	**Temp.** 55°F 30°F Precipitation 10% Cloud cover 9%	Feels like Wind	Dew point Humidity 56%
Thursday	**Temp.** 59°F 37°F	Feels like	Dew point

Figure 8 The Weather App

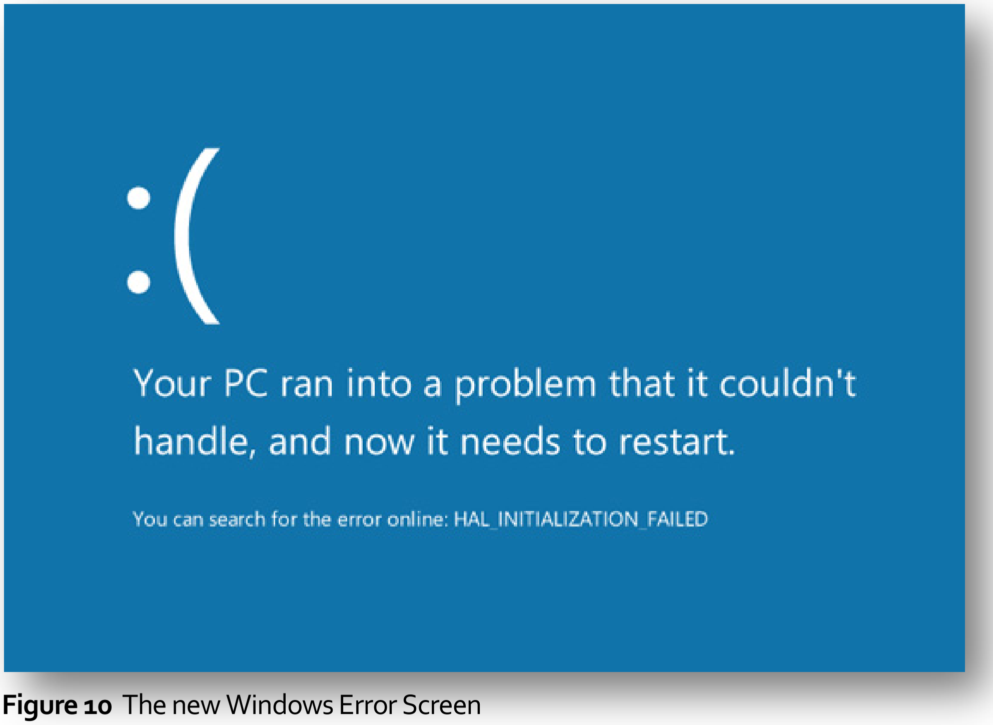
And Windows wouldn't be Windows without everyone's favorite – the Error Screen, or as most of us know it - the Blue Screen of Death (*Figure 9*). Unfortunately, we are probably all too familiar with this screen and have been frustrated with how quickly the error code zips by before we can even catch a glimpse and before you know it, your PC is restarting.



```
A problem has been detected and windows has been shut down to pr
to your computer.

The problem seems to be caused by the following file: SPCMDCON.S

PAGE_FAULT_IN_NONPAGED_AREA

If this is the first time you've seen this Stop error screen,
restart your computer. If this screen appears again, follow
these steps:

Check to make sure any new hardware or software is properly inst
If this is a new installation, ask your hardware or software man
for any windows updates you might need.

If problems continue, disable or remove any newly installed hard
or software. Disable BIOS memory options such as caching or shad
If you need to use Safe Mode to remove or disable components, re
your computer, press F8 to select Advanced Startup Options, and
select Safe Mode.

Technical information:

*** STOP: 0x00000050 (0xFD3094C2,0x00000001,0xFBFE7617,0x0000000

***  SPCMDCON.SYS - Address FBFE7617 base at FBFE5000, DateStamp
```

Figure 9 Windows Error Screen

But there’s hope! The error code is now in plain English. And maybe we won’t be *as* angry with Windows because the new Blue Screen of Death appears to empathize with you (*Figure 10*):



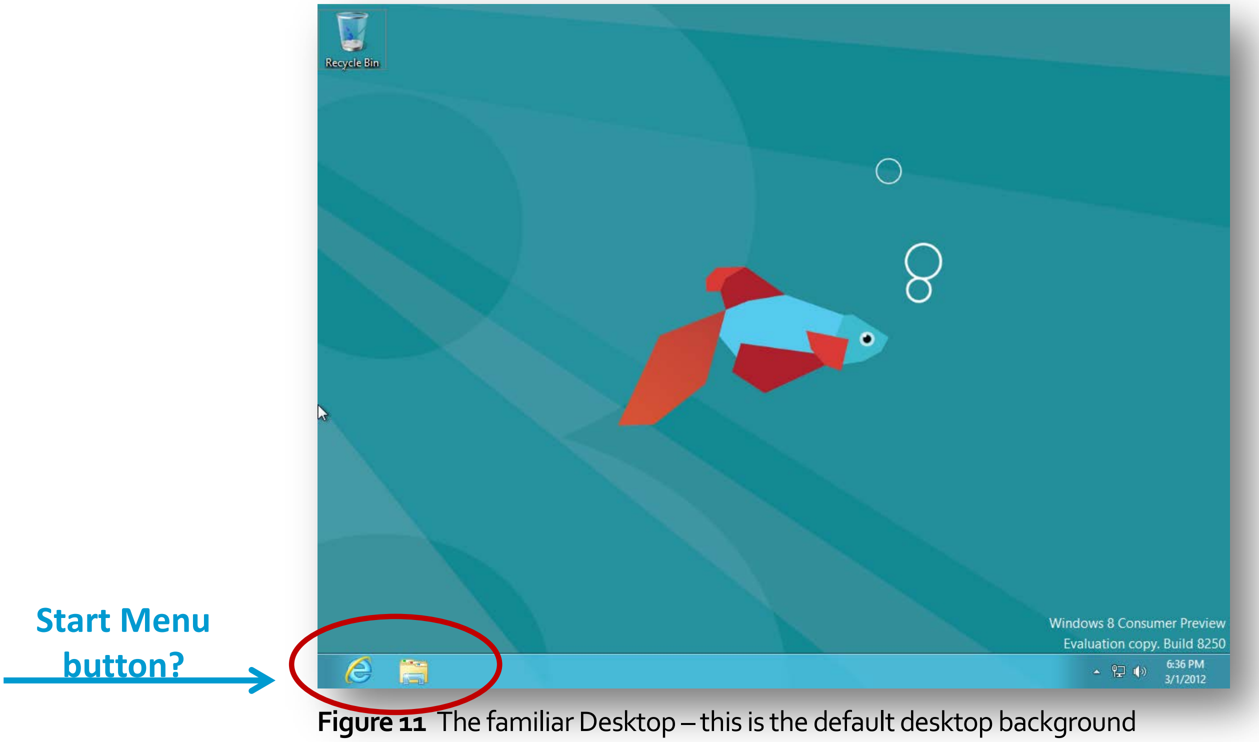
:(

Your PC ran into a problem that it couldn't handle, and now it needs to restart.

You can search for the error online: HAL_INITIALIZATION_FAILED

Figure 10 The new Windows Error Screen

You can access the familiar Windows Desktop from the Desktop Tile in the new Metro UI. *Figure 11* shows what the default looks like. One of the first things I noticed that was different from *Developer Preview* was that in *Consumer Preview*, the Start Menu button was missing. Since *Consumer Preview* is still a testing platform, it is unknown at this time if the Start Menu button will make a re-appearance when the final version hits store shelves later this year.



Start Menu button?

Figure 11 The familiar Desktop – this is the default desktop background

Even though the Start Menu button is missing, it is still possible to access Start Menu items (*Figure 12*). Hovering the mouse in the bottom left-hand corner will allow you to access the Metro UI and hovering over the left side of the screen will display a list of apps that you've used and are currently still running. The app at the top-left was the last one used.

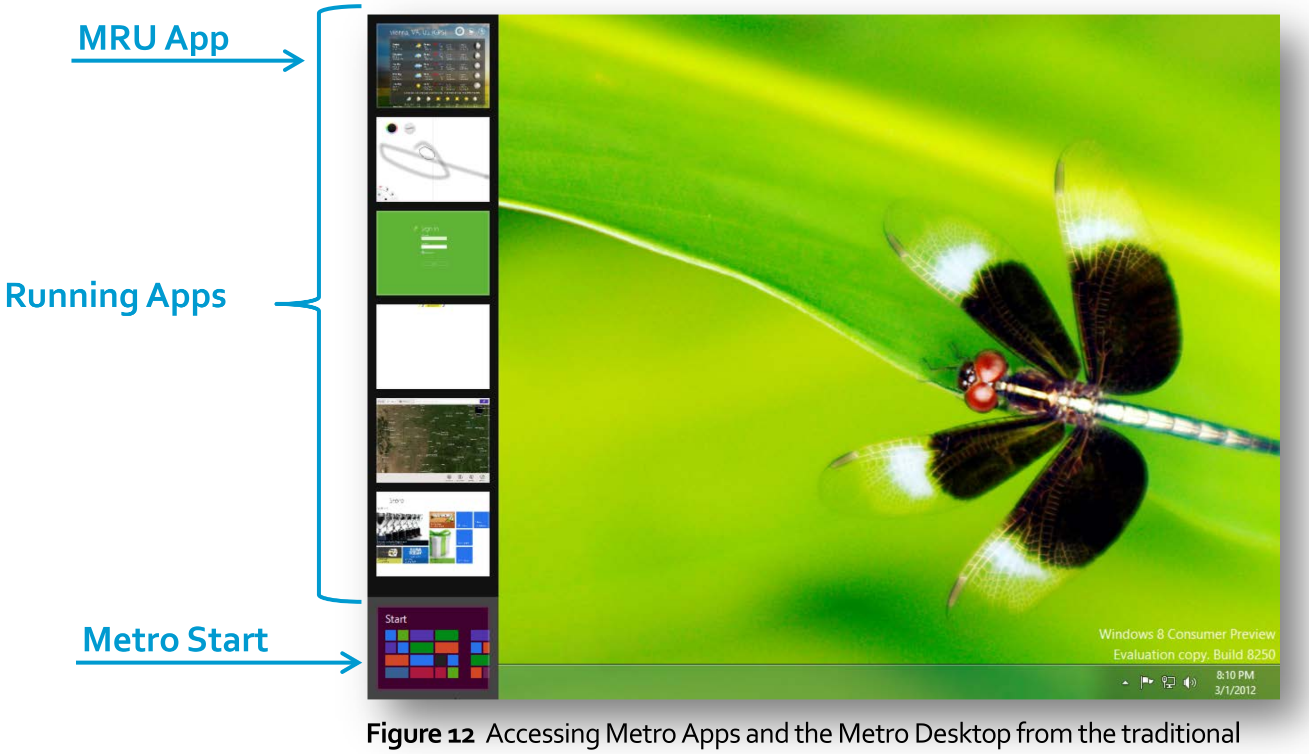
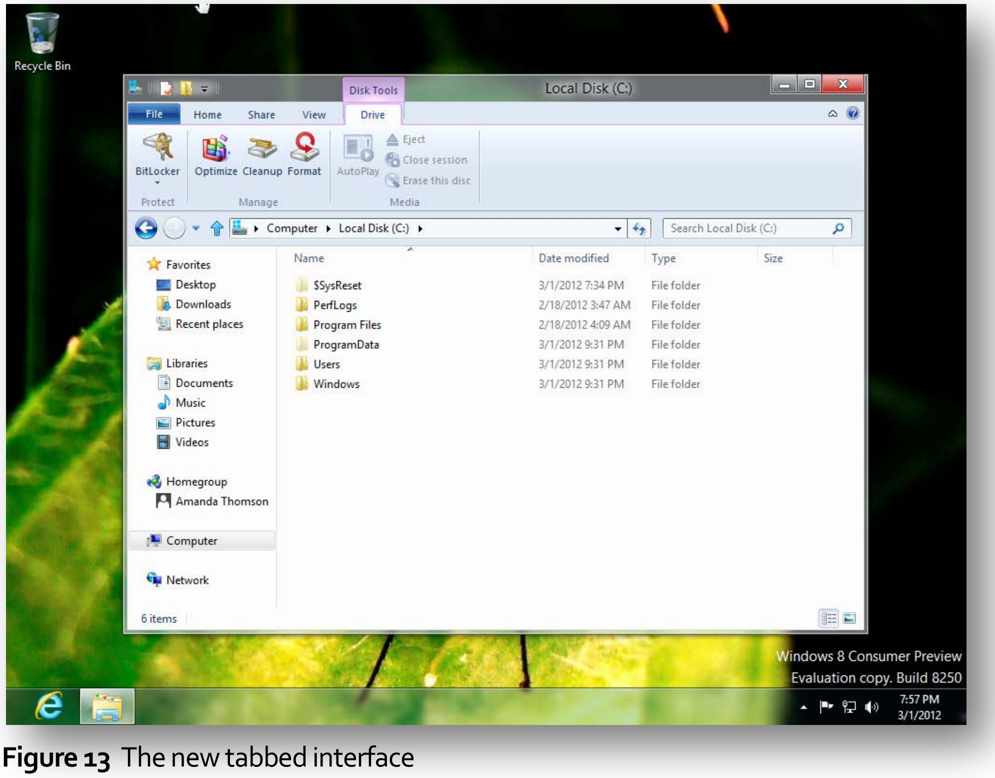


Figure 12 Accessing Metro Apps and the Metro Desktop from the traditional Desktop

Windows Explorer also has a new look and feel. *Figure 13* shows that Windows Explorer has a tabbed interface, similar to newer versions of Microsoft Office.



Name	Date modified	Type	Size
$SysReset	3/1/2012 7:34 PM	File folder	
PerfLogs	2/18/2012 3:47 AM	File folder	
Program Files	2/18/2012 4:09 AM	File folder	
ProgramData	3/1/2012 9:31 PM	File folder	
Users	3/1/2012 9:31 PM	File folder	
Windows	3/1/2012 9:31 PM	File folder	

Figure 13 The new tabbed interface

Windows Artifacts

Just like other versions of Windows, Windows 8 contains valuable information known as "artifacts". The user is oftentimes unaware that the operating system is leaving traces of their activity behind that is specific to their usage. Knowing where these artifacts are stored can assist us in re-creating that user account's experience.

With the advent of Windows Vista, Microsoft introduced the Application Data folder structure, which made it much easier for forensic examiners to determine which data belonged to the operating system and which data belonged to the user.

Local Folder

The AppData\Local folder contains data that does not roam with the user. The data that is stored here is usually too large to roam with the user. This was previously known as "Documents and Settings\%UserName%\Local Settings\Application Data" in Windows XP. Forensically interesting items that can be found here include temporary Internet files, Internet history, and several items that are new to Windows 8. The following chart contains locations that are of forensic interest in the Local folder. A majority of these locations will also work with Windows Vista and Windows 7 (unless noted with the Windows 8 icon, which is found above in the Icon Key).

ICON KEY	
🐟	Windows 8
✳	More info

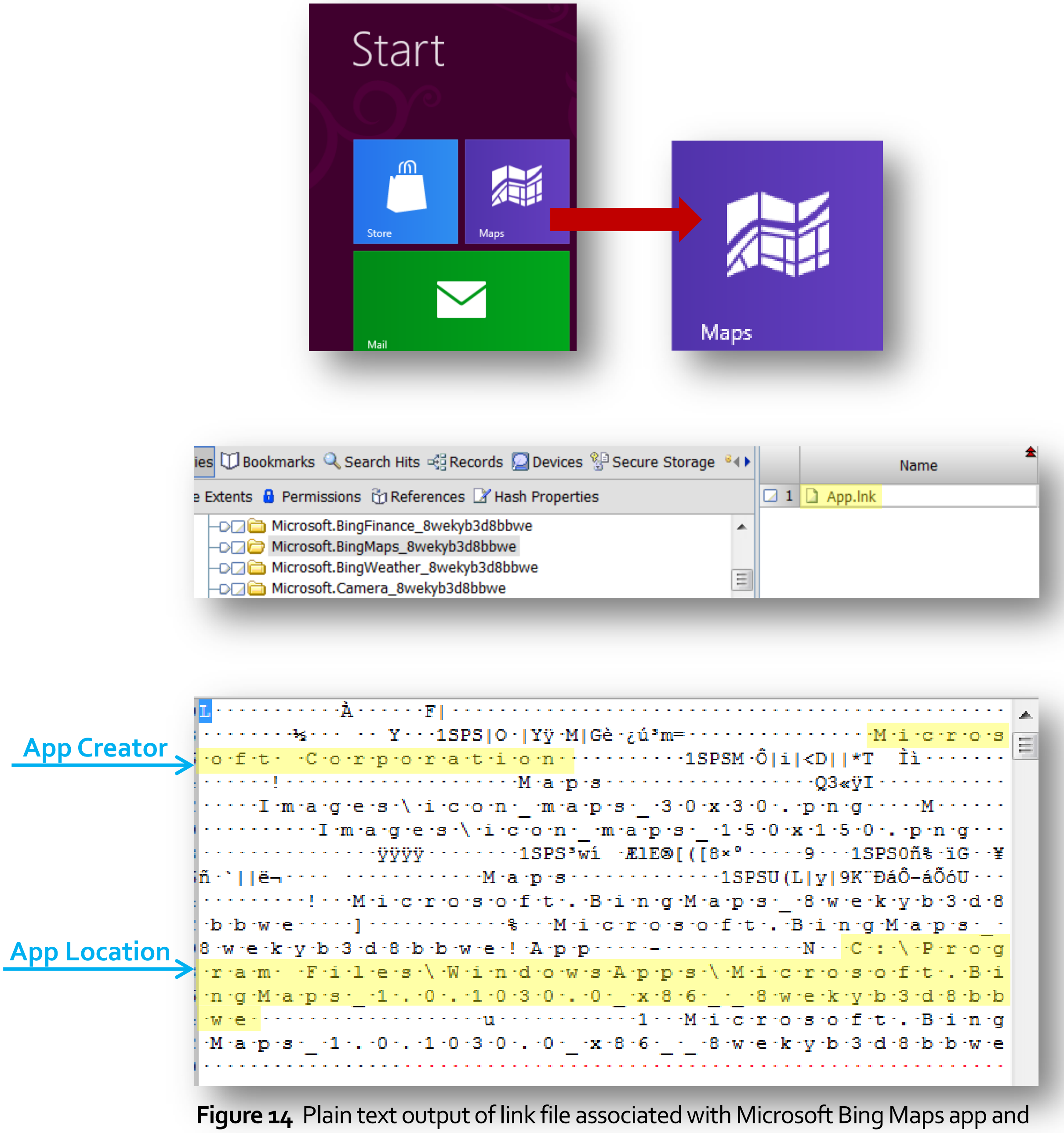
%Root%\Users\%User%\AppData\Local\

Application	Location	Purpose
Metro Apps	Microsoft\Windows\Application Shortcuts	Apps that are displayed on the Metro interface
IE 10 Websites Visited	Microsoft\InternetExplorer\Recovery\Immersive\Active AND Microsoft\InternetExplorer\Recovery\Immersive\Last Active	Websites user visited while browsing with IE10.
Taskbar Apps	Microsoft\Windows\Caches	Apps pinned to the Desktop
Journal Notes	Microsoft\Journal\Cache\msnb.dat	Contains a history of journal notes created by user and their location.
User-Added IE 10 Favorites	Microsoft\Windows\RoamingTiles	Websites the user has pinned to their favorites.
Internet History	Microsoft\Windows\History\History.IE5\MSHist01YYYYMMDD YYYYMMDD	User's Internet history. More research is needed as this contained empty "container.dat" files
Temporary Internet Files	Microsoft\Windows\Temporary Internet Files\Low\Content.IE5	Stores temporary Internet files
Protected Mode Temporary Internet Files	Microsoft\Windows\Temporary Internet Files\Virtualized\%Local Disk%\Users\%User%\AppData	Storage location of temporary Internet files when IE runs in Protected Mode (not to be confused with InPrivate Browsing)
Desktop	Microsoft\Windows\WinX	Contains link files for applications such as Device Manager, Command Prompt, and Run.

Application	Location	Purpose
Windows Sidebar Weather App	Microsoft\Windows\Windows Sidebar\Cache\168522d5-1082-4df2-b2f6-9185c31f9472	Contains a XML file with location name and zip code as file name. This file can contain location coordinates, date, and time. This class ID is the same for Vista/7/8.
Metro App Web Cache	Packages\%MetroAppName%\AC\INetCache	Contains web cache specific to Metro App.
Metro App Cookies	Packages\%MetroAppName%\AC\INetCookies	Contains cookie files specific to Metro App. Data is contained in a text file.
Metro App Web History	Packages\%MetroAppName%\AC\INetHistory	Contains Internet history files specific to Metro App and the format of the data is consistent with previous versions.
Metro Settings	Packages\%MetroAppName%\AC\LocalState	Contains settings specific to Metro App and can be viewed in plain text.

Metro Apps

Figure 14 demonstrates Metro Apps that are displayed on the Metro Desktop will have a link file associated with them that will display who created the app and the app's location. This data will be available in plain text. In this example, the Microsoft Bing Map App was used. The link file tells us that Microsoft is the creator of this app and it is stored under Program Files.



```
Microsoft.BingFinance_8wekyb3d8bbwe
Microsoft.BingMaps_8wekyb3d8bbwe
Microsoft.BingWeather_8wekyb3d8bbwe
Microsoft.Camera_8wekyb3d8bbwe
```

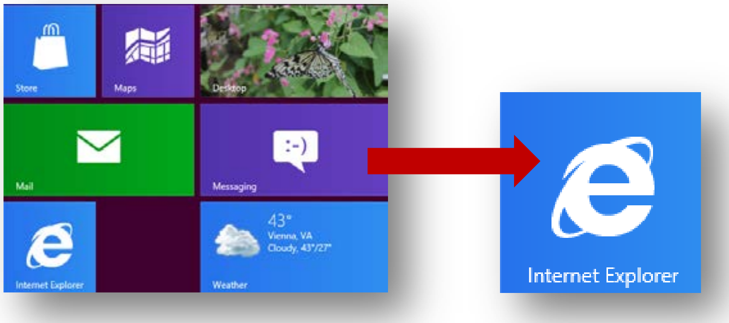
```
☐ 1  App.lnk
```

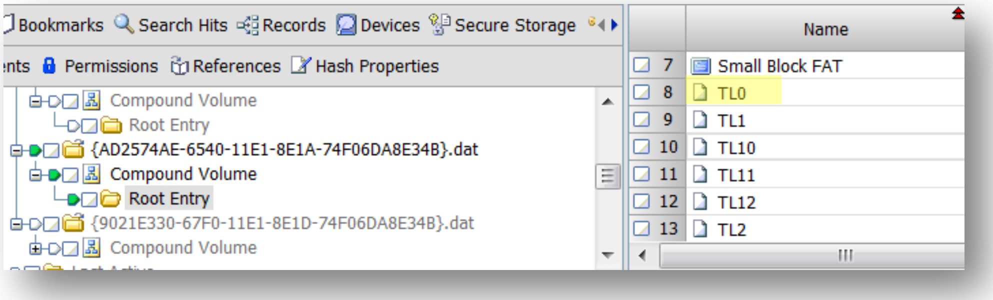
```
L···········À·····F|················································
·······¼···  ·· Y···1SPS|O·|Yÿ·M|Gè·¿ú³m=················M·i·c·r·o·s
·o·f·t· ·C·o·r·p·o·r·a·t·i·o·n··········1SPSM·Ô|i|<D||*T  Ìì·······
······!·························M·a·p·s····················Q3«ÿI·········
····I·m·a·g·e·s·\·i·c·o·n·_·m·a·p·s·_·3·0·x·3·0·.·p·n·g····M······
··········I·m·a·g·e·s·\·i·c·o·n·_·m·a·p·s·_·1·5·0·x·1·5·0·.·p·n·g···
···············ÿÿÿÿ·······1SPS³wí  ·ÆlE®[([8×°·····9···1SPS0ñ‰·ïG··¥
ñ·`||ë¬····  ············M·a·p·s··············1SPSU(L|y|9K¨ÐáÔ–áÕóU···
··········!···M·i·c·r·o·s·o·f·t·.·B·i·n·g·M·a·p·s·_·8·w·e·k·y·b·3·d·8
·b·b·w·e·····]·············%···M·i·c·r·o·s·o·f·t·.·B·i·n·g·M·a·p·s·_·
8·w·e·k·y·b·3·d·8·b·b·w·e·!·A·p·p·····–··············N···C·:·\·P·r·o·g
·r·a·m· ·F·i·l·e·s·\·W·i·n·d·o·w·s·A·p·p·s·\·M·i·c·r·o·s·o·f·t·.·B·i
·n·g·M·a·p·s·_·1·.·0·.·1·0·3·0·.·0·_·x·8·6·_·_·8·w·e·k·y·b·3·d·8·b·b
·w·e·················u··············1···M·i·c·r·o·s·o·f·t·.·B·i·n·g
·M·a·p·s·_·1·.·0·.·1·0·3·0·.·0·_·x·8·6·_·_·8·w·e·k·y·b·3·d·8·b·b·w·e
·····································································
```

Figure 14 Plain text output of link file associated with Microsoft Bing Maps app and its location

IE10 Websites Visited

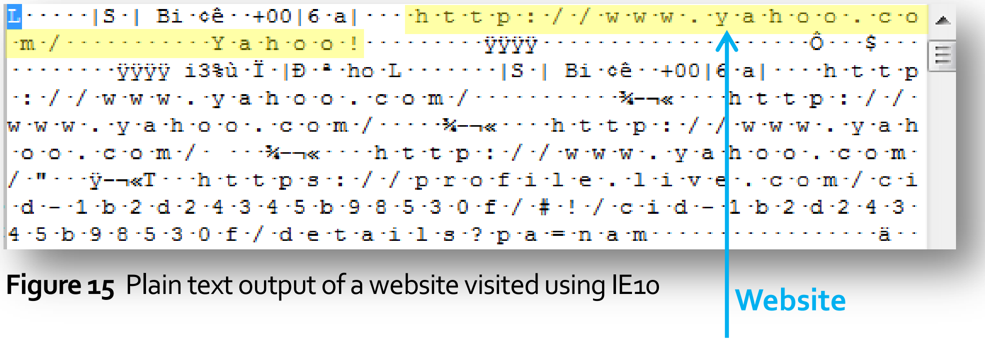
Figure 15 shows a Website I visited while browsing with IE10. These are found in compound DAT files with the file name similar to a Class ID. It is not know at this time if the file name is a Class ID as more research needs to be conducted. Once the file is unpacked, look for entries that are named "TL#". These are possibly known as "Travel Logs" and they contain the websites the user visited in plain text (some of the entry is in hex). The TL with the highest number is likely the oldest website visited.





```
Compound Volume
  Root Entry
{AD2574AE-6540-11E1-8E1A-74F06DA8E34B}.dat
  Compound Volume
    Root Entry
{9021E330-67F0-11E1-8E1D-74F06DA8E34B}.dat
  Compound Volume
```

	Name
7	Small Block FAT
8	TL0
9	TL1
10	TL10
11	TL11
12	TL12
13	TL2

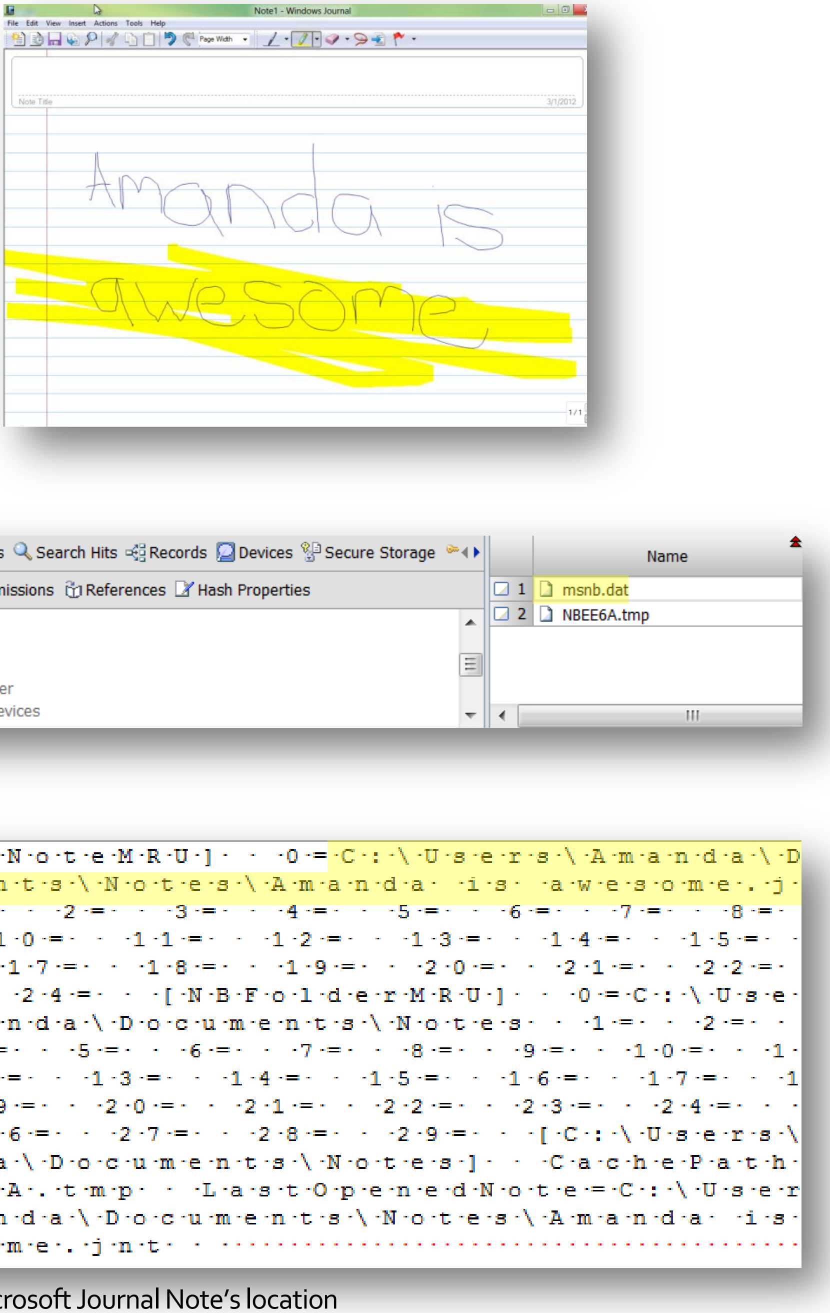


```
L·····|S·| Bi·¢ê··+00|6·a|···h·t·t·p·:·/·/·w·w·w·.·y·a·h·o·o·.·c·o
·m·/··········Y·a·h·o·o·!·········ÿÿÿÿ················Ô···$····
········ÿÿÿÿ i3¾ù·Ï·|Ð·ª·ho·L········|S·| Bi·¢ê··+00|6·a|····h·t·t·p
·:·/·/·w·w·w·.·y·a·h·o·o·.·c·o·m·/·············¾—¬«···h·t·t·p·:·/·/·
w·w·w·.·y·a·h·o·o·.·c·o·m·/·····¾—¬«···h·t·t·p·:·/·/·w·w·w·.·y·a·h
·o·o·.·c·o·m·/·  ···¾—¬«···h·t·t·p·:·/·/·w·w·w·.·y·a·h·o·o·.·c·o·m·
/·"···ÿ—¬«T···h·t·t·p·s·:·/·/·p·r·o·f·i·l·e·.·l·i·v·e·.·c·o·m·/·c·i
·d·–·1·b·2·d·2·4·3·4·5·b·9·8·5·3·0·f·/·#·!·/·c·i·d·–·1·b·2·d·2·4·3·
4·5·b·9·8·5·3·0·f·/·d·e·t·a·i·l·s·?·p·a·=·n·a·m··················ä··
```

Figure 15 Plain text output of a website visited using IE10

Journal Notes

Journal Notes is a program that came with Windows 7, but we will probably see greater use with Windows 8. This application maintains a DAT file that gives the stored location of Journal Notes (*Figure 16*). This information is in plain text. It is unknown at this time if other types of information are contained in this DAT file.



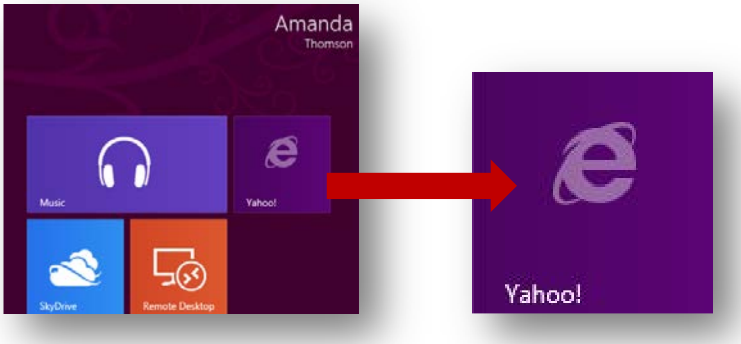
	Name
1	msnb.dat
2	NBEE6A.tmp

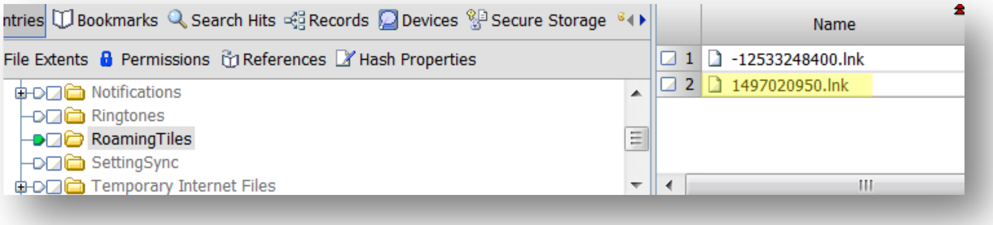
```
ÿþ · ·[·N·B·N·o·t·e·M·R·U·]· · ·0·=·C·:·\·U·s·e·r·s·\·A·m·a·n·d·a·\·D
·o·c·u·m·e·n·t·s·\·N·o·t·e·s·\·A·m·a·n·d·a· ·i·s· ·a·w·e·s·o·m·e·.·j·
n·t· · ·1·=· · ·2·=· · ·3·=· · ·4·=· · ·5·=· · ·6·=· · ·7·=· · ·8·=·
· ·9·=· · ·1·0·=· · ·1·1·=· · ·1·2·=· · ·1·3·=· · ·1·4·=· · ·1·5·=· ·
 ·1·6·=· · ·1·7·=· · ·1·8·=· · ·1·9·=· · ·2·0·=· · ·2·1·=· · ·2·2·=·
· ·2·3·=· · ·2·4·=· · ·[·N·B·F·o·l·d·e·r·M·R·U·]· · ·0·=·C·:·\·U·s·e·
r·s·\·A·m·a·n·d·a·\·D·o·c·u·m·e·n·t·s·\·N·o·t·e·s· · ·1·=· · ·2·=· ·
·3·=· · ·4·=· · ·5·=· · ·6·=· · ·7·=· · ·8·=· · ·9·=· · ·1·0·=· · ·1·
1·=· · ·1·2·=· · ·1·3·=· · ·1·4·=· · ·1·5·=· · ·1·6·=· · ·1·7·=· · ·1
·8·=· · ·1·9·=· · ·2·0·=· · ·2·1·=· · ·2·2·=· · ·2·3·=· · ·2·4·=· · ·
2·5·=· · ·2·6·=· · ·2·7·=· · ·2·8·=· · ·2·9·=· · ·[·C·:·\·U·s·e·r·s·\
·A·m·a·n·d·a·\·D·o·c·u·m·e·n·t·s·\·N·o·t·e·s·]· · ·C·a·c·h·e·P·a·t·h·
=·N·B·E·E·6·A·.·t·m·p· · ·L·a·s·t·O·p·e·n·e·d·N·o·t·e·=·C·:·\·U·s·e·r
·s·\·A·m·a·n·d·a·\·D·o·c·u·m·e·n·t·s·\·N·o·t·e·s·\·A·m·a·n·d·a· ·i·s·
 ·a·w·e·s·o·m·e·.·j·n·t· · ·······························
```

Figure 16 Microsoft Journal Note's location

IE10 Pinned Favorites

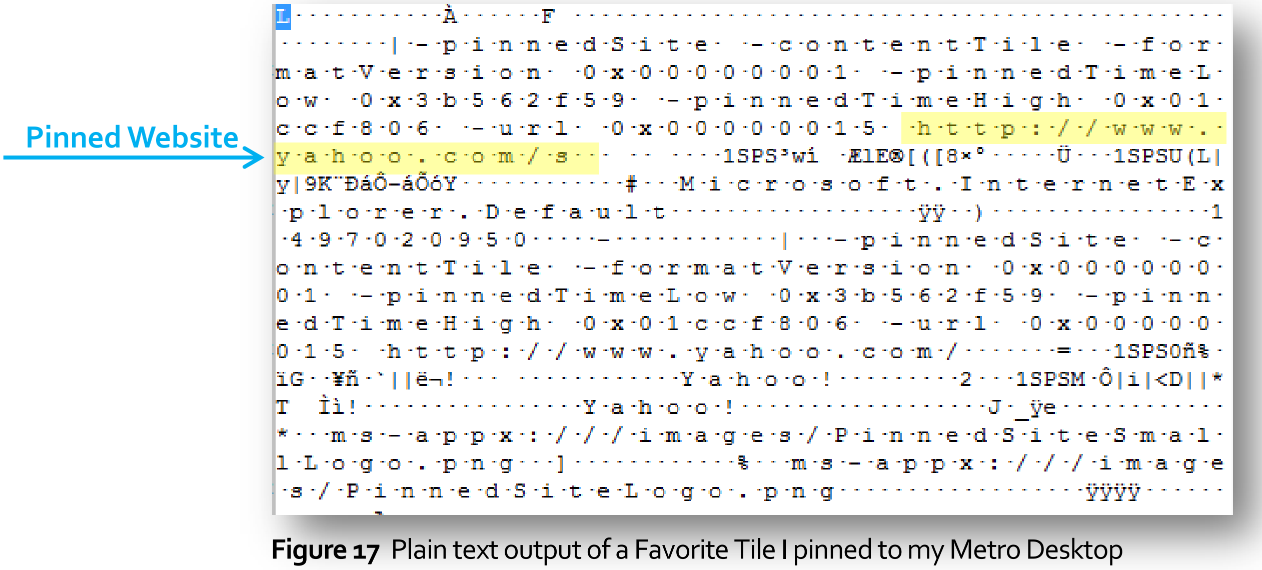
This section shows favorite websites I pinned to my Metro Desktop (*Figure 17*). For each Favorite, there is a corresponding link file. The file name of this link file is made up of several digits and it is unknown at this time as to how this file name is derived. The link file contains plain text output of the website the Favorite Tile belongs to.





```
Notifications
Ringtones
RoamingTiles
SettingSync
Temporary Internet Files
```

	Name
1	-12533248400.lnk
2	1497020950.lnk

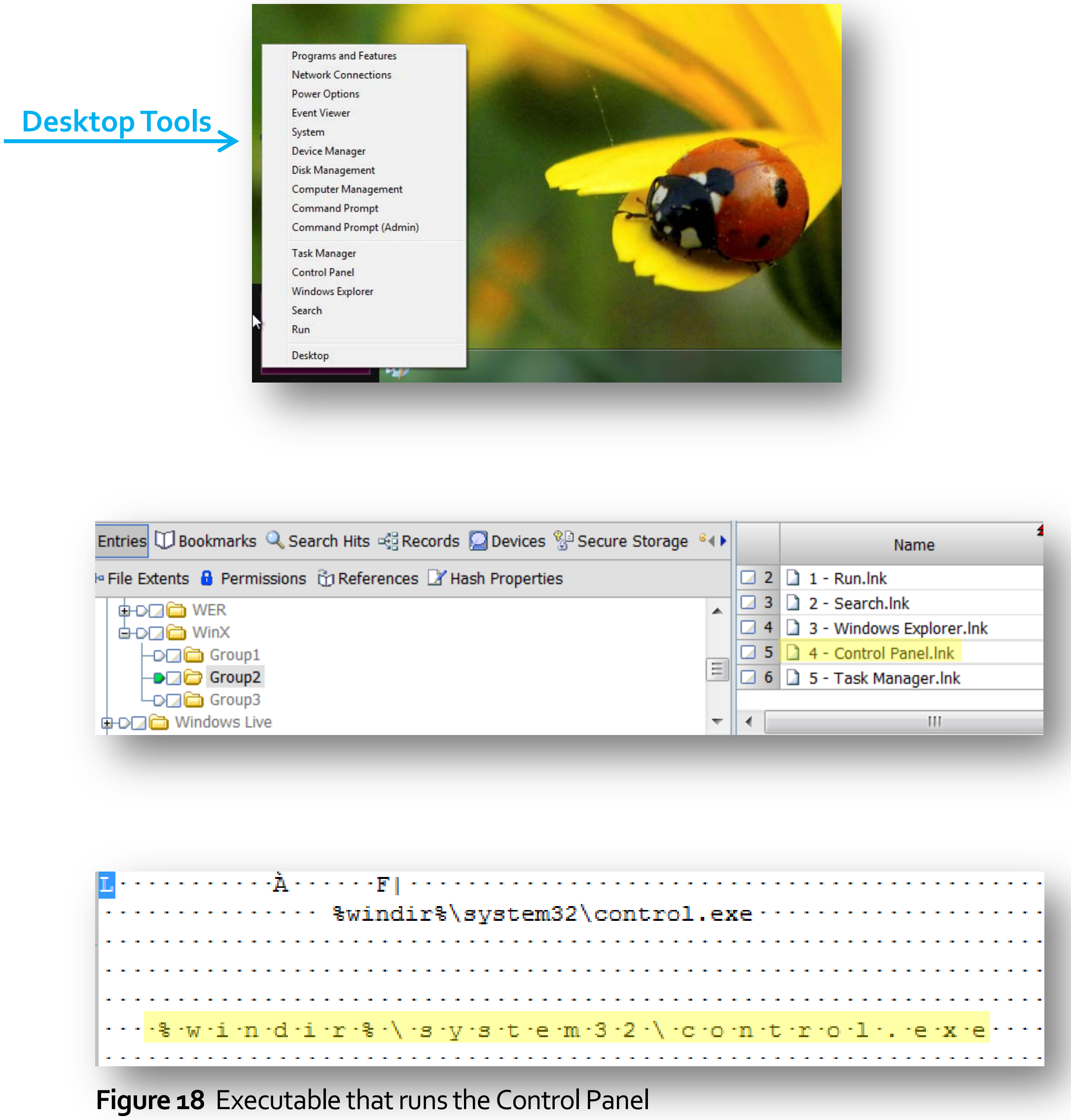


```
L············À······F ·············································
········|–·p·i·n·n·e·d·S·i·t·e· ·–·c·o·n·t·e·n·t·T·i·l·e· ·–·f·o·r·
m·a·t·V·e·r·s·i·o·n· ·0·x·0·0·0·0·0·0·0·1· ·–·p·i·n·n·e·d·T·i·m·e·L·
o·w· ·0·x·3·b·5·6·2·f·5·9· ·–·p·i·n·n·e·d·T·i·m·e·H·i·g·h· ·0·x·0·1·
c·c·f·8·0·6· ·–·u·r·l· ·0·x·0·0·0·0·0·0·1·5· ·h·t·t·p·:·/·/·w·w·w·.·
y·a·h·o·o·.·c·o·m·/·s··· ·· ····1SPS³wí ·Æ1E®[([8×°·····Ü···1SPSU(L|
y|9K¨ÐáÔ–áÕóY·············#···M·i·c·r·o·s·o·f·t·.·I·n·t·e·r·n·e·t·E·x
·p·l·o·r·e·r·.·D·e·f·a·u·l·t···················ÿÿ··)···············1
·4·9·7·0·2·0·9·5·0·····–··············|···–·p·i·n·n·e·d·S·i·t·e· ·–·c
o·n·t·e·n·t·T·i·l·e· ·–·f·o·r·m·a·t·V·e·r·s·i·o·n· ·0·x·0·0·0·0·0·0·
0·1· ·–·p·i·n·n·e·d·T·i·m·e·L·o·w· ·0·x·3·b·5·6·2·f·5·9· ·–·p·i·n·n·
e·d·T·i·m·e·H·i·g·h· ·0·x·0·1·c·c·f·8·0·6· ·–·u·r·l· ·0·x·0·0·0·0·0·
0·1·5· ·h·t·t·p·:·/·/·w·w·w·.·y·a·h·o·o·.·c·o·m·/·······=···1SPS0ñ%·
ïG··¥ñ·`||ë¬!··· ···········Y·a·h·o·o·!·········2···1SPSM·Ô|i|<D||*
T  Ìì!················Y·a·h·o·o·!··················J·_ÿe············
*···m·s·–·a·p·p·x·:·/·/·/·i·m·a·g·e·s·/·P·i·n·n·e·d·S·i·t·e·S·m·a·l·
l·L·o·g·o·.·p·n·g···]·············%···m·s·–·a·p·p·x·:·/·/·/·i·m·a·g·e
·s·/·P·i·n·n·e·d·S·i·t·e·L·o·g·o·.·p·n·g··················ÿÿÿÿ······
```

Figure 17 Plain text output of a Favorite Tile I pinned to my Metro Desktop

Desktop Tools

Desktop Tools is similar to the old Start Menu's Accessories and System Tools folders and is accessible by right-clicking on the task bar (*Figure 18*). They are broken down into three groups and each application in a group has their own link file that contains which executable runs that application. It is probable that a user could change the tool for a different application. Group 1 contains the Desktop. Group 2 consists of the Run command, Search, Windows Explorer, Control Panel, and Task Manager. Group 3 is made up of Run as Administrator Command Prompt, Command Prompt, Computer Management, Disk Management, Device Manager, System, Event Viewer, Power Options, Network Connections, and Programs and Features.



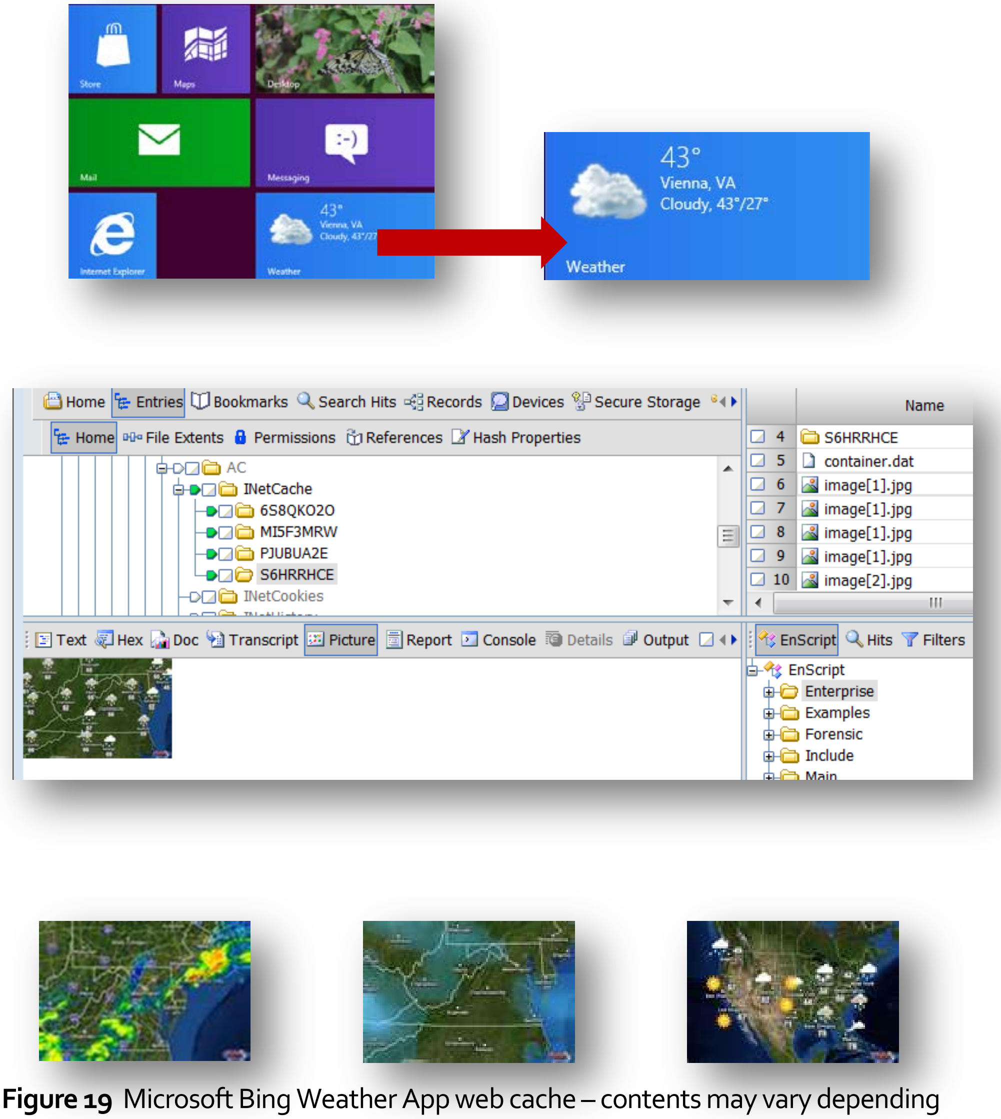
```
2  1 - Run.lnk
3  2 - Search.lnk
4  3 - Windows Explorer.lnk
5  4 - Control Panel.lnk
6  5 - Task Manager.lnk
```

```
L··········À·····F|·········································
··············%windir%\system32\control.exe·················
····························································
····························································
····························································
···%·w·i·n·d·i·r·%·\·s·y·s·t·e·m·3·2·\·c·o·n·t·r·o·l·.·e·x·e····
····························································
```

Figure 18 Executable that runs the Control Panel

Metro App Web Cache

Everything is connected to the Internet with a Windows Live Account and each app is considered to be what Windows calls an "immersive" environment. This means that from within each app, you can access other apps, so essentially, that app becomes the operating system. As a result of this immersive concept, each app will have its own Internet artifacts. *Figure* 19 shows web cache for the Microsoft Bing Weather App.



```
AC
  INetCache
    6S8QKO2O
    MI5F3MRW
    PJUBUA2E
    S6HRRHCE
  INetCookies
```

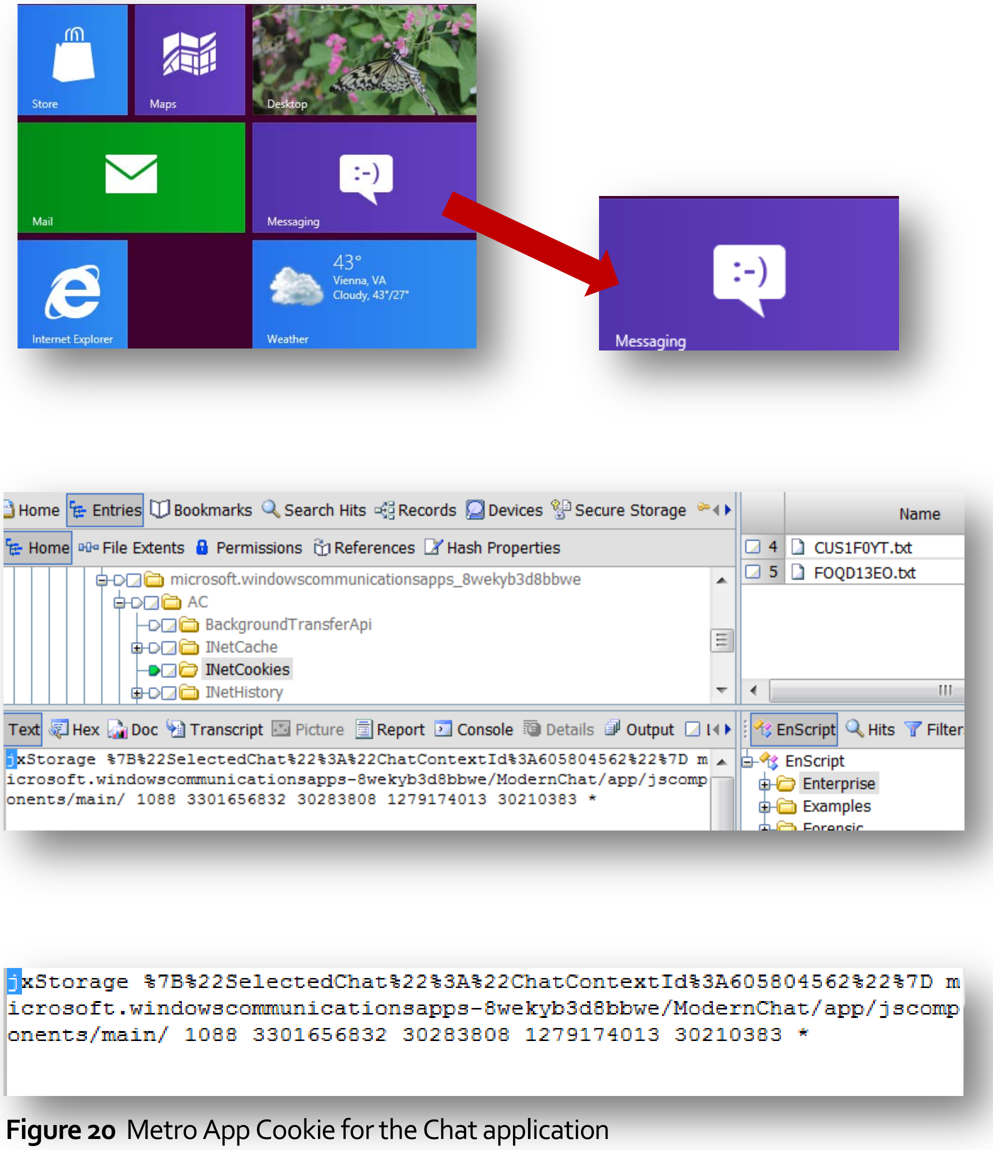
	Name
4	S6HRRHCE
5	container.dat
6	image[1].jpg
7	image[1].jpg
8	image[1].jpg
9	image[1].jpg
10	image[2].jpg

```
EnScript
  Enterprise
  Examples
  Forensic
  Include
  Main
```

Figure 19 Microsoft Bing Weather App web cache – contents may vary depending on the application

Metro App Cookies

Cookies can also be found for each Metro App. *Figure* 20 shows the cookies are text files and the content of a cookie found here is similar to any other cookie content you might come across.



```
jxStorage %7B%22SelectedChat%22%3A%22ChatContextId%3A605804562%22%7D m
icrosoft.windowscommunicationsapps-8wekyb3d8bbwe/ModernChat/app/jscomp
onents/main/ 1088 3301656832 30283808 1279174013 30210383 *
```

```
jxStorage %7B%22SelectedChat%22%3A%22ChatContextId%3A605804562%22%7D m
icrosoft.windowscommunicationsapps-8wekyb3d8bbwe/ModernChat/app/jscomp
onents/main/ 1088 3301656832 30283808 1279174013 30210383 *
```

Figure 20 Metro App Cookie for the Chat application

Roaming Folder

The AppData\Roaming folder is independent of the computer and holds data that is specific to the application and roams with the user's profile. In Windows XP, this data was contained in Documents and Settings\%UserName%\Application Data. Artifacts that are of use to us that are found here include applications pinned to the Task Bar, cookies, and Internet Explorer downloads history.

%Root%\Users\%User%\AppData\Roaming\

Application	Location	Purpose
Credentials	Credentials	Can contain data used by EFS[v].
RSA-based Certificates	Crypto\RSA	Contains private keys for Microsoft RSAbased CSPs. Also see "Master Key"[vi].
Pinned to Task Bar	Internet Explorer\Quick Launch\User Pinned\TaskBar	Applications the user pinned to their task bar. Data is contained in a link file.
Master Key	Protect\%SID%	Used to encrypt the user's private key. Contains the user's Master Key, which contains the Password Key and the backup/restore form for the Master Key. Data is encrypted twice .
User's Credentials	Vault	Credentials that are used to automatically logon the user to Websites, servers, and programs[vii].
Cookies	Windows\Cookies\Low	Internet cookies with data contained in text files.

Application	Location	Purpose
IE Compatibility Mode Cache	Windows\IECompatCache\Low	Contains cache data when IE uses Compatibility Mode.
IE Compatibility UA Cache	Windows\IECompatUACache\Low	Unknown at this time.
IE Download History	Windows\IEDownloadHistory	Contains a history of files the user downloaded.
IE Top Level Domain Cache	Windows\IETldCache	Contains TLDs – user could add "TLDs" that may not necessarily be recognized as TLDs. File format data is stored in is unknown at this time.
Libraries	Windows\Libraries	Contains info on Documents, Music, Pictures, etc. and whether library is pinned, the owner's SID, and the class ID of the folder. Data is contained in XML format.
Logon	Windows\Logon	Unknown at this time
Network Shortcuts	Windows\Network Shortcuts	Contains servers user accessed and could also contain information about user's internal network. Data output is unknown at this time.
Printer Shortcuts	Windows\Printer Shortcuts	Contains shortcuts to printers the user has added. Data output is unknown at this time.

Application	Location	Purpose
InPrivate Filtering	Windows\PrivacIE\Low	Stores URLs to third party content[viii]. This is different from InPrivate Browsing.
Recent User Activity	Windows\Recent\AutomaticDestinations	Data is stored in compound files similar to this format: “0-9&a-z.AutomaticDestinations-ms”. Can contain information on user’s web activity, files copied, and files created. Files within compound files have the following structure: “1”, “2”, “3”…, “a”, “b”, “c”…

Communications App

Windows 8 Metro Apps individually store forensically useful information about the user's activity[ix]. This can be quite handy for forensic examiners as it gives us another place to look for data (or the absence thereof) if the user tried to cover their tracks.

The Communications App basically includes the user's e-mail, chat clients, Facebook, and other social networking sites. Anything that can allow the user to interact with another person appears to fall under "Communications Apps".

Cache

Similar to the other Apps, the Communications App maintains its own web cache, which can be found in the following location[x]:

Application	Location	Purpose
Communication App Web Cache	%Root%\Users\%User%\AppData\Local\Packages\microsoft.windowscommunicatisapps_8wekyb3d8bbwe\AC\INetCache	Contains items such as pictures from profiles the user viewed

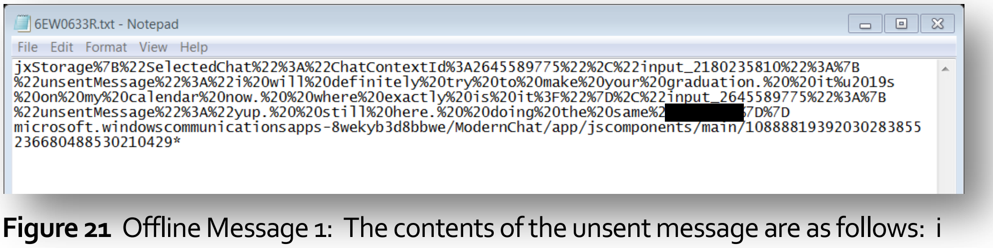
The cache that is found in this location appears to be data that is specific to a website that was viewed through the Communications App. In this case, it was all Facebook pictures, to include profile pictures and pictures that were on that account's page.

Cookies

The Communications App also has cookies. In the test image I used for examining Windows 8, a cookie was found that contained the offline content of a Facebook chat conversation between a friend and I who had just gone offline while I was responding. The following location contains cookies for the Communications App:

Application	Location	Purpose
Communication App Cookies	%Root%\Users\%User%\AppData\Local\Packages\microsoft.windows communicatisapps_8wekyb3d8bbwe\AC\INetCookies	Contains cookies for applications connected with the Communications App

Figure 21 is an example of what could be found in this location. In this image, there was a cookie that contained the contents of two offline messages:



```
jxStorage%7B%22SelectedChat%22%3A%22ChatContextId%3A2645589775%22%2C%22input_2180235810%22%3A%7B
%22unsentMessage%22%3A%22i%20will%20definitely%20try%20to%20make%20your%20graduation.%20%20it%u2019s
%20on%20my%20calendar%20now.%20%20where%20exactly%20is%20it%3F%22%7D%2C%22input_2645589775%22%3A%7B
%22unsentMessage%22%3A%22yup.%20%20still%20here.%20%20doing%20the%20same%2[redacted]7D%7D
microsoft.windowscommunicationsapps-8wekyb3d8bbwe/ModernChat/app/jscomponents/main/10888819392030283855
236680488530210429*
```

Figure 21 Offline Message 1: The contents of the unsent message are as follows: i will definitely try to make your graduation. it's on my calendar now. where exactly is it?

The offline messages that were found in this cookie file were typed by me, who is the user associated with the Communications App. In this cookie, it is known who the conversation was between, but outside of a testing environment, it is unknown if the contact the user was communicating with can be identified. More research should be conducted on this.

Microsoft Folder

Digital Certificates

The Microsoft folder stores digital certificates, which are used to authenticate clients and servers when surfing the Internet or sending e-mails. This ensures that communications are secure and helps maintain data integrity. How do they work? See *Figure 22* if you're aware of their existence but unfamiliar with how digital certificates work.

Digital certificates link the certificate owner's identity to a public key and a private key. These act like credentials to authenticate the sender and receiver. If a sender encrypts a message with their private key, then the receiver must decrypt the message with their public key.

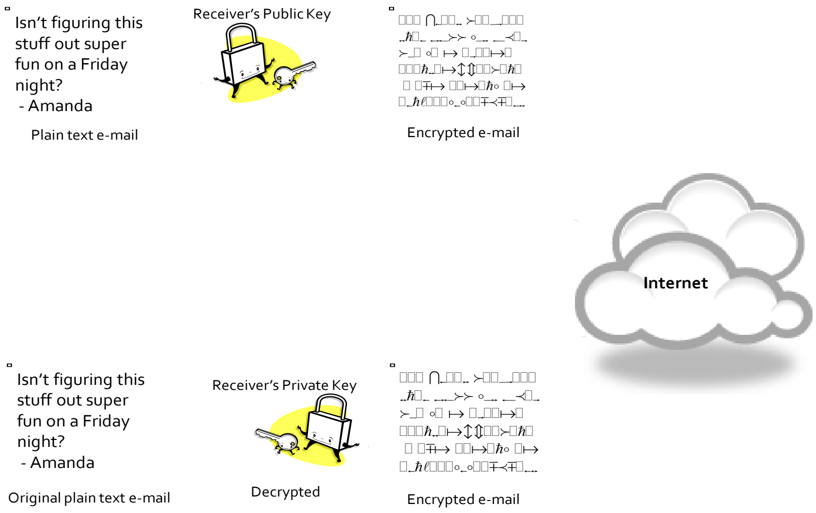

Isn't figuring this stuff out super fun on a Friday night?
- Amanda
Plain text e-mail
Receiver's Public Key
Encrypted e-mail
Internet
Isn't figuring this stuff out super fun on a Friday night?
- Amanda
Original plain text e-mail
Receiver's Private Key
Decrypted
Encrypted e-mail

Figure 22 How digital certificates work

At a minimum, digital certificates must contain the following information:

- Owner's public key
- Owner's name/alias
- Expiration date of the certificate
- Serial number of the certificate
- Organization that issued the certificate
- Digital signature of issuing organization[xi]

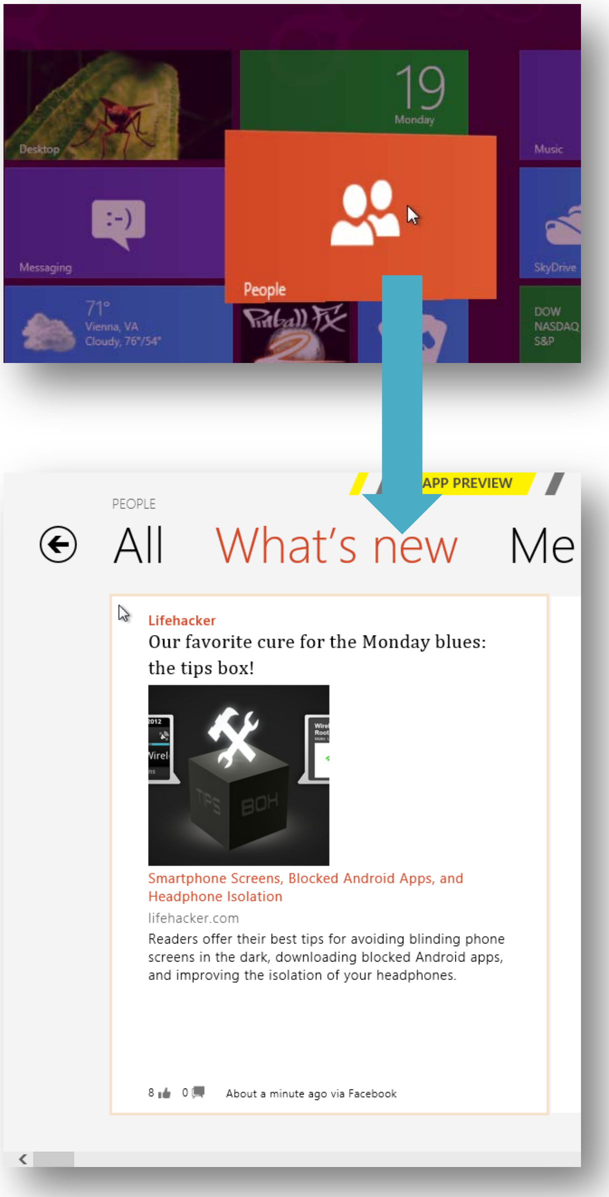
A majority of this information will be in plain text.

The following location contains digital certificates for the Communications App:

Application	Location	Purpose
Communication App Digital Certificates	%Root%\Users\%User%\AppData\Local\Packages\microsoft.windowscommunicationsapps_8wekyb3d8bbwe\AC\Microsoft\CryptnetURLCache\Content	Contains certificates for the Communications App

What's New

This same folder, Microsoft, also contains a subfolder called "Internet Explorer", which contained updates for the user called "What's New". An example of what the user will see is found in *Figure 23*[xii]:



Lifehacker

Our favorite cure for the Monday blues: the tips box!

Smartphone Screens, Blocked Android Apps, and Headphone Isolation

lifehacker.com

Readers offer their best tips for avoiding blinding phone screens in the dark, downloading blocked Android apps, and improving the isolation of your headphones.

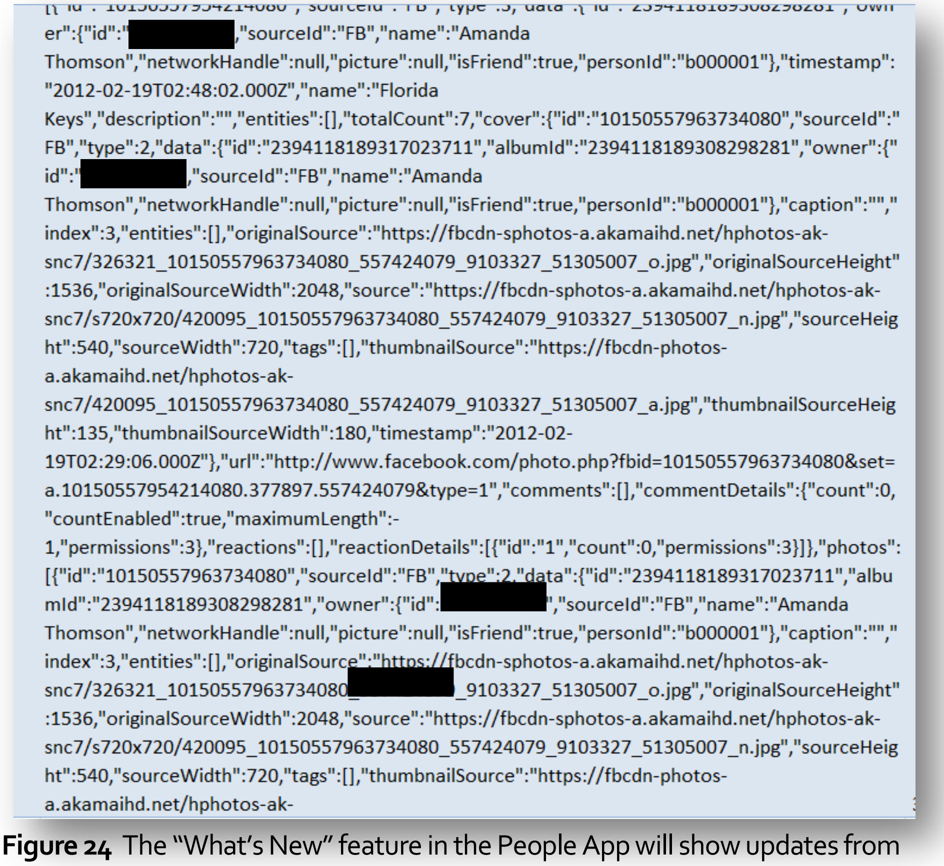
8 0 About a minute ago via Facebook

Figure 23 The People App will allow the user to see updates, such as those belonging to Facebook

The following is the location in which this data can be viewed:

Application	Location	Purpose
User's "What's New" Updates	%Root%\Users\%User%\AppData\Local\Packages\microsoft.windowscommunicationsapps_8wekyb3d8bbwe\AC\Microsoft\Internet Explorer\DOMStore\%History-Folder%\ microsoft[#].xml	Contains contact info, such as e-mail addresses, physical addresses, phone numbers, etc.

Facebook information showed up in the test image I created and included pictures that were uploaded. Information found in the location in the above table includes date and time, whether the person is a "friend", and the file uploaded. *Figure 24*is an example of what could be contained in the XML file[xiii]:



```
[{"id":"10150557954214080","sourceId":"FB","type":3,"data":{"id":"2394118189308298281","own
er":{"id":"[redacted],"sourceId":"FB","name":"Amanda
Thomson","networkHandle":null,"picture":null,"isFriend":true,"personId":"b000001"},"timestamp":
"2012-02-19T02:48:02.000Z","name":"Florida
Keys","description":"","entities":[],"totalCount":7,"cover":{"id":"10150557963734080","sourceId":"
FB","type":2,"data":{"id":"2394118189317023711","albumId":"2394118189308298281","owner":{"
id":"[redacted],"sourceId":"FB","name":"Amanda
Thomson","networkHandle":null,"picture":null,"isFriend":true,"personId":"b000001"},"caption":"","
index":3,"entities":[],"originalSource":"https://fbcdn-sphotos-a.akamaihd.net/hphotos-ak-
snc7/326321_10150557963734080_557424079_9103327_51305007_o.jpg","originalSourceHeight"
:1536,"originalSourceWidth":2048,"source":"https://fbcdn-sphotos-a.akamaihd.net/hphotos-ak-
snc7/s720x720/420095_10150557963734080_557424079_9103327_51305007_n.jpg","sourceHeig
ht":540,"sourceWidth":720,"tags":[],"thumbnailSource":"https://fbcdn-photos-
a.akamaihd.net/hphotos-ak-
snc7/420095_10150557963734080_557424079_9103327_51305007_a.jpg","thumbnailSourceHeig
ht":135,"thumbnailSourceWidth":180,"timestamp":"2012-02-
19T02:29:06.000Z"},"url":"http://www.facebook.com/photo.php?fbid=10150557963734080&set=
a.10150557954214080.377897.557424079&type=1","comments":[],"commentDetails":{"count":0,
"countEnabled":true,"maximumLength":-
1,"permissions":3},"reactions":[],"reactionDetails":[{"id":"1","count":0,"permissions":3}]},"photos":
[{"id":"10150557963734080","sourceId":"FB","type":2,"data":{"id":"2394118189317023711","albu
mId":"2394118189308298281","owner":{"id":[redacted]","sourceId":"FB","name":"Amanda
Thomson","networkHandle":null,"picture":null,"isFriend":true,"personId":"b000001"},"caption":"","
index":3,"entities":[],"originalSource":"https://fbcdn-sphotos-a.akamaihd.net/hphotos-ak-
snc7/326321_10150557963734080[redacted]_9103327_51305007_o.jpg","originalSourceHeight"
:1536,"originalSourceWidth":2048,"source":"https://fbcdn-sphotos-a.akamaihd.net/hphotos-ak-
snc7/s720x720/420095_10150557963734080_557424079_9103327_51305007_n.jpg","sourceHeig
ht":540,"sourceWidth":720,"tags":[],"thumbnailSource":"https://fbcdn-photos-
a.akamaihd.net/hphotos-ak-
```

Figure 24 The "What's New" feature in the People App will show updates from Facebook

E-mail

Windows 8 maintains several artifacts pertaining to e-mail that were in the user's account if the account was linked to the Communications app. One such artifact is streams, which "contain the data that is written to a file, and that gives more information about a file than attributes and properties"[xiv]. The streams that are located here contain the sender's name, the sender's e-mail address, the e-mail's subject, the name of any attachments, the receiver's name, and the receiver's e-mail address.

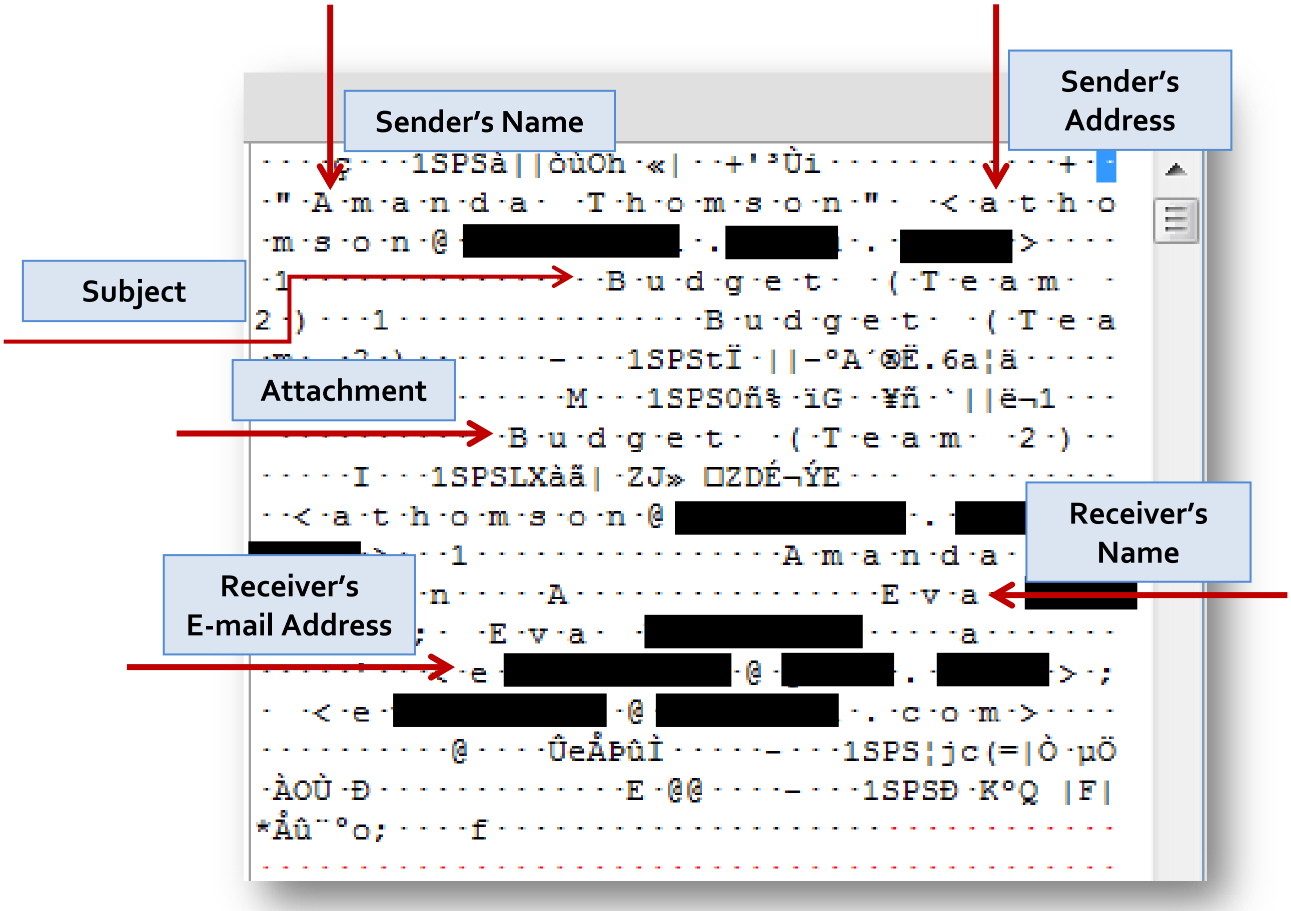
The streams in this image had the following naming convention:

12000001-9/a-f_##################.eml.OECustomProperty
(18 digits)

The following location contains streams:

Application	Location	Purpose
E-mail Streams from User's Communications App	%Root%\Users\%User%\AppData\Local\Packages\microsoft.windowscommunicationsapps_8wekyb3d8bbwe\LocalState\Indexed\LiveComm\%User'sWindowsLiveAccount%\%AppCurrentVersion%\Mail	Contains contact info, such as e-mail addresses, physical addresses, phone numbers, etc.

Figure 25 is an example of the data you may find in a file ending in "eml.OECustomProperty".



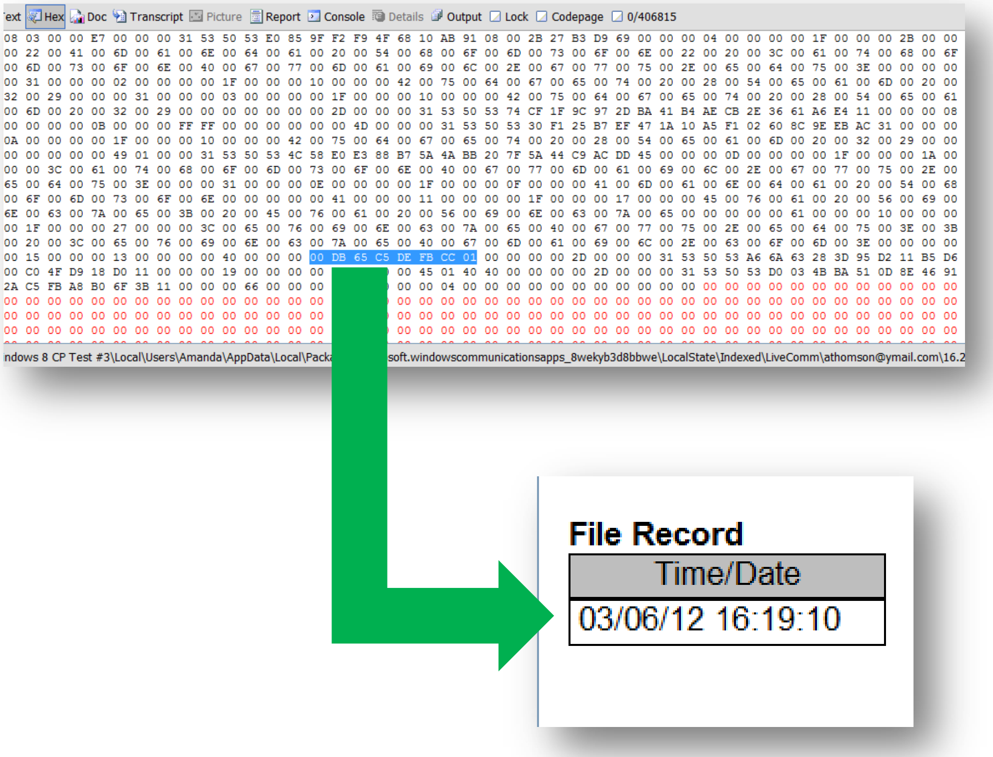
```
····Ç···1SPSà||òùOh·«|··+'³Ùi·············+·
·"·A·m·a·n·d·a· ·T·h·o·m·s·o·n·"· ·<·a·t·h·o
·m·s·o·n·@·[redacted]·.·[redacted]·.·[redacted]>····
·1················B·u·d·g·e·t· ·(·T·e·a·m· ·
2·)···1················B·u·d·g·e·t· ·(·T·e·a
·m· ·2·)··········–···1SPStÏ·||–°A´®Ë.6a¦ä·····
·········M···1SPSOñ%·ïG··¥ñ·`||ë¬1···
·············B·u·d·g·e·t· ·(·T·e·a·m· ·2·)··
·····I···1SPSLXàã|·ZJ» □ZDÉ¬ÝE··· ·········
··<·a·t·h·o·m·s·o·n·@·[redacted]·.·[redacted]
[redacted]>···1················A·m·a·n·d·a·
·n·····A················E·v·a·[redacted]
;· ·E·v·a· ·[redacted]·····a·······
·········<·e·[redacted]·@·[redacted]·.·[redacted]>·;
· ·<·e·[redacted]·@·[redacted]·.·c·o·m·>····
··········@····ÛeÅÞûÌ·····–···1SPS¦jc(=|Ò·µÖ
·ÀOÙ·Ð·············E·@@····–···1SPSÐ·K°Q |F|
*Åû¨°o;····f··························
······································
```

Figure 25 An example of the data that may be found in the streams from the user's email account(s)

The name of the stream used in *Figure 25* is:

1200012f_1297555571580314877.eml·OECustomProperty

Another interesting find is that in all of these streams I found that the time and date the e-mail was sent or received is contained in this data. It appears the date and time is always 106 bytes from the *end* of the stream. The date is contained as Windows FILETIME in the time zone that was set on the user's system. See *Figure 26* for an example.

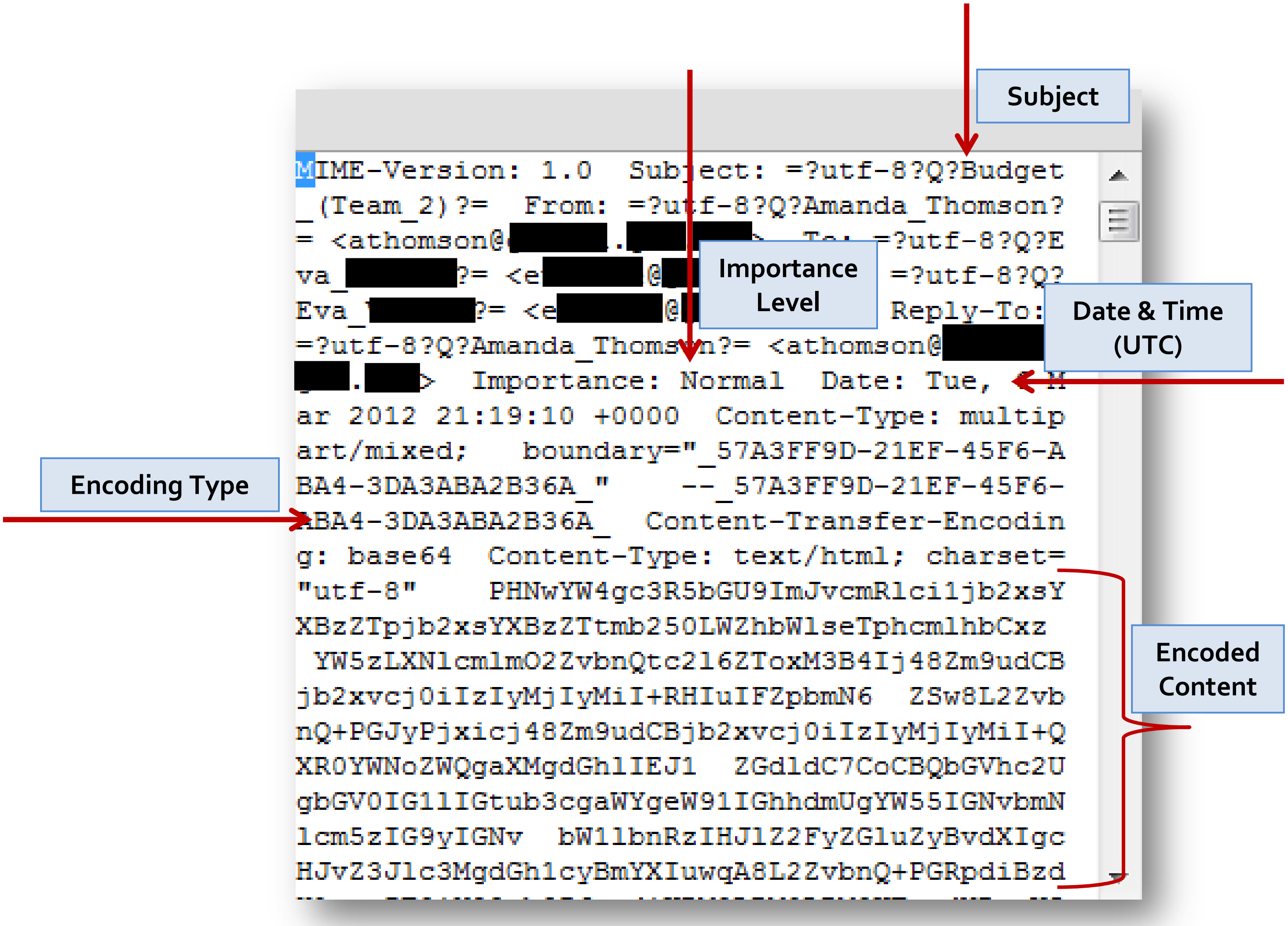


```
08 03 00 00 E7 00 00 00 31 53 50 53 E0 85 9F F2 F9 4F 68 10 AB 91 08 00 2B 27 B3 D9 69 00 00 00 04 00 00 00 00 1F 00 00 00 2B 00 00
00 22 00 41 00 6D 00 61 00 6E 00 64 00 61 00 20 00 54 00 68 00 6F 00 6D 00 73 00 6F 00 6E 00 22 00 20 00 3C 00 61 00 74 00 68 00 6F
00 6D 00 73 00 6F 00 6E 00 40 00 67 00 77 00 6D 00 61 00 69 00 6C 00 2E 00 67 00 77 00 75 00 2E 00 65 00 64 00 75 00 3E 00 00 00 00
00 31 00 00 00 02 00 00 00 00 1F 00 00 00 10 00 00 00 42 00 75 00 64 00 67 00 65 00 74 00 20 00 28 00 54 00 65 00 61 00 6D 00 20 00
32 00 29 00 00 00 31 00 00 00 03 00 00 00 00 1F 00 00 00 10 00 00 00 42 00 75 00 64 00 67 00 65 00 74 00 20 00 28 00 54 00 65 00 61
00 6D 00 20 00 32 00 29 00 00 00 00 00 00 00 2D 00 00 00 31 53 50 53 74 CF 1F 9C 97 2D BA 41 B4 AE CB 2E 36 61 A6 E4 11 00 00 00 08
00 00 00 00 0B 00 00 00 FF FF 00 00 00 00 00 00 4D 00 00 00 31 53 50 53 30 F1 25 B7 EF 47 1A 10 A5 F1 02 60 8C 9E EB AC 31 00 00 00
0A 00 00 00 00 1F 00 00 00 10 00 00 00 42 00 75 00 64 00 67 00 65 00 74 00 20 00 28 00 54 00 65 00 61 00 6D 00 20 00 32 00 29 00 00
00 00 00 00 00 49 01 00 00 31 53 50 53 4C 58 E0 E3 88 B7 5A 4A BB 20 7F 5A 44 C9 AC DD 45 00 00 00 0D 00 00 00 00 1F 00 00 00 1A 00
00 00 3C 00 61 00 74 00 68 00 6F 00 6D 00 73 00 6F 00 6E 00 40 00 67 00 77 00 6D 00 61 00 69 00 6C 00 2E 00 67 00 77 00 75 00 2E 00
65 00 64 00 75 00 3E 00 00 00 31 00 00 00 0E 00 00 00 00 1F 00 00 00 0F 00 00 00 41 00 6D 00 61 00 6E 00 64 00 61 00 20 00 54 00 68
00 6F 00 6D 00 73 00 6F 00 6E 00 00 00 00 00 41 00 00 00 11 00 00 00 00 1F 00 00 00 17 00 00 00 45 00 76 00 61 00 20 00 56 00 69 00
6E 00 63 00 7A 00 65 00 3B 00 20 00 45 00 76 00 61 00 20 00 56 00 69 00 6E 00 63 00 7A 00 65 00 00 00 00 00 61 00 00 00 10 00 00 00
00 1F 00 00 00 27 00 00 00 3C 00 65 00 76 00 69 00 6E 00 63 00 7A 00 65 00 40 00 67 00 77 00 75 00 2E 00 65 00 64 00 75 00 3E 00 3B
00 20 00 3C 00 65 00 76 00 69 00 6E 00 63 00 7A 00 65 00 40 00 67 00 6D 00 61 00 69 00 6C 00 2E 00 63 00 6F 00 6D 00 3E 00 00 00 00
00 15 00 00 00 13 00 00 00 00 40 00 00 00 00 DB 65 C5 DE FB CC 01 00 00 00 00 2D 00 00 00 31 53 50 53 A6 6A 63 28 3D 95 D2 11 B5 D6
00 C0 4F D9 18 D0 11 00 00 00 19 00 00 00 00        0 00 45 01 40 40 00 00 00 00 2D 00 00 00 31 53 50 53 D0 03 4B BA 51 0D 8E 46 91
2A C5 FB A8 B0 6F 3B 11 00 00 00 66 00 00 00        0 00 00 04 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00        0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00        0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00        0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
ndows 8 CP Test #3\Local\Users\Amanda\AppData\Local\Packa          soft.windowscommunicationsapps_8wekyb3d8bbwe\LocalState\Indexed\LiveComm\athomson@ymail.com\16.2
```

File Record
Time/Date
03/06/12 16:19:10

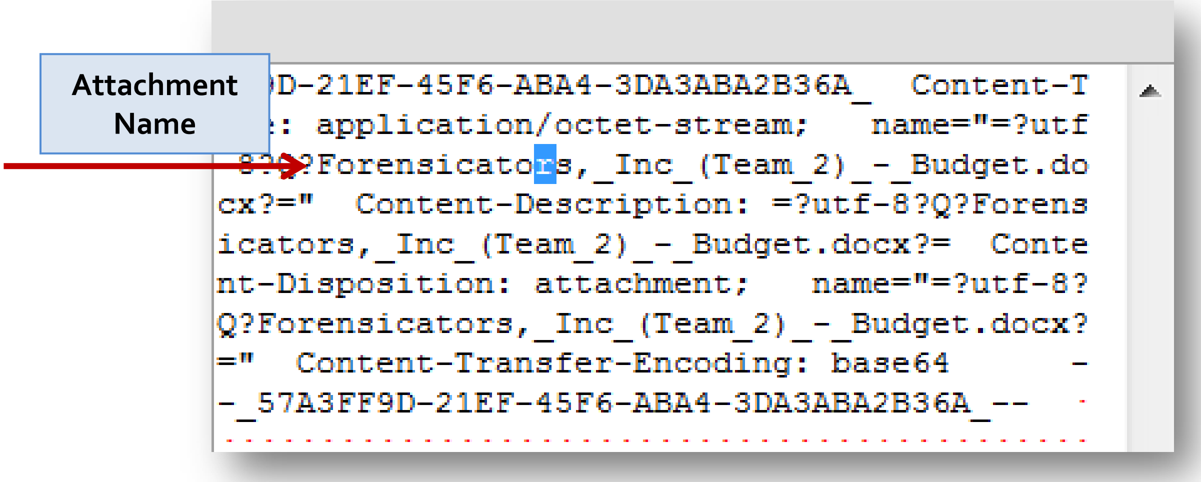
Figure 26 This is the hex output of the stream. 106 bytes from the end of the stream is a group of 8 bytes, which are the date and time for this stream.

The name of the e-mail stream is the same name as the EML file, which contains the content of the e-mail. The EML file in *Figure 25* is named “1200012f_129755557158031487.eml”. The name of the stream is “1200012f_129755557158031487.eml·OECustomProperty”[xv]. The data you will see in this file is the subject, the sender’s name, the sender’s e-mail addres, the receiver’s name, the receiver’s e-mail address, the importance level, and the date and time (UTC +0000) in ASCII (*Figure 27*). The end of the EML file will contain the name of any attachments (*Figure 28*). The EML file also contains the content of the message; however, it needs to be converted as the content of the e-mail is Base64 Encoded (*Figure 29*).



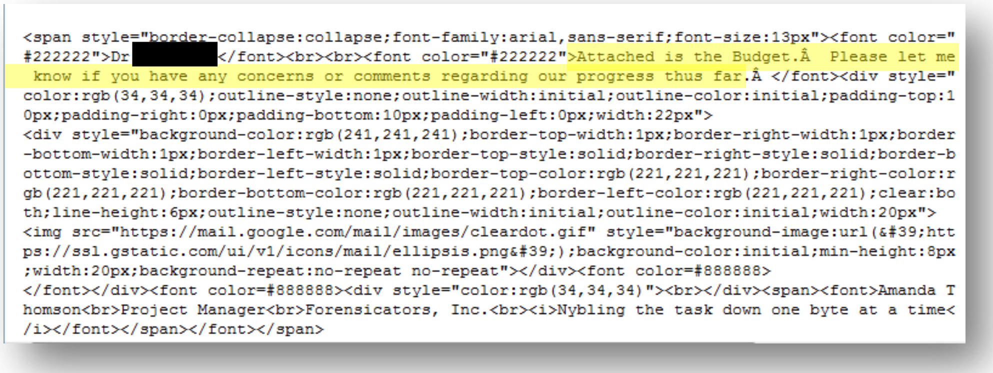
```
MIME-Version: 1.0  Subject: =?utf-8?Q?Budget
_(Team_2)?=  From: =?utf-8?Q?Amanda_Thomson?
= <athomson@[redacted].[redacted]>  To: =?utf-8?Q?E
va_[redacted]?= <e[redacted]@[redacted]        =?utf-8?Q?
Eva_[redacted]?= <e[redacted]@[redacted]       Reply-To:
=?utf-8?Q?Amanda_Thomson?= <athomson@[redacted]
[redacted].[redacted]>  Importance: Normal  Date: Tue,   M
ar 2012 21:19:10 +0000  Content-Type: multip
art/mixed;   boundary="_57A3FF9D-21EF-45F6-A
BA4-3DA3ABA2B36A_"    --_57A3FF9D-21EF-45F6-
ABA4-3DA3ABA2B36A_  Content-Transfer-Encodin
g: base64  Content-Type: text/html; charset=
"utf-8"    PHNwYW4gc3R5bGU9ImJvcmRlci1jb2xsY
XBzZTpjb2xsYXBzZTtmb250LWZhbWlseTphcmlhbCxz
 YW5zLXNlcmlmO2ZvbnQtc2l6ZToxM3B4Ij48Zm9udCB
jb2xvcj0iIzIyMjIyMiI+RHIuIFZpbmN6  ZSw8L2Zvb
nQ+PGJyPjxicj48Zm9udCBjb2xvcj0iIzIyMjIyMiI+Q
XR0YWNoZWQgaXMgdGhlIEJ1  ZGdldC7CoCBQbGVhc2U
gbGV0IG1lIGtub3cgaWYgeW91IGhhdmUgYW55IGNvbmN
lcm5zIG9yIGNv  bW1lbnRzIHJlZ2FyZGluZyBvdXIgc
HJvZ3Jlc3MgdGh1cyBmYXIuwqA8L2ZvbnQ+PGRpdiBzd
```

Figure 27 This is the output of the EML file, which directly correlates to *Figure 30*



```
D-21EF-45F6-ABA4-3DA3ABA2B36A_  Content-T
: application/octet-stream;  name="=?utf
8?Q?Forensicators,_Inc_(Team_2)_-_Budget.do
cx?="  Content-Description: =?utf-8?Q?Forens
icators,_Inc_(Team_2)_-_Budget.docx?=  Conte
nt-Disposition: attachment;  name="=?utf-8?
Q?Forensicators,_Inc_(Team_2)_-_Budget.docx?
="  Content-Transfer-Encoding: base64       -
-_57A3FF9D-21EF-45F6-ABA4-3DA3ABA2B36A_--
```

Figure 28 The name of any e-mail attachments will be located at the end of the file



```
<span style="border-collapse:collapse;font-family:arial,sans-serif;font-size:13px"><font color="
#222222">Dr [redacted]</font><br><br><font color="#222222">Attached is the Budget.Â  Please let me
 know if you have any concerns or comments regarding our progress thus far.Â </font><div style="
color:rgb(34,34,34);outline-style:none;outline-width:initial;outline-color:initial;padding-top:1
0px;padding-right:0px;padding-bottom:10px;padding-left:0px;width:22px">
<div style="background-color:rgb(241,241,241);border-top-width:1px;border-right-width:1px;border
-bottom-width:1px;border-left-width:1px;border-top-style:solid;border-right-style:solid;border-b
ottom-style:solid;border-left-style:solid;border-top-color:rgb(221,221,221);border-right-color:r
gb(221,221,221);border-bottom-color:rgb(221,221,221);border-left-color:rgb(221,221,221);clear:bo
th;line-height:6px;outline-style:none;outline-width:initial;outline-color:initial;width:20px">
<img src="https://mail.google.com/mail/images/cleardot.gif" style="background-image:url(&#39;htt
ps://ssl.gstatic.com/ui/v1/icons/mail/ellipsis.png&#39;);background-color:initial;min-height:8px
;width:20px;background-repeat:no-repeat no-repeat"></div><font color=#888888>
</font></div><font color=#888888><div style="color:rgb(34,34,34)"><br></div><span><font>Amanda T
homson<br>Project Manager<br>Forensicators, Inc.<br><i>Nybling the task down one byte at a time<
/i></font></span></font></span>
```

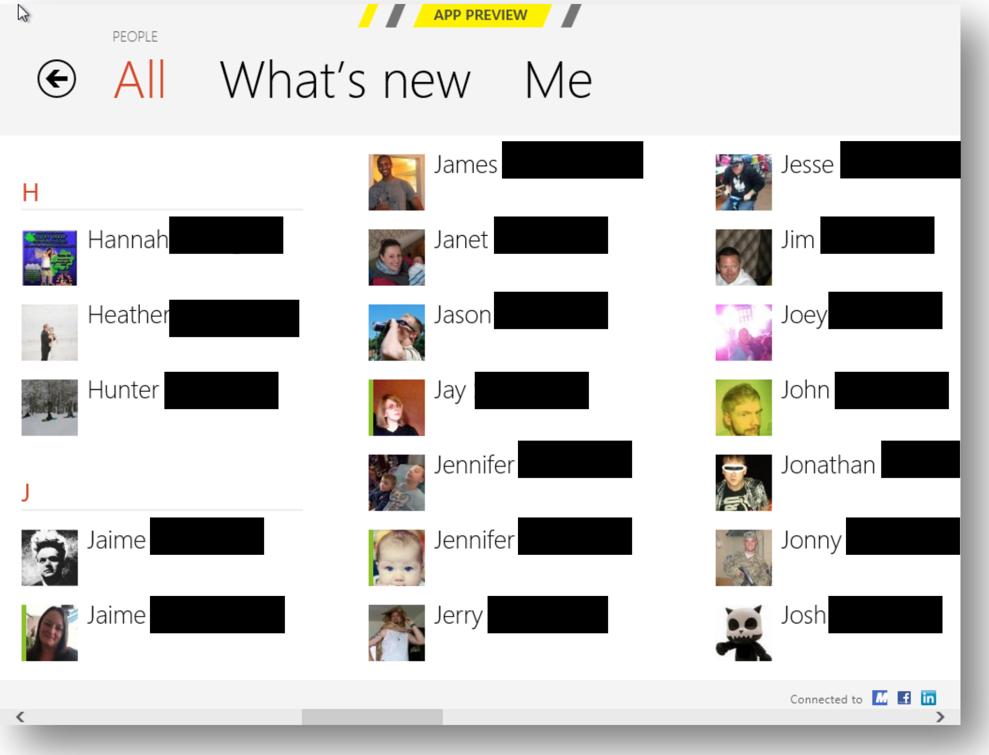
Figure 29 The contents of the e-mail decoded from Base64

User's Contacts

With Windows 8 Consumer Preview, I found that the user's contact information can be associated with an avatar represented by a picture or photo, if they have one.

What's so great about this? The Communications App consolidates social networking and messaging into one place, and as a result, the user's contacts are stored in one location, along with their contact's picture.

A Windows 8 user will see their contacts if they are logged in with a Windows Live account and their social networking and messaging accounts are linked similar to what's shown in *Figure 30*:



H

Hannah [redacted]

Heather [redacted]

Hunter [redacted]

J

Jaime [redacted]

Jaime [redacted]

James [redacted]

Janet [redacted]

Jason [redacted]

Jay [redacted]

Jennifer [redacted]

Jennifer [redacted]

Jerry [redacted]

Jesse [redacted]

Jim [redacted]

Joey [redacted]

John [redacted]

Jonathan [redacted]

Jonny [redacted]

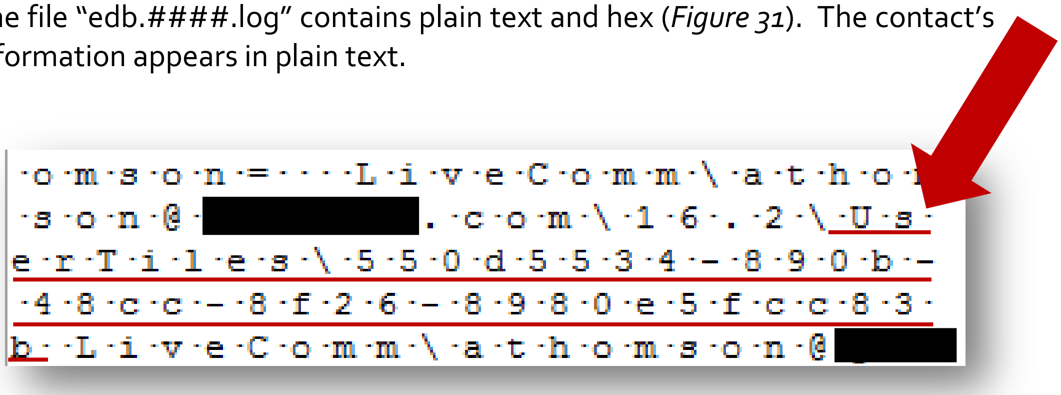
Josh [redacted]

Figure 30 How the user will see their contacts. PII has been omitted

Next, you probably want to attribute a contact with their avatar. Here's where you can go to do this:

Application	Location	Purpose
User's Contacts from Communications App	%Root%\Users\%User%\AppData\Local\Packages\microsoft.windowscommunicationsapps_8wekyb3d8bbwe\LocalState\LiveComm\%User's WindowsLiveEmail Address%\%App-CurrentVersion%\DBStore\LogFiles\edb####.log	Contains contact info, such as e-mail addresses, physical addresses, phone numbers, etc.

The file "edb.####.log" contains plain text and hex (*Figure 31*). The contact's information appears in plain text.



```
·o·m·s·o·n·=···L·i·v·e·C·o·m·m·\·a·t·h·o·
·s·o·n·@·[redacted].·c·o·m·\·1·6·.·2·\·U·s·
e·r·T·i·l·e·s·\·5·5·0·d·5·5·3·4·–·8·9·0·b·–
·4·8·c·c·–·8·f·2·6·–·8·9·8·0·e·5·f·c·c·8·3·
b··L·i·v·e·C·o·m·m·\·a·t·h·o·m·s·o·n·@[redacted]
```

Figure 31 Example of some of the contents in an "edb.####.log" file

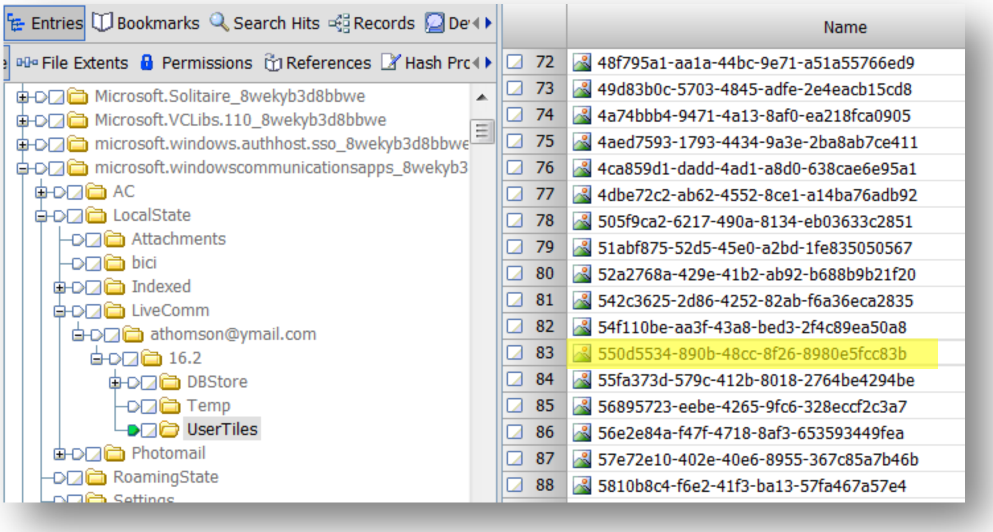
In *Figure 31*, this example shows that my e-mail account, athomson@xxxx.com, has a contact that is associated with a User Tile. A User Tile is the picture the contact uses as a profile picture on Facebook or their e-mail avatar. The User Tile tied to this contact is "550d5534-890b-48cc-8f26-8980e5fcc83b"[xvi].

Once you have this information, you can see what picture is being used for that contact at this next location.

Application	Location	Purpose
User Tile Associated with Contact	%Root%\Users\%User%\AppData\Local\Packages\microsoft.windowscommunicationsapps_8wekyb3d8bbwe\ LocalState\LiveComm\ %User'sWindowsLiveEmail Address%\%AppCurrentVersion%\DBStore\UserTiles	Contains Facebook picture or e-mail avatar of contact

Note that the picture you find here is one the contact associated with themselves. This is not a picture I associated with my contact.

Figure 32 shows what the forensic examiner will see when this location is viewed.

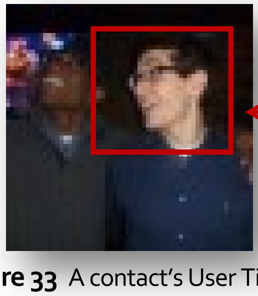


```
Microsoft.Solitaire_8wekyb3d8bbwe
Microsoft.VCLibs.110_8wekyb3d8bbwe
microsoft.windows.authhost.sso_8wekyb3d8bbwe
microsoft.windowscommunicationsapps_8wekyb3
    AC
    LocalState
        Attachments
        bici
        Indexed
        LiveComm
            athomson@ymail.com
                16.2
                    DBStore
                    Temp
                    UserTiles
        Photomail
    RoamingState
    Settings
```

	Name
72	48f795a1-aa1a-44bc-9e71-a51a55766ed9
73	49d83b0c-5703-4845-adfe-2e4eacb15cd8
74	4a74bbb4-9471-4a13-8af0-ea218fca0905
75	4aed7593-1793-4434-9a3e-2ba8ab7ce411
76	4ca859d1-dadd-4ad1-a8d0-638cae6e95a1
77	4dbe72c2-ab62-4552-8ce1-a14ba76adb92
78	505f9ca2-6217-490a-8134-eb03633c2851
79	51abf875-52d5-45e0-a2bd-1fe835050567
80	52a2768a-429e-41b2-ab92-b688b9b21f20
81	542c3625-2d86-4252-82ab-f6a36eca2835
82	54f110be-aa3f-43a8-bed3-2f4c89ea50a8
83	550d5534-890b-48cc-8f26-8980e5fcc83b
84	55fa373d-579c-412b-8018-2764be4294be
85	56895723-eebe-4265-9fc6-328eccf2c3a7
86	56e2e84a-f47f-4718-8af3-653593449fea
87	57e72e10-402e-40e6-8955-367c85a7b46b
88	5810b8c4-f6e2-41f3-ba13-57fa467a57e4

Figure 32 Example of contents in the "UserTiles" folder. The highlighted portion is the User Tile that was indicated in the contact's information in *Figure 31*.

The User Tile can then be viewed and a face can be put with the name of the contact (*Figure 33*)[xvii].



Contact associated with 550d5534-890b-48cc-8f26-8980e5fcc83b

Figure 33 A contact's User Tile

App Settings

The Communications App's settings are found in "settings.dat", which needs to be unpacked because it's a compound file. This file can be found at the following location:

Application	Location	Purpose
Communications App Settings	%Root%\Users\%User%\AppData\Local\Packages\microsoft.windowscommunicationsapps_8wekyb3d8bbwe\Settings	Contains settings for the Communications App

Once "settings.dat" is unpacked, you will see folders for the user's Windows Live account, their calendar, chat, e-mail, and people, among others. Much of the contents of the "Settings" folder are unknown at the time of this writing; however, it seems that the last 8 bytes of any of these entries contain the date and time, which is stored as Windows FILETIME[xviii].

Windows Registry

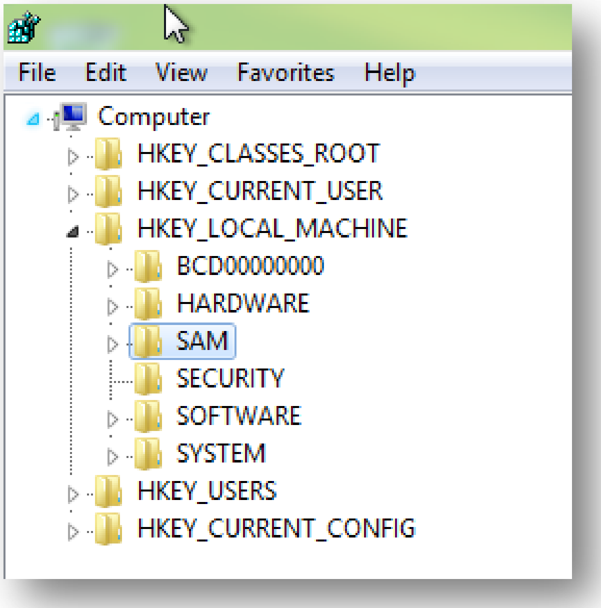
As forensic examiners, we should be familiar with the standard Windows Registry definition, which is that it is "[a] central hierarchical database used in Windows… [which is] used to store information that is necessary to configure the system for one or more users, applications, and hardware devices"[xix]*. As far as finding out what was really going on with the system and what the user was really doing, the going through the Registry is like winning the jackpot.*

Mining the Registry for forensically useful data is certainly a daunting task, and flipping through a couple hundred pages or trying to remember where a quick reference guide for a certain version of Windows was placed is inconvenient. In this section I will list forensically useful locations in the Windows Registry. Similar to the previous sections, unless otherwise noted, many of these locations are also compatible with Windows Vista and Windows 7. *Figure 34* shows there is no change to the Registry Structure within Windows 8.

ICON KEY

🐟 Windows 8

✳ More info



```
Computer
    HKEY_CLASSES_ROOT
    HKEY_CURRENT_USER
    HKEY_LOCAL_MACHINE
        BCD00000000
        HARDWARE
        SAM
        SECURITY
        SOFTWARE
        SYSTEM
    HKEY_USERS
    HKEY_CURRENT_CONFIG
```

Figure 34 The Windows 8 Registry as viewed from Regedit

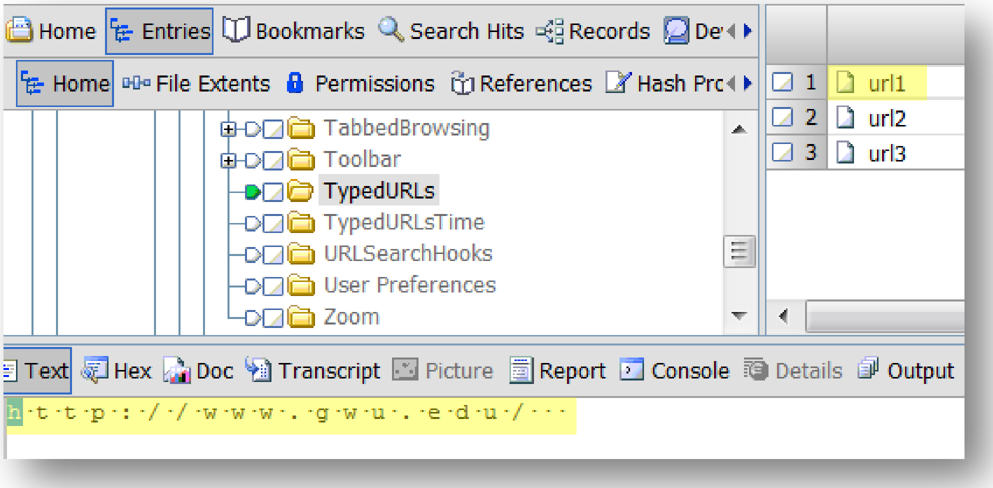
NTUSER.DAT

NTUSER.DAT stores information that is specific to the user. If there are multiple user accounts on the computer, there are also multiple NTUSER.DAT files – one for each user. NTUSER.DAT stores data that is specific to the user, such as which files they opened, which applications they used, and which websites they visited. All of this data can be found here:

%SystemRoot%\Users\%User%\NTUSER.DAT\Software\Microsoft\

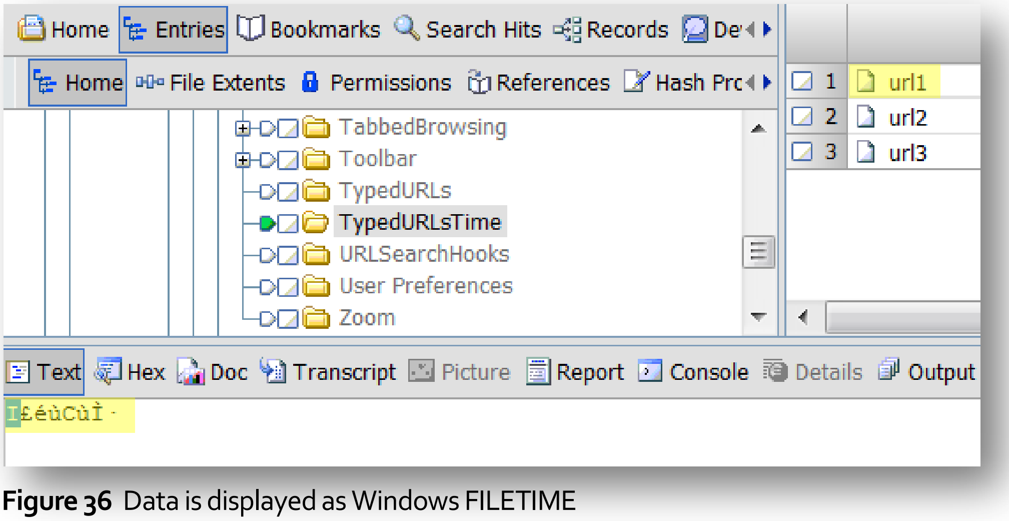
Data Stored	Registry Key Location
Recent Docs	Windows\CurrentVersion\Explorer\Recent Docs
Recently Opened/Saved Files	Windows\CurrentVersion\Explorer\ComDlg32\OpenSavePidlMRU
Recently Opened/Saved Folders	Windows\CurrentVersion\Explorer\ComDlg32\LastVisitedPidlMRU
Last Visited Folder	Windows\CurrentVersion\Explorer\ComDlg32\LastVisitedPidlMRULegacy
Recently Used Apps (Non-Metro Apps)	Windows\CurrentVersion\Explorer\ComDlg32\CIDSizeMRU
Recently Used Apps with Saved Files	Windows\CurrentVersion\Explorer\ComDlg32\FirstFolder
Recently Run Items	Windows\CurrentVersion\Explorer\Policies\RunMRU
Computer Name & Volume S/N	Windows Media\WMSDK\General
File Extension Associations	Windows\CurrentVersion\Explorer\FileExts
Typed URLs	Microsoft\Internet Explorer\TypedURLs
Typed URL Time (Figure 35, Figure 36, and Figure 37)	Microsoft\Internet Explorer\TypedURLsTime

The Typed URL Time is stored in binary and represents the number of 100-nanosecond intervals since January 1, 1601 at 00:00:00 GMT. The FILETIME structure consists of two 32-bit values that combine to form a single 64-bit value[xx]. The URLs found in TypedURLs (*Figure 35*) can be correlated to TypedURLsTime. The stored value, which is a FILETIME object, can give the time down to a fraction of a second from when the user typed that specific URL (Refer to *Figure 36* and *Figure 37*). More research needs to be conducted on this key as at the time of this writing, there is very little information on TypedURLsTime.



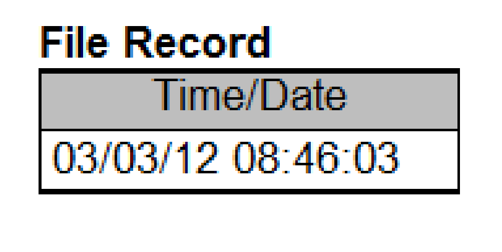
```
h·t·t·p·:·/·/·w·w·w·.·g·w·u·.·e·d·u·/···
```

Figure 35 The typed URL for "URL 1" in the Typed URLs key is http://www.gwu.edu



```
I£éùCùÌ ·
```

Figure 36 Data is displayed as Windows FILETIME



File Record

Time/Date
03/03/12 08:46:03

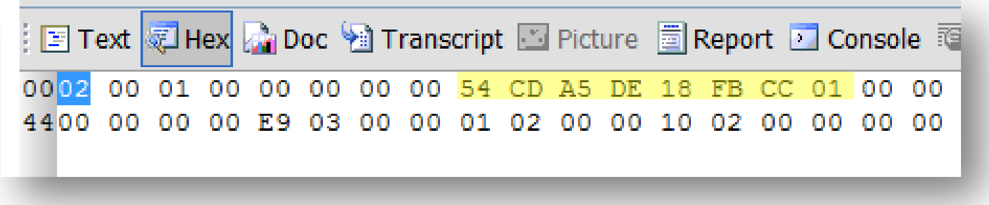
Figure 37 URL1 found in TypedURLsTime directly corresponds to URL1 found in TypedURLs

SAM

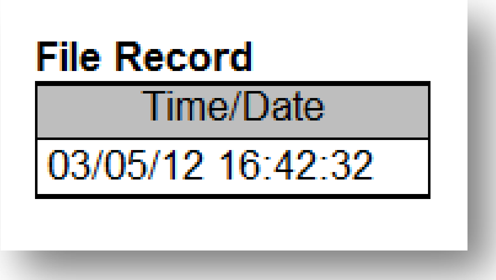
SAM (Security Accounts Manager) stores information that pertains to accounts, whether locally or on a domain. The SAM key stores user names that are used for login and the user's RID (Relative Identifier) for each account. Data stored in the SAM can be found here:

%SystemRoot%\Windows\System32\Config\SAM\Domains\Account\Users

Data Stored	Registry Key Location
Last Logon (Figure 38)	F
Last Password Change (Figure 39)	F
Account Expiration (Figure 40)	F
Last Failed Logon (Figure 41)	F
User's RID (Figure 42)	F
🐟 Internet User Name	InternetUserName (Windows Live Account)
User's First Name	GivenName
User's Last Name	Surname
🐟 User's Tile (Figure 43)	UserTile



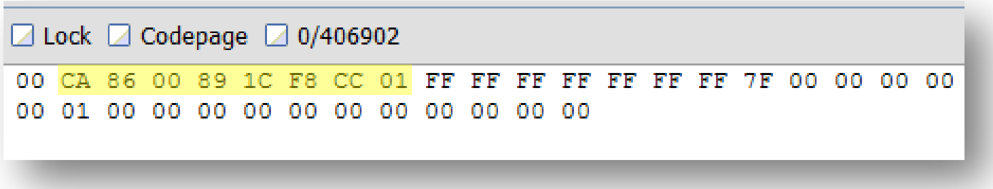
```
0002 00 01 00 00 00 00 00 54 CD A5 DE 18 FB CC 01 00 00
4400 00 00 00 E9 03 00 00 01 02 00 00 10 02 00 00 00 00
```



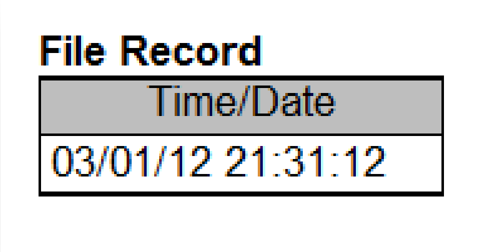
File Record

Time/Date
03/05/12 16:42:32

Figure 38 The user's last logon time is stored in bytes ox 8-15



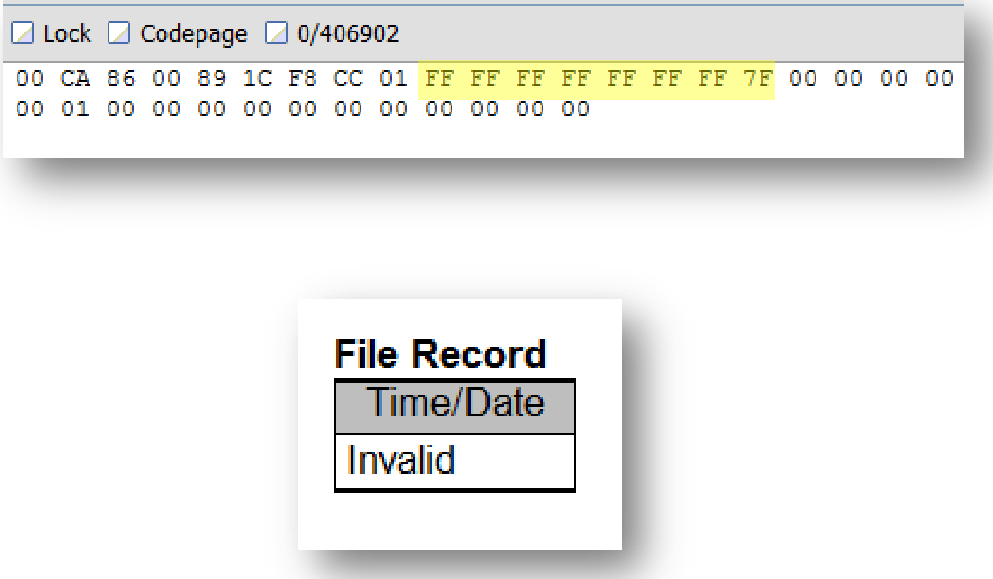
```
00 CA 86 00 89 1C F8 CC 01 FF FF FF FF FF FF FF 7F 00 00 00 00
00 01 00 00 00 00 00 00 00 00 00 00 00
```



File Record

Time/Date
03/01/12 21:31:12

Figure 39 The user's last password change is stored in bytes ox 24-31

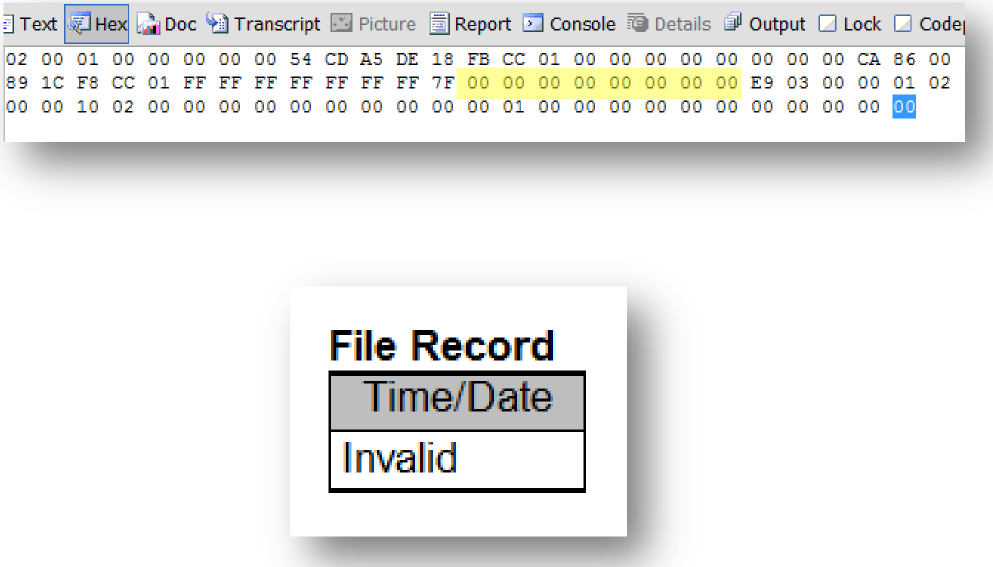


```
00 CA 86 00 89 1C F8 CC 01 FF FF FF FF FF FF FF 7F 00 00 00 00
00 01 00 00 00 00 00 00 00 00 00 00 00
```

File Record

Time/Date
Invalid

Figure 40 If the user's account was set to expire, a valid FILETIME would be here at 0x 32-39



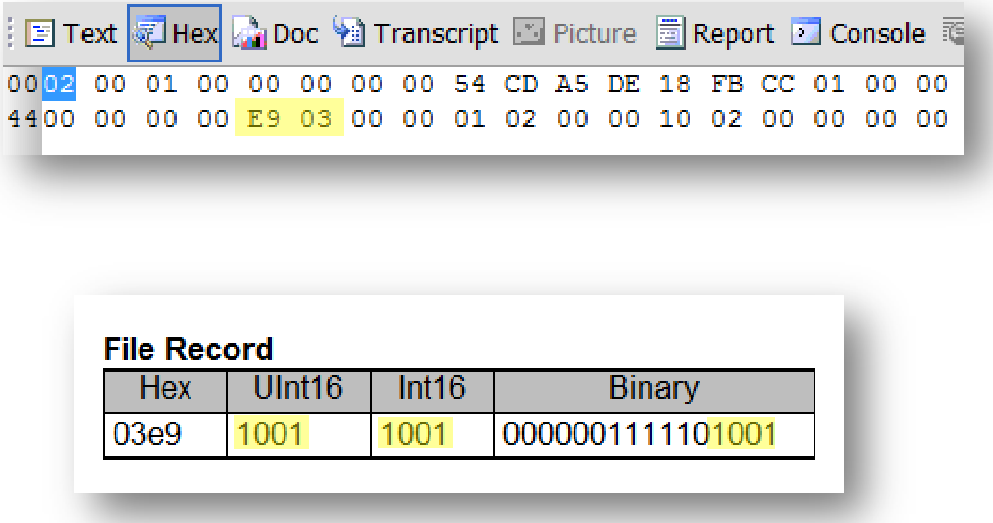
Text Hex Doc Transcript Picture Report Console Details Output ☐ Lock ☐ Code

```
02 00 01 00 00 00 00 00 54 CD A5 DE 18 FB CC 01 00 00 00 00 00 00 00 00 CA 86 00
89 1C F8 CC 01 FF FF FF FF FF FF FF 7F 00 00 00 00 00 00 00 00 E9 03 00 00 01 02
00 00 10 02 00 00 00 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00
```

File Record

Time/Date
Invalid

Figure 41 If the user had a failed logon, a valid FILETIME would be found at 0x 40-47

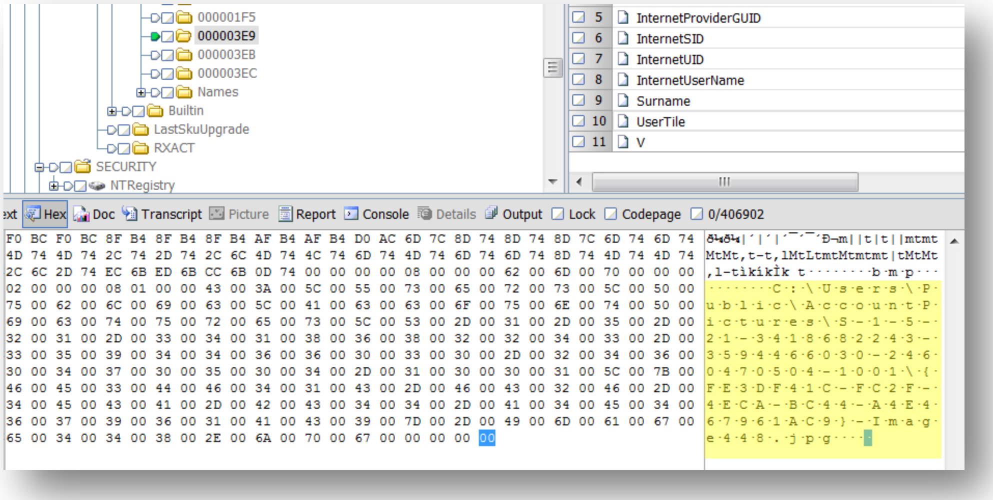


```
0002 00 01 00 00 00 00 00 54 CD A5 DE 18 FB CC 01 00 00
4400 00 00 00 E9 03 00 00 01 02 00 00 10 02 00 00 00 00
```

File Record

Hex	UInt16	Int16	Binary
03e9	1001	1001	0000001111101001

Figure 42 The user's relative identifier (RID), which is the last segment of the SID, is found at 0x 48-49



5	InternetProviderGUID
6	InternetSID
7	InternetUID
8	InternetUserName
9	Surname
10	UserTile
11	V

```
F0 BC F0 BC 8F B4 8F B4 8F B4 AF B4 AF B4 D0 AC 6D 7C 8D 74 8D 74 8D 7C 6D 74 6D 74  ð¼ð¼|´|´|¯´¯´Ð¬m||t|t||mtmt
4D 74 4D 74 2C 74 2D 74 2C 6C 4D 74 4C 74 6D 74 4D 74 6D 74 6D 74 8D 74 4D 74 4D 74  MtMt,t-t,lMtLtmtMtmtmt|tMtMt
2C 6C 2D 74 EC 6B ED 6B CC 6B 0D 74 00 00 00 00 08 00 00 00 62 00 6D 00 70 00 00 00  ,l-tìkíkÌk t········b·m·p···
02 00 00 00 08 01 00 00 43 00 3A 00 5C 00 55 00 73 00 65 00 72 00 73 00 5C 00 50 00  ········C·:·\·U·s·e·r·s·\·P·
75 00 62 00 6C 00 69 00 63 00 5C 00 41 00 63 00 63 00 6F 00 75 00 6E 00 74 00 50 00  u·b·l·i·c·\·A·c·c·o·u·n·t·P·
69 00 63 00 74 00 75 00 72 00 65 00 73 00 5C 00 53 00 2D 00 31 00 2D 00 35 00 2D 00  i·c·t·u·r·e·s·\·S·-·1·-·5·-·
32 00 31 00 2D 00 33 00 34 00 31 00 38 00 36 00 38 00 32 00 32 00 34 00 33 00 2D 00  2·1·-·3·4·1·8·6·8·2·2·4·3·-·
33 00 35 00 39 00 34 00 34 00 36 00 36 00 30 00 33 00 30 00 2D 00 32 00 34 00 36 00  3·5·9·4·4·6·6·0·3·0·-·2·4·6·
30 00 34 00 37 00 30 00 35 00 30 00 34 00 2D 00 31 00 30 00 30 00 31 00 5C 00 7B 00  0·4·7·0·5·0·4·-·1·0·0·1·\·{·
46 00 45 00 33 00 44 00 46 00 34 00 31 00 43 00 2D 00 46 00 43 00 32 00 46 00 2D 00  F·E·3·D·F·4·1·C·-·F·C·2·F·-·
34 00 45 00 43 00 41 00 2D 00 42 00 43 00 34 00 34 00 2D 00 41 00 34 00 45 00 34 00  4·E·C·A·-·B·C·4·4·-·A·4·E·4·
36 00 37 00 39 00 36 00 31 00 41 00 43 00 39 00 7D 00 2D 00 49 00 6D 00 61 00 67 00  6·7·9·6·1·A·C·9·}·-·I·m·a·g·
65 00 34 00 34 00 38 00 2E 00 6A 00 70 00 67 00 00 00 00                             e·4·4·8·.·j·p·g····
```

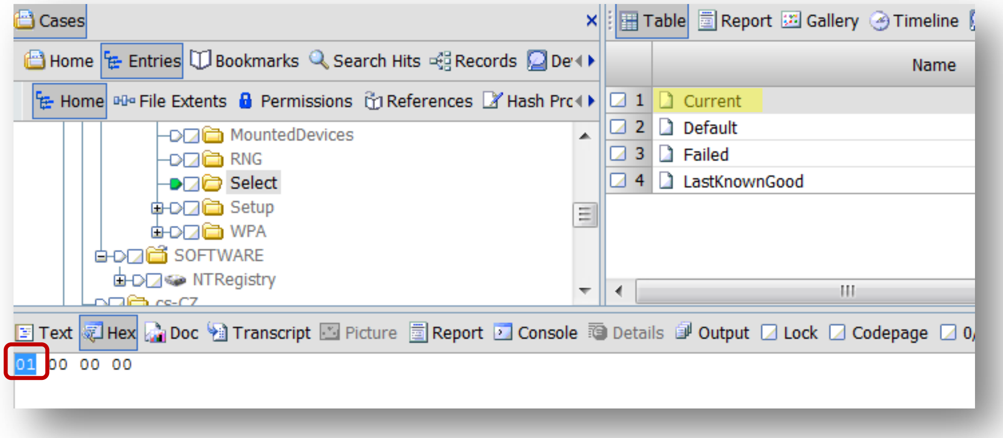
Figure 43 The file used for the user's tile can be found at the end of the UserTile key

SYSTEM

The SYSTEM key contains information about the operating system, such as which devices were assigned a drive letter, the name of the computer, time zone setting, and USB devices attached to the system. It also keeps track of control sets, which is "a collection of configuration data needed to control system boot"[xxi].

%SystemRoot%\Windows\System32\config\SYSTEM\

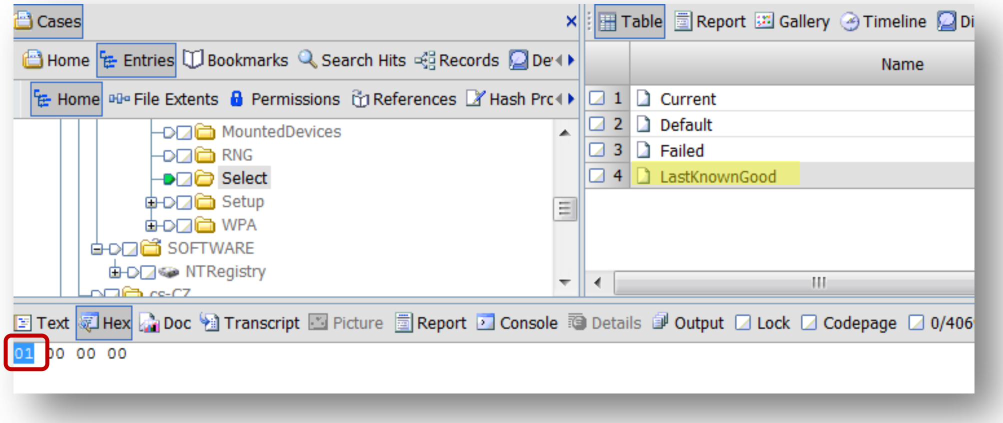
Data Stored	Registry Key Location
Current Control Set (Figure 24)	Select\Current
Last Known Good Control Set (Figure 25)	Select\LastKnownGood
Mounted Devices (Figures 26-28)	MountedDevices
Files Excluded from Restore	%CurrentControlSet%\Control\BackupRestore
Computer Name	%CurrentControlSet%\Control\ComputerName
Time Zone	%CurrentControlSet%\Control\TimeZoneInformation\TimeZoneKeyName
Last Graceful Shutdown Time (Figure 29)	%CurrentControlSet%\Control\Windows\ShutdownTime (Data stored in Windows FILETIME)
Printers	%CurrentControlSet%\Enum\SWD\PRINTENUM\FriendlyName
Sensors & Location Devices	%CurrentControlSet%\Enum\SWD\SensorsAndLocationEnum\HardwareID
USB Storage Devices	%CurrentControlSet%\Enum\USBSTOR



	Name
1	Current
2	Default
3	Failed
4	LastKnownGood

```
01 00 00 00
```

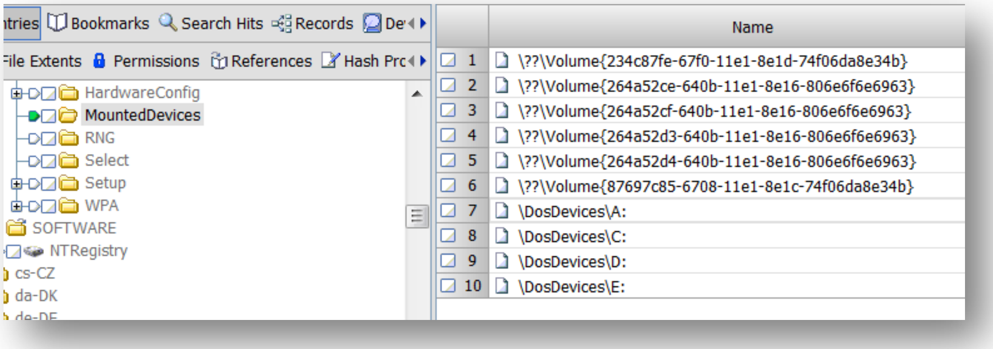
Figure 44 The Current Control Set is "01". Whichever Control Set is current, that is where a majority of the system's information will come from. Of course, it never hurts to check the other control sets.



	Name
1	Current
2	Default
3	Failed
4	LastKnownGood

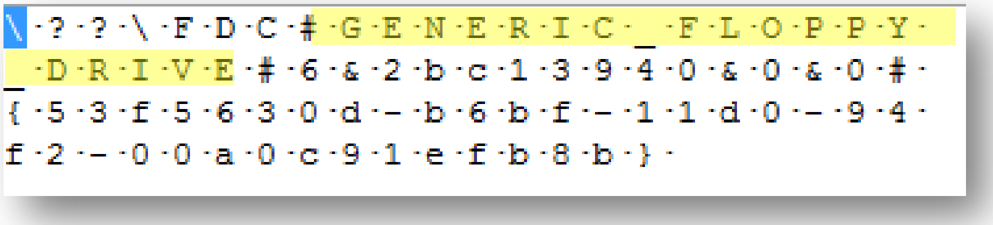
```
01 00 00 00
```

Figure 45 The Last Known Good Control Set is "01". This is the control set that was used during the last successful boot.



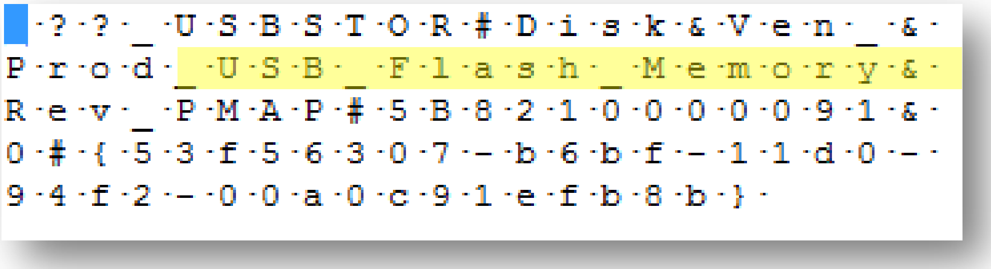
	Name
1	\??\Volume{234c87fe-67f0-11e1-8e1d-74f06da8e34b}
2	\??\Volume{264a52ce-640b-11e1-8e16-806e6f6e6963}
3	\??\Volume{264a52cf-640b-11e1-8e16-806e6f6e6963}
4	\??\Volume{264a52d3-640b-11e1-8e16-806e6f6e6963}
5	\??\Volume{264a52d4-640b-11e1-8e16-806e6f6e6963}
6	\??\Volume{87697c85-6708-11e1-8e1c-74f06da8e34b}
7	\DosDevices\A:
8	\DosDevices\C:
9	\DosDevices\D:
10	\DosDevices\E:

Figure 46 Four devices were assigned a drive letter. Note that the devices assigned a drive letter are the most recent device to have that drive letter.



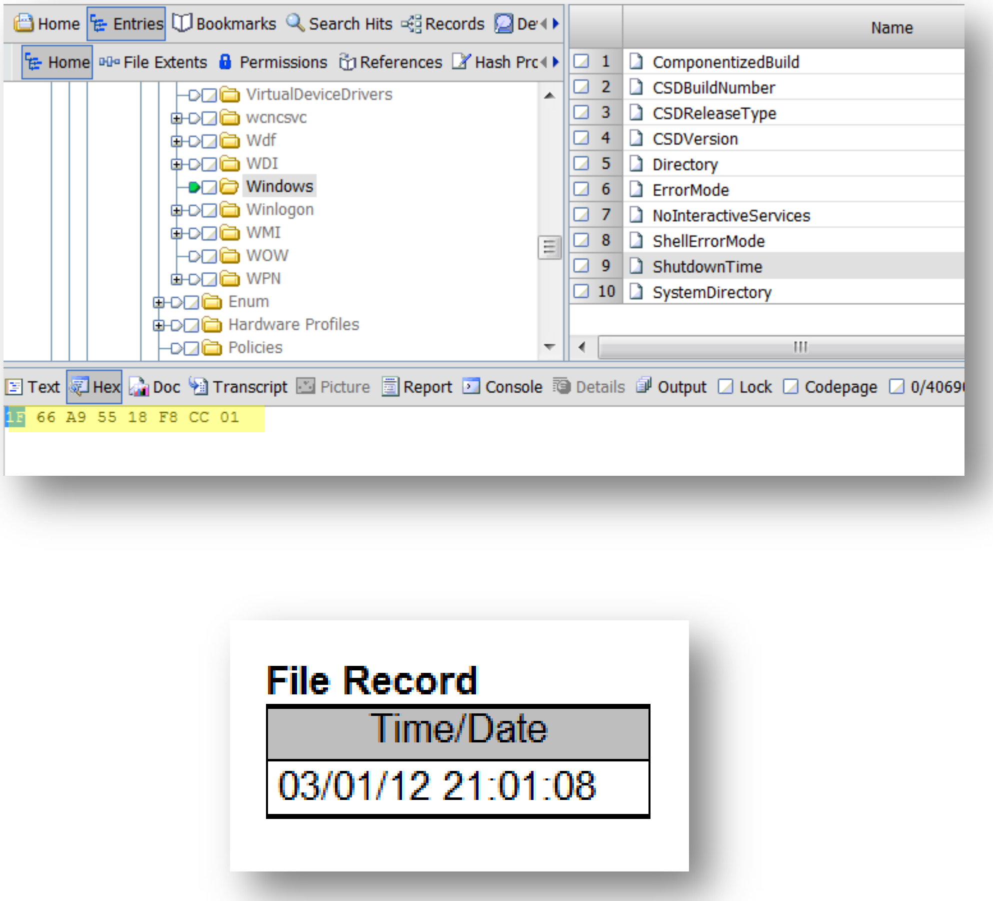
```
\·?·?·\·F·D·C·#·G·E·N·E·R·I·C·_·F·L·O·P·P·Y·
_·D·R·I·V·E·#·6·&·2·b·c·1·3·9·4·0·&·0·&·0·#·
{·5·3·f·5·6·3·0·d·-·b·6·b·f·-·1·1·d·0·-·9·4·
f·2·-·0·0·a·0·c·9·1·e·f·b·8·b·}·
```

Figure 47 "A" was assigned to "Generic Floppy Drive"



```
·?·?·_·U·S·B·S·T·O·R·#·D·i·s·k·&·V·e·n·_·&·
P·r·o·d·_·U·S·B·_·F·l·a·s·h·_·M·e·m·o·r·y·&·
R·e·v·_·P·M·A·P·#·5·B·8·2·1·0·0·0·0·0·9·1·&·
0·#·{·5·3·f·5·6·3·0·7·-·b·6·b·f·-·1·1·d·0·-·
9·4·f·2·-·0·0·a·0·c·9·1·e·f·b·8·b·}·
```

Figure 48 "E" was assigned to "USB Flash Memory"



	Name
1	ComponentizedBuild
2	CSDBuildNumber
3	CSDReleaseType
4	CSDVersion
5	Directory
6	ErrorMode
7	NoInteractiveServices
8	ShellErrorMode
9	ShutdownTime
10	SystemDirectory

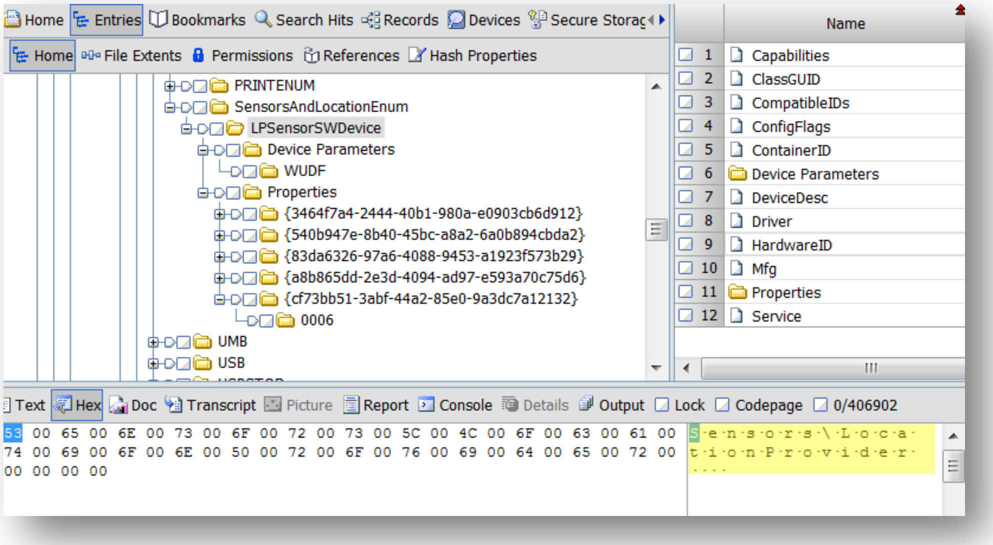
```
1F 66 A9 55 18 F8 CC 01
```

File Record

Time/Date
03/01/12 21:01:08

Figure 49 The last graceful shutdown time

Sensor and Location Devices is a new feature implemented with Windows 7. Enabling sensors allows users to have a more personalized experience with the OS and Internet-based activities, to include GPS information[xxii]. *Figure 50* shows that a Location Sensor was enabled on Windows 8. More research needs to be conducted as there is possibly a yet-to-be-found log file that corresponds to the sensor and other information relating to the device (if that is the type of sensor used)[xxiii].



```
PRINTENUM
SensorsAndLocationEnum
  LPSensorSWDevice
    Device Parameters
      WUDF
    Properties
      {3464f7a4-2444-40b1-980a-e0903cb6d912}
      {540b947e-8b40-45bc-a8a2-6a0b894cbda2}
      {83da6326-97a6-4088-9453-a1923f573b29}
      {a8b865dd-2e3d-4094-ad97-e593a70c75d6}
      {cf73bb51-3abf-44a2-85e0-9a3dc7a12132}
        0006
UMB
USB
```

	Name
1	Capabilities
2	ClassGUID
3	CompatibleIDs
4	ConfigFlags
5	ContainerID
6	Device Parameters
7	DeviceDesc
8	Driver
9	HardwareID
10	Mfg
11	Properties
12	Service

```
53 00 65 00 6E 00 73 00 6F 00 72 00 73 00 5C 00 4C 00 6F 00 63 00 61 00   S·e·n·s·o·r·s·\·L·o·c·a·
74 00 69 00 6F 00 6E 00 50 00 72 00 6F 00 76 00 69 00 64 00 65 00 72 00   t·i·o·n·P·r·o·v·i·d·e·r·
00 00 00 00                                                               ····
```

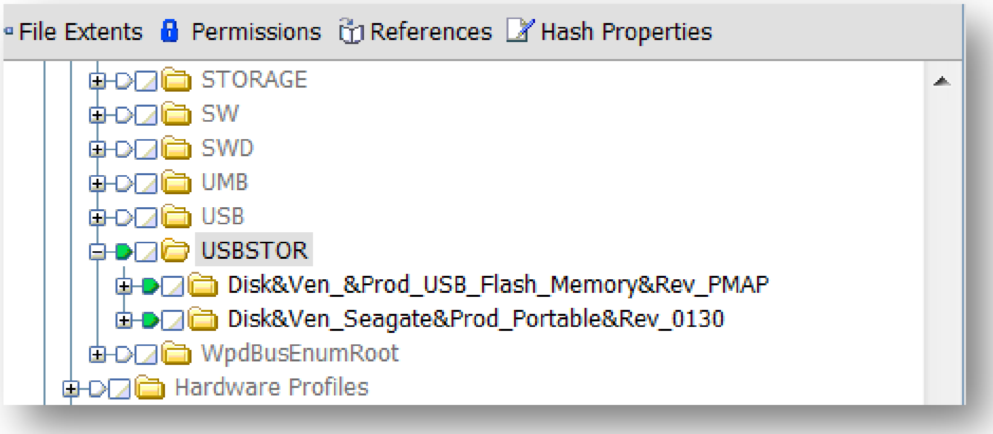
Figure 50 The type of Sensor and Location device used on this system is a Location Provider

USB STORAGE DEVICES

USB storage devices that were attached to the system can be uniquely identified in the System key by checking a few other keys. They can also be attributed to which port and hub they were plugged into, the date and time stamp, and a drive letter. If USB storage devices were not attached to the system, the USBSTOR key will not be present; however, you should also check link files, restore points, shadow copies and "setupapi.dev.log", as these may contain evidence of a USB device having been present on the system. Under USBSTOR (*Figure 51*), the Registry stores the USB device's friendly name (*Figure 52)*, and it also stores the device's vendor ID, product ID, revision number, and serial number (*Figure 53)*. If the device does not have a serial number, Windows will create a Unique Instance ID. If the second character of the serial number is "&", then it is not a serial number, but rather a Unique Instance ID, which will look similar to the following:

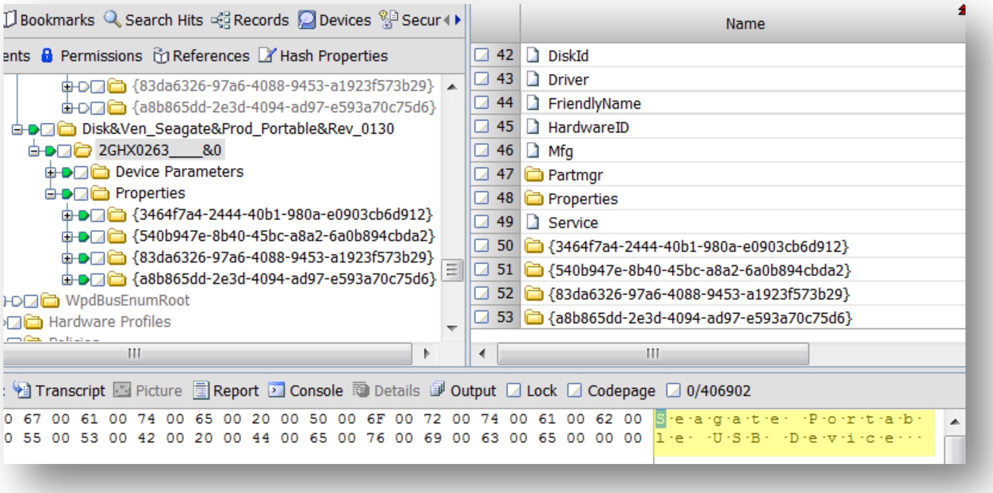
0&26D88A54&0[xxiv]

Either way, it is referred to as a Unique Instance ID.



```
STORAGE
SW
SWD
UMB
USB
USBSTOR
    Disk&Ven_&Prod_USB_Flash_Memory&Rev_PMAP
    Disk&Ven_Seagate&Prod_Portable&Rev_0130
WpdBusEnumRoot
Hardware Profiles
```

Figure 51 The USBSTOR key

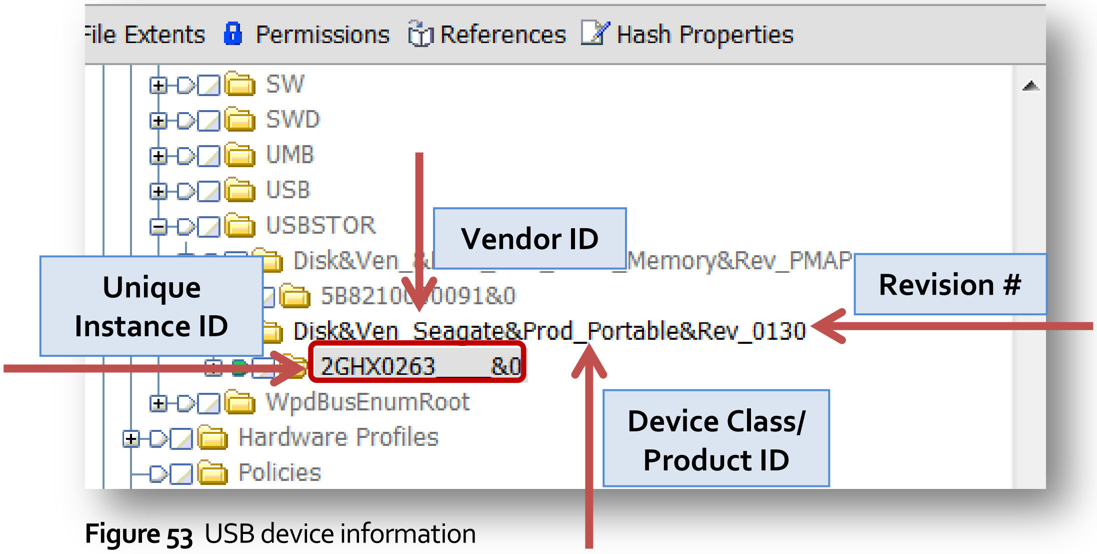


```
{83da6326-97a6-4088-9453-a1923f573b29}
{a8b865dd-2e3d-4094-ad97-e593a70c75d6}
Disk&Ven_Seagate&Prod_Portable&Rev_0130
  2GHX0263____&0
    Device Parameters
    Properties
      {3464f7a4-2444-40b1-980a-e0903cb6d912}
      {540b947e-8b40-45bc-a8a2-6a0b894cbda2}
      {83da6326-97a6-4088-9453-a1923f573b29}
      {a8b865dd-2e3d-4094-ad97-e593a70c75d6}
WpdBusEnumRoot
Hardware Profiles
```

	Name
42	DiskId
43	Driver
44	FriendlyName
45	HardwareID
46	Mfg
47	Partmgr
48	Properties
49	Service
50	{3464f7a4-2444-40b1-980a-e0903cb6d912}
51	{540b947e-8b40-45bc-a8a2-6a0b894cbda2}
52	{83da6326-97a6-4088-9453-a1923f573b29}
53	{a8b865dd-2e3d-4094-ad97-e593a70c75d6}

```
0 67 00 61 00 74 00 65 00 20 00 50 00 6F 00 72 00 74 00 61 00 62 00  S·e·a·g·a·t·e· ·P·o·r·t·a·b·
0 55 00 53 00 42 00 20 00 44 00 65 00 76 00 69 00 63 00 65 00 00 00  l·e· ·U·S·B· ·D·e·v·i·c·e···
```

Figure 52 USB Device Friendly Name



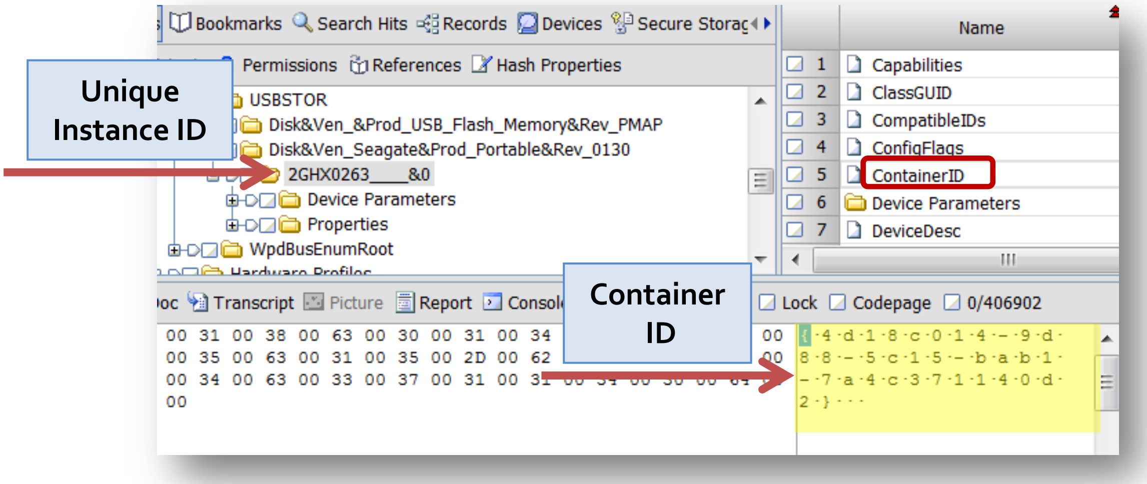
```
SW
SWD
UMB
USB
USBSTOR
  Disk&Ven_&I_ _ _Memory&Rev_PMAP
    5B8210 0091&0
  Disk&Ven_Seagate&Prod_Portable&Rev_0130
    2GHX0263    &0
WpdBusEnumRoot
Hardware Profiles
Policies
```

Figure 53 USB device information

In Mounted Devices (*Figure 46*), there was a drive letter for one USB device, but in the USBSTOR (*Figure 51*), there are two USB storage devices noted. So, why is only one of the USB storage devices assigned a drive letter? We know that the most recent device plugged into a system is assigned a drive letter, but that still doesn't tell us why only one device has a drive letter. Did the user try to cover their tracks? What's going on in the Registry when a USB storage device is plugged in? Is it even possible to figure out if a drive letter was assigned to that other USB storage device?

Of course we can figure this out! It just takes a bit of digging and note taking. So let's get started.

Go to the Unique Instance ID for the Seagate Portable USB Device (derived from the Friendly Name), which was found in *Figure 52*. Make a note of the Unique Instance ID because we will need to refer to it a few times throughout this process. Under the Unique Instance ID find the Container ID and note that value because we will need to refer to it a few times throughout this process.



USBSTOR
Disk&Ven_&Prod_USB_Flash_Memory&Rev_PMAP
Disk&Ven_Seagate&Prod_Portable&Rev_0130
2GHX0263____&0
Device Parameters
Properties
WpdBusEnumRoot
Hardware Profiles

	Name
1	Capabilities
2	ClassGUID
3	CompatibleIDs
4	ConfigFlags
5	ContainerID
6	Device Parameters
7	DeviceDesc

```
00 31 00 38 00 63 00 30 00 31 00 34              00   {·4·d·1·8·c·0·1·4·–·9·d·
00 35 00 63 00 31 00 35 00 2D 00 62              00   8·8·–·5·c·1·5·–·b·a·b·1·
00 34 00 63 00 33 00 37 00 31 00 31                   –·7·a·4·c·3·7·1·1·4·0·d·
00                                                    2·}···
```

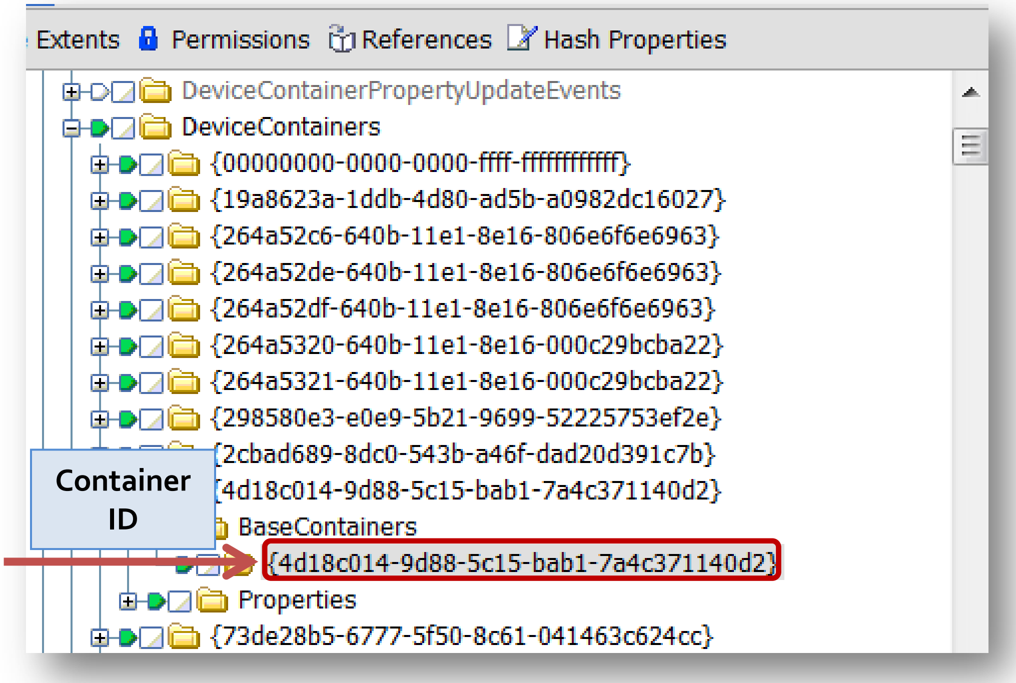
Figure 54 Container ID for Seagate Portable USB Device

What exactly is a Container ID? Beginning with Windows 7, the operating system uses Container IDs for each instance that a physical device installed on the system[xxv].

"A system-supplied device identification string that uniquely groups the functional devices associated with a single-function or multi-function device… Starting with Windows 7, the Plug-n-Play (PNP) manager uses the Container ID to group one or more device nodes (devnodes) that originated from and belong to each instance of a particular physical device. This instance is referred to as the device container[xxvi]."

In *Figure 54*, we found that the Container ID is {4d18c014-9d88-5c15-bab1-7a4c371140d2}. Remember to make note of this value. Next, go to the following Registry location and search for the Container ID again:

Device Containers	%CurrentControlSet%\Control\DeviceContainers\ %ContainerID%\BaseContainers **AND** %CurrentControlSet%\Control\DeviceContainers\ %ContainerID%\Properties

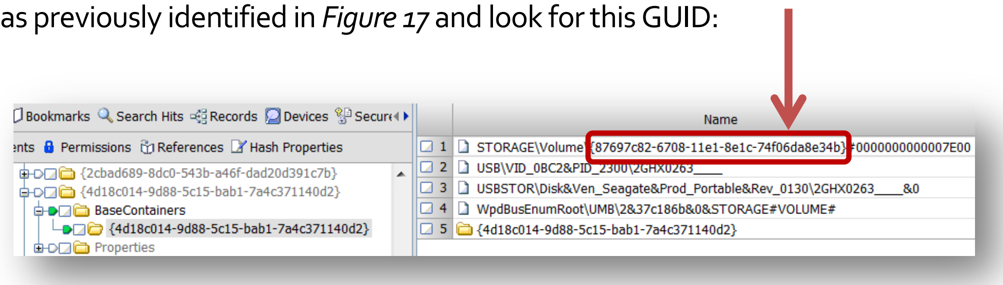


```
DeviceContainerPropertyUpdateEvents
DeviceContainers
  {00000000-0000-0000-ffff-ffffffffffff}
  {19a8623a-1ddb-4d80-ad5b-a0982dc16027}
  {264a52c6-640b-11e1-8e16-806e6f6e6963}
  {264a52de-640b-11e1-8e16-806e6f6e6963}
  {264a52df-640b-11e1-8e16-806e6f6e6963}
  {264a5320-640b-11e1-8e16-000c29bcba22}
  {264a5321-640b-11e1-8e16-000c29bcba22}
  {298580e3-e0e9-5b21-9699-52225753ef2e}
  {2cbad689-8dc0-543b-a46f-dad20d391c7b}
  {4d18c014-9d88-5c15-bab1-7a4c371140d2}
    BaseContainers
      {4d18c014-9d88-5c15-bab1-7a4c371140d2}
    Properties
  {73de28b5-6777-5f50-8c61-041463c624cc}
```

Container ID

Figure 55 Device Containers

From Base Containers (shown above in *Figure 18)*, find the same Container ID that was previously identified in *Figure 17* and look for this GUID:

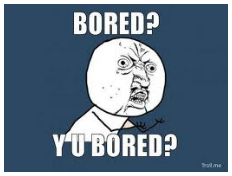


```
{2cbad689-8dc0-543b-a46f-dad20d391c7b}
{4d18c014-9d88-5c15-bab1-7a4c371140d2}
  BaseContainers
    {4d18c014-9d88-5c15-bab1-7a4c371140d2}
  Properties
```

	Name
1	STORAGE\Volume\{87697c82-6708-11e1-8e1c-74f06da8e34b}#0000000000007E00
2	USB\VID_0BC2&PID_2300\2GHX0263____
3	USBSTOR\Disk&Ven_Seagate&Prod_Portable&Rev_0130\2GHX0263____&0
4	WpdBusEnumRoot\UMB\2&37c186b&0&STORAGE#VOLUME#
5	{4d18c014-9d88-5c15-bab1-7a4c371140d2}

Figure 56 Properties under the Container ID in Device Containers

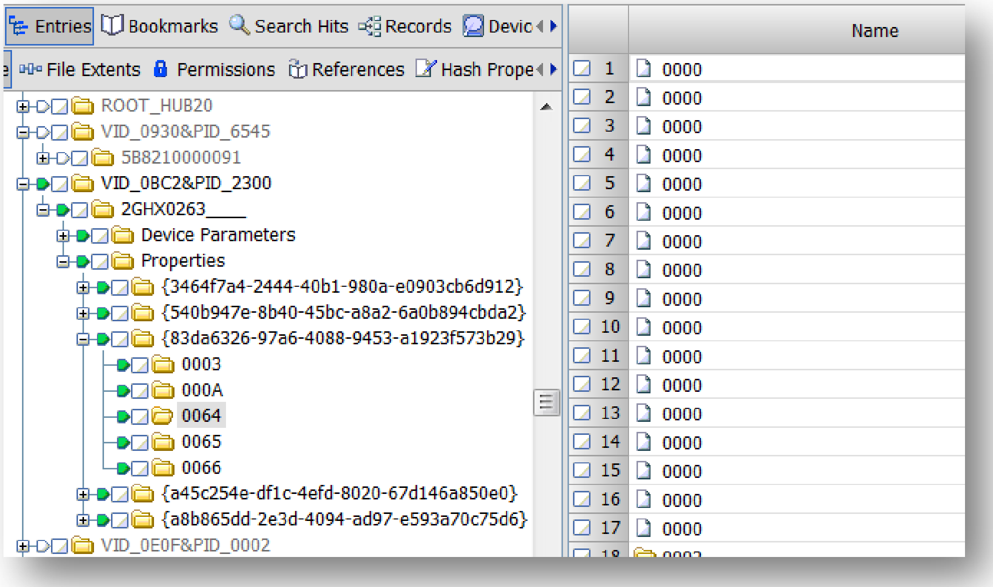
Take note of the GUID found here, which may be different than {87697c82-6708-11e1-8e1c-74f06da8e34b} identified in *Figure 56*, as this will be needed later on when we go back to Mounted Devices.



In order to figure out why the Seagate Portable USB Device doesn't have a drive letter, there are a few more steps (don't worry; there is a point to this).

In order to help us better understand what's going on with these USB devices, it might benefit us to figure out the date and time the Seagate Portable USB Device was plugged into the computer. Check this location:

USB Date & Time	%CurrentControlSet%\Enum\USB\%USBDevice%\%UniqueInstanceID%\Properties\{83da6326-97a6-4088-9453-a1923f573b29}

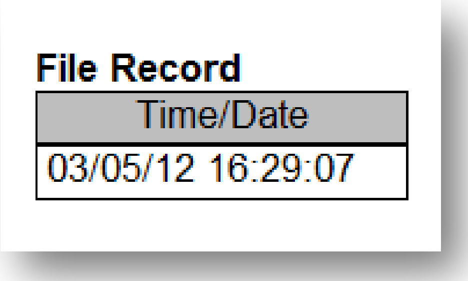


```
ROOT_HUB20
VID_0930&PID_6545
    5B8210000091
VID_0BC2&PID_2300
    2GHX0263____
        Device Parameters
        Properties
            {3464f7a4-2444-40b1-980a-e0903cb6d912}
            {540b947e-8b40-45bc-a8a2-6a0b894cbda2}
            {83da6326-97a6-4088-9453-a1923f573b29}
                0003
                000A
                0064
                0065
                0066
            {a45c254e-df1c-4efd-8020-67d146a850e0}
            {a8b865dd-2e3d-4094-ad97-e593a70c75d6}
VID_0E0F&PID_0002
```

	Name
1	0000
2	0000
3	0000
4	0000
5	0000
6	0000
7	0000
8	0000
9	0000
10	0000
11	0000
12	0000
13	0000
14	0000
15	0000
16	0000
17	0000

Figure 57 Location for date and time stamp (Windows FILETIME)

The third entry in *Figure 57* contained the Windows FILETIME the Seagate Portable USB Device was plugged into the computer (*Figure 58*).



File Record

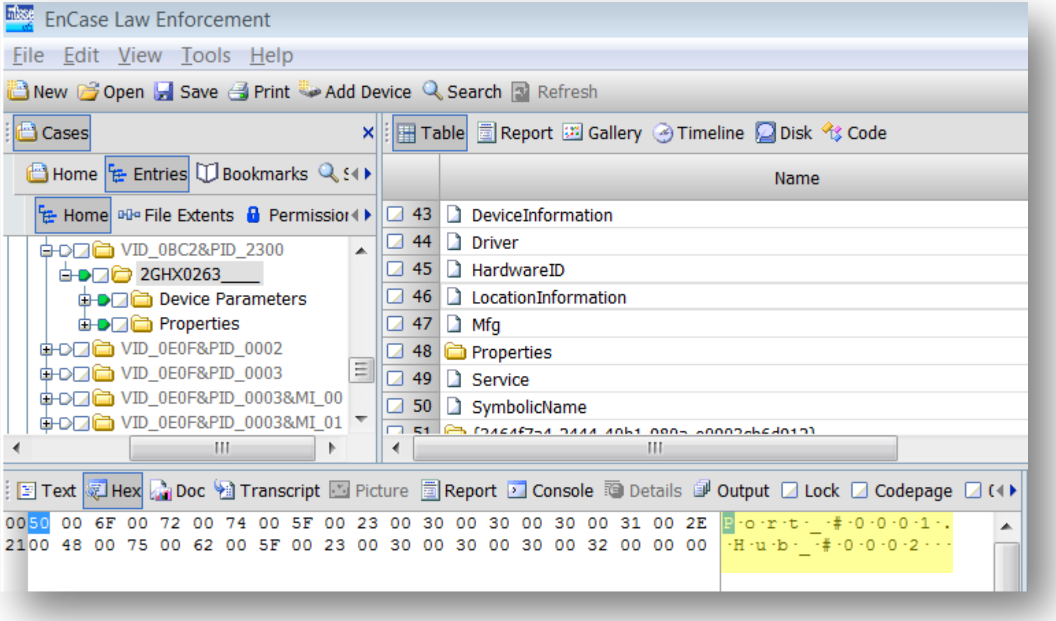
Time/Date
03/05/12 16:29:07

Figure 58 Seagate Portable USB Device's date and time stamp

Next, navigate to the following Registry location so you can figure out which port the USB device was plugged into:

USB Port	%CurrentControlSet%\Enum\USB\LocationInformation

Look for the USB device you've been working with. This should be easy if you noted the device's Unique Instance Identifier.


EnCase Law Enforcement
File Edit View Tools Help
New Open Save Print Add Device Search Refresh
Cases
Home Entries Bookmarks
Home File Extents Permission
Table Report Gallery Timeline Disk Code
Text Hex Doc Transcript Picture Report Console Details Output Lock Codepage

- VID_0BC2&PID_2300
  - 2GHX0263____
    - Device Parameters
    - Properties
- VID_0E0F&PID_0002
- VID_0E0F&PID_0003
- VID_0E0F&PID_0003&MI_00
- VID_0E0F&PID_0003&MI_01

	Name
43	DeviceInformation
44	Driver
45	HardwareID
46	LocationInformation
47	Mfg
48	Properties
49	Service
50	SymbolicName

```
0050 00 6F 00 72 00 74 00 5F 00 23 00 30 00 30 00 31 00 2E  P·o·r·t·_·#·0·0·0·1·.
2100 48 00 75 00 62 00 5F 00 23 00 30 00 30 00 30 00 32 00 00 00  ·H·u·b·_·#·0·0·0·2···
```

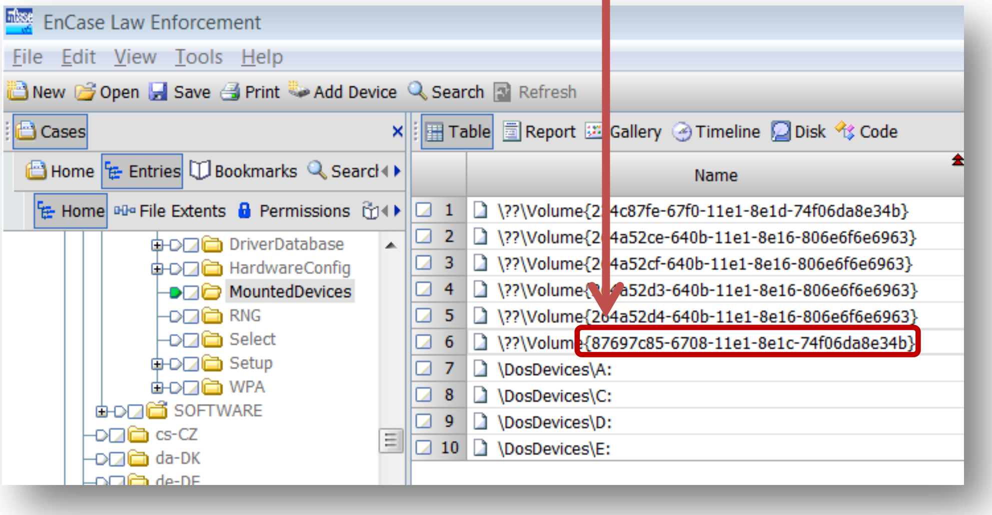
Figure 59 Port and hub device was plugged into

In *Figure 59*, this USB device was plugged into Port 1 on Hub 2. Why should you care? Knowing the port number could help us figure out why there may not be a drive letter associated with the USB device and it could also help build a timeline of the user's activities.

Now go back to Mounted Devices.

Mounted Devices	Mounted Devices

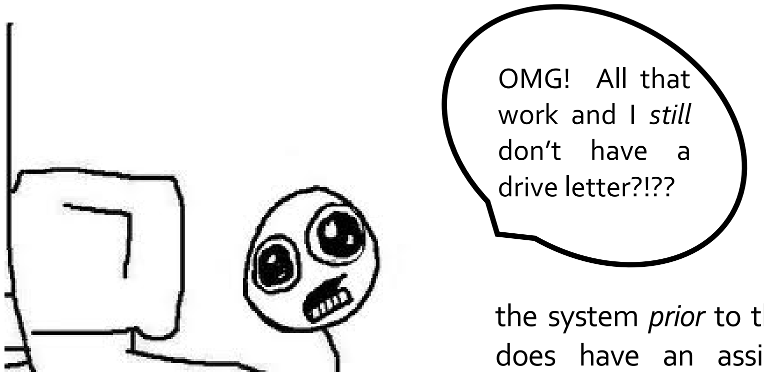
The GUID that should have been noted from *Figure 56* comes into play here ({87697c82-6708-11e1-8e1c-74f06da8e34b}). Under Mounted Devices, look for that GUID.


EnCase Law Enforcement
File Edit View Tools Help
New Open Save Print Add Device Search Refresh
Cases
Home Entries Bookmarks Searcl
Home File Extents Permissions
DriverDatabase
HardwareConfig
MountedDevices
RNG
Select
Setup
WPA
SOFTWARE
cs-CZ
da-DK
de-DE
Table Report Gallery Timeline Disk Code

	Name
1	\??\Volume{2d4c87fe-67f0-11e1-8e1d-74f06da8e34b}
2	\??\Volume{2d4a52ce-640b-11e1-8e16-806e6f6e6963}
3	\??\Volume{2d4a52cf-640b-11e1-8e16-806e6f6e6963}
4	\??\Volume{2d4a52d3-640b-11e1-8e16-806e6f6e6963}
5	\??\Volume{2d4a52d4-640b-11e1-8e16-806e6f6e6963}
6	\??\Volume{87697c85-6708-11e1-8e1c-74f06da8e34b}
7	\DosDevices\A:
8	\DosDevices\C:
9	\DosDevices\D:
10	\DosDevices\E:

Figure 60 Mounted devices

Sadly, they do not match, as the GUID noted here is {87697c8**5**-6708-11e1-8e1c-74foda8e34b}.

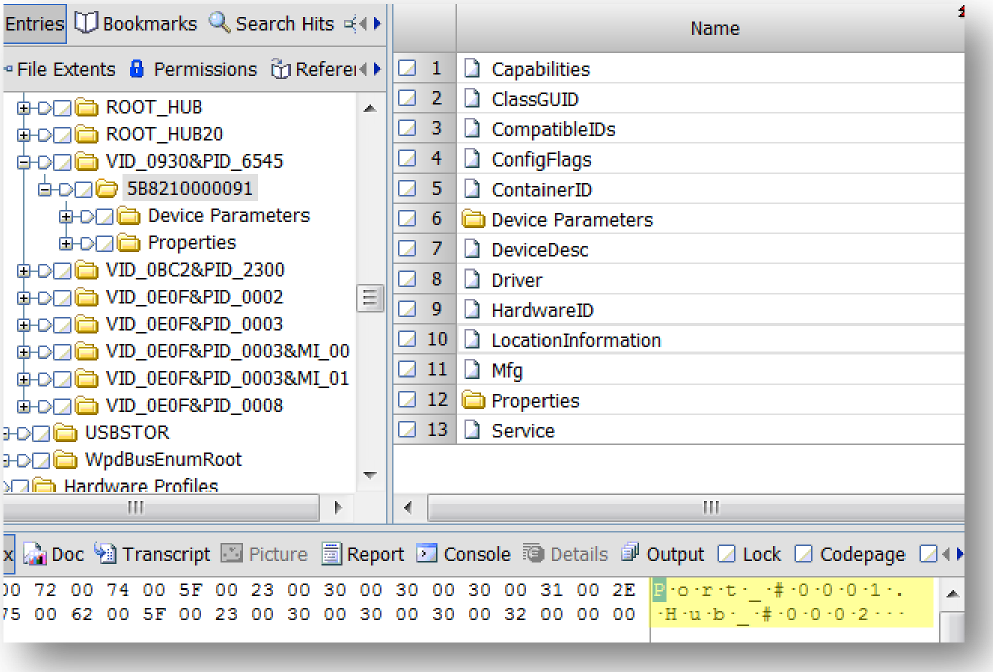

OMG! All that work and I *still* don't have a drive letter?!??

So, at least we know it's probable that a different USB device was plugged into Port 1, Hub 2 of the system *prior* to the USB device that does have an assigned drive letter. Again, check link files, restore points, shadow copies, and "setupapi.dev.log" to figure out what the user may have been doing.

So, why did I go through all of this? Again, I knew there were two USB storage devices plugged into this system, but only one was showing up under Mounted Devices. So I started digging. And digging. And ended up in a pretty deep rabbit hole. Finally, I figured it out, and since I spent nearly a week trying to dig myself out, I thought I'd share.

Now, to figure out if the USB device that's present in Mounted Devices was using the same port and hub as the first one, repeat the steps just described.

The other USB device that was probably plugged in subsequent to Seagate Portable USB Device was USB Flash Memory USB Device, as indicated by its Friendly Name. The Unique Instance ID is 5B8210000091&0. Under the USB key, its Location Information shows the following:



```
ROOT_HUB
ROOT_HUB20
VID_0930&PID_6545
  5B8210000091
    Device Parameters
    Properties
VID_0BC2&PID_2300
VID_0E0F&PID_0002
VID_0E0F&PID_0003
VID_0E0F&PID_0003&MI_00
VID_0E0F&PID_0003&MI_01
VID_0E0F&PID_0008
USBSTOR
WpdBusEnumRoot
Hardware Profiles
```

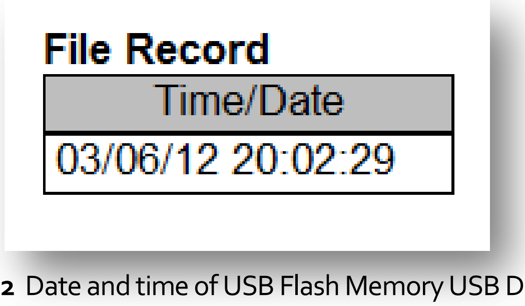
	Name
1	Capabilities
2	ClassGUID
3	CompatibleIDs
4	ConfigFlags
5	ContainerID
6	Device Parameters
7	DeviceDesc
8	Driver
9	HardwareID
10	LocationInformation
11	Mfg
12	Properties
13	Service

```
)0 72 00 74 00 5F 00 23 00 30 00 30 00 30 00 31 00 2E  P·o·r·t·_·#·0·0·0·1·.
75 00 62 00 5F 00 23 00 30 00 30 00 30 00 32 00 00 00  ·H·u·b·_·#·0·0·0·2···
```

Figure 61 Location Information of second USB device

Figure 61 shows that USB Flash Memory USB Device was plugged into Port 1, Hub 2, which is the same location that the Seagate Portable USB Device was plugged into.

In order to check that USB Flash Memory USB Device was plugged in *after* Seagate Portable USB Device, the date and time stamp was checked:



File Record

Time/Date
03/06/12 20:02:29

Figure 62 Date and time of USB Flash Memory USB Device

This date does indeed occur after the time and date for Seagate Portable USB Device, which was March 5, 2012 at 16:29:07.

So, now we can say it is probable that Seagate Portable USB Device does not have an assigned drive letter because USB Flash Memory USB Device was plugged into the same port afterwards. This gave Seagate Portable USB Device's drive letter to USB Flash Memory USB Device. Why only "probable"? Remember that the date and time may not be reliable as there are several situations in which there could be discrepancies.

Knowing the date and time USB Flash Memory USB Device was plugged into the system may help you identify where else you can look for information on Seagate Portable USB Device, such as Restore Points and Shadow Copies. If these exist on the system, previous versions of the Registry may have other data that is useful to your examination.

A timeline can be derived from this information and by examining link files, you may be able to find out which files were being transferred to and from the thumb drive (if any).

SOFTWARE

The SOFTWARE key contains information about the operating system, such as the version, when it was installed, who is the registered owner, who was the last user to log on, and who are the members of a group (if there is one).

%SystemRoot%\Windows\System32\config\SOFTWARE\

Data Stored	Registry Key Location
Current OS Build	Microsoft\Windows NT\CurrentVersion\CurrentBuild
Current OS Version	Microsoft\Windows NT\CurrentVersion\CurrentVersion
OS Edition	Microsoft\Windows NT\CurrentVersion\EditionID
OS Install Date	Microsoft\Windows NT\CurrentVersion\InstallDate
OS Install Location	Microsoft\Windows NT\CurrentVersion\PathName
OS Product Name	Microsoft\Windows NT\CurrentVersion\ProductName
Register Organization	Microsoft\Windows NT\CurrentVersion\RegisteredOrganization
Registered Owner	Microsoft\Windows NT\CurrentVersion\RegisteredOwner
Metro Apps Installed on System	Microsoft\Windows\CurrentVersion\Appx\AppxAllUserStore\Applications
User Account Installed Metro Apps	Microsoft\Windows\CurrentVersion\Appx\AppxAllUserStore\%SID%
Last Logged On User	Microsoft\Windows\CurrentVersion\Authentication\LogonUI\LastLoggedOnUser
Last Logged On SAM User	Microsoft\Windows\CurrentVersion\Authentication\LogonUI\LastLoggedOnSAMUser
Last Logged On SID User	Microsoft\Windows\CurrentVersion\Authentication\LogonUI\LastLoggedOnSIDUser
Group Members	Microsoft\Windows\CurrentVersion\HomeGroup\HME

Data Stored	Registry Key Location
File/Folder Sharing (by SID)	Microsoft\Windows\CurrentVersion\HomeGroup\HME\SharingPreferences\%SID%
Applications that Run at Startup	Microsoft\Windows\CurrentVersion\Run

Final Thoughts

Several times throughout my research I considered dropping this project because I felt like I had gotten myself in way over my head. It was so awful that at one point if I heard "There's a light at the end of the tunnel" or some variation thereof one more time, that person probably would have ended up with a fork in their skull. Part of the problem is the pressure I place on myself and my terrible procrastination habit, but so far, it's worked for me, so why change it?

When I first set out with this research, I only intended to come up with about 35 pages of material. Silly me forgot that when something bothers me, I become almost obsessive about it, and that I *have* to try to understand it, and figure it out and make sure it works a second and third time – this is probably also known as curiosity.

So, the end result is about 70-ish pages of what I hope is usable information for the computer forensic community. As I previously mentioned at the very beginning of this guide, I really do hope to keep this research going. There is so much more that can be researched in Windows 8, and where I stated more work needs to be done, that's where I hope to begin next. For updates, or if you'd like to contribute, please visit http://*propellerheadforensics.com* or follow me on Twitter *@propellerhead23*. Again, please contact me at *propellerheadforensics@gmail.com* for any suggestions, artifacts and objects you have discovered, or criticisms.

Two more things and then I'm done – Thank you Dr. Vincze for supporting this research and allowing me to take on this project. Also, I need to thank my coworkers, Shawn Howell and Theresa Kline, for their support during the last couple of months as they are the ones that had to put up with me for 8 hours everyday, so thank you for being more than just "colleagues".

Index

[i] Windows. (2012). *Windows 8 Consumer Preview*. Windows. Retrieved from http://windows.microsoft.com/en-US/windows-8/consumer-preview.

[ii] Windows 8 Consumer Preview 32-bit Edition downloaded from:
http://windows.microsoft.com/en-US/windows-8/iso.
VMWare Workstation 8: http://downloads.vmware.com/d/info/desktop_end_user_computing/vmware_workstation/8_0.

[iii] FTK Imager 3 downloaded from: http://accessdata.com/support/adownloads.

[iv] Guidance Software's EnCase Forensic: http://www.guidancesoftware.com/

[v] Digital Detective. (2010). *Microsoft Internet Explorer PrivacIE Entries*. Digital Detective. Retrieved from http://blog.digital-detective.co.uk/2010/04/microsoft-internet-explorer-privacie.html.

[vi] Microsoft TechNet. (2012). *How Private Keys Are Stored*. Microsoft TechNet. Retrieved from http://technet.microsoft.com/en-us/library/cc962112.aspx.

[vii] Stanek, W. (2009). Pre-Press Windows 7 Administrator's Pocket Guide (pp. 23). Retrieved from *download.microsoft.com/.../626997_Win7PktConsult_prePress.pdf.*

[viii] Microsoft TechNet. (2010). *Managing Roaming User Data Deployment Guide*. Microsoft TechNet. Retrieved from http://technet.microsoft.com/en-us/library/cc766489(v=ws.10).aspx.

[ix] These findings are based on this author's own independent research using a single test platform. Results may vary under other circumstances, to include changes made to the operating system prior to its official release.

[x] It is unknown at this time how "8wekyb3d8bbwe"is derived or what "AC" signifies in the Communication App's location

[xi] TechNet. (2012). *Digital Certificates (Chapter 6)*. Microsoft TechNet. Retrieved from http://technet.microsoft.com/en-us/library/dd361898.aspx

[xii] More research will need to be conducted in order to determine if the updates in "What's New" also include content from other social networking websites.

[xiii] More research needs to be conducted in order to ascertain whether the "[#]" in the name of the XML file increments sequentially or if it uses a First In, First Out (FIFO) sequence.

[xiv] Windows Dev Center. 2012. *File Streams*. Windows Dev Center – Desktop. Retrieved March 20, 2012, from http://msdn.microsoft.com/en-us/library/windows/desktop/aa364404(v=vs.85).aspx.

[xv] The naming convention of the e-mail stream and the EML file is unknown at this time

[xvi] It is unknown at this time how these contacts are named, as each contact appears to have a random string of numbers associated with their name.

[xvii] All contacts that contained in this Windows 8 image are known to this author.

[xviii] It is unknown at the time of this writing what many of the dates and time are referring to (last update, last backup, when setting was applied, etc.)

[xix] Microsoft Support. (2008). *Windows registry information for advanced users*. Microsoft Support. Retrieved from http://support.microsoft.com/kb/256986.

[xx] Microsoft Support. (2007). *INFO: Working with the FILETIME structure*. Microsoft Support. Retrieved from http://support.microsoft.com/kb/188768.

[xxi] Microsoft Support. (2006). *Information on Last Known Good Control Set*. Microsoft Support. Retrieved from http://support.microsoft.com/kb/101790.

[xxii] MSDN. (2008). *Windows Sensor and Location Platforms*. Microsoft MSDN. Retrieved from http://archive.msdn.microsoft.com/SensorsAndLocation.

[xxiii] It is this author's opinion that forensic examiners will see more information pertaining to Sensors and Location Devices, since Windows 8's goal is to give the user an "immersive" experience.

[xxiv] Carvey, H. (2009). *Windows Forensic Analysis* (pp. 206-211). Burlington, MA: Syngress Publishing, Inc.

[xxv] Dev-Center. (2012). Overview of Container IDs. Windows Dev-Center – Hardware. Retrieved from http://msdn.microsoft.com/en-us/library/windows/hardware/ff549447 (v=vs.85).aspx.

[xxvi] Dev-Center. (2012). *Container IDs*. Windows Dev-Center – Hardware. Retrieved from http://msdn.microsoft.com/en-us/library/windows/hardware/ff540024(v=vs.85).aspx.