Identification of Forensic Artifacts in VMWare Virtualized Computing

Cory Smith, Glenn Dietrich, and Kim-Kwang Raymond Choo(✉)

Department of Information Systems and Cyber Security,
University of Texas at San Antonio, San Antonio, TX 78249, USA
raymond.choo@fulbrightmail.org

Abstract. With popularity of virtualized computing continuing to grow, it is crucial that digital forensic knowledge keeps pace. This research sought out to identify the forensic artifacts and their locations that may be recovered from a VMware Workstation virtual machine running Windows 7 x64. Several common forensic tools were used to conduct this research, namely AccessData's Forensic Toolkit (FTK), FTK Imager, and FTK Registry Viewer. This research verified the processes required to gather digital evidence from a virtual machine disk (VMDK) file, creation of a forensic image, and mounting of evidence into these forensic tools. This research then proceeded to document recovered artifacts and their locations related to system configuration, internet usage, file creation and deletion, user administration, and more.

Keywords: Digital forensics · Forensic artifacts · Virtualization
Virtual machine · VMDK · Forensic Toolkit · FTK Registry Viewer

1 Introduction

Virtualization is often a term that you hear in relation to cloud computing. Virtualization, while it is a separate technology, is one of the most fundamental and critical components which enables versatility and scalability of cloud computing. Virtualization, as defined by VMware is "the process of creating software based representations of something rather than a physical one" [51]. These software-based representations are known as Virtual Machines (VMs). The real benefit of virtualization software is the ability to run 1-N virtual machines on a single physical server – this is done using a Hypervisor. Hypervisors are the software packages that are deployed to "virtualize" a server. These software packages turn the physical machine into a "host", which can then provide its resources to the "guests" contained on it. The hypervisor's role is to dynamically distribute the host's resources to the hosted virtual machines on an as-needed basis [5].

There are two types of hypervisors in use today [47]. Type I hypervisors are known as "Bare Metal Hypervisors", meaning that the hypervisor software is deployed right onto the physical hardware, without the use of any underlying operating systems. Due to the lack of an underlying operating system, the hypervisor is much more efficient when interacting with the host machines resources because the interaction is direct.

X. Lin et al. (Eds.): SecureComm 2017 Workshops, LNICST 239, pp. 85–103, 2018.
https://doi.org/10.1007/978-3-319-78816-6_7

Type II Hypervisors are deployed onto an already running operating system. This model requires the hypervisor to communicate with the operating system to use the host resources. Although it requires an extra step to interact with the host resources, performance delays are not noticeable [5].

For these guest virtual machines to work properly, there are several configuration files that must exist and be accessible by the virtualization software being used. These files are incredibly important for both a virtual machine to run and a digital forensics investigation. For the purposes of this research we are using a type II hypervisor in VMware Workstation Pro.

Table 1 shows the critical configuration files needed for a virtual machine to run properly.

Table 1. VMware configuration files [52]

File extension	File purpose
.log	Keeps a log of all the VM workstation activity
.nvram	Stores the state of the VMs BIOS
.vmdk	Virtual Machine Disk File; stores the contents of the VMs hard drive
.vmsd	Stores centralized metadata about VM snapshots
.vmsn	Snapshot State File; stores the running state of a VM at the time the snapshot is taken
.vmss	Suspended State File; stores the state of a suspended VM
.vmtm	Configuration file containing team metadata
.vmx	Primary configuration file for the VM, stores all the settings of the VM
.vmxf	Supplemental configuration file for VMs that are in a team

Virtualization reduces the need for hundreds or thousands of physical servers. This reduction in equipment means smaller datacenters, thus less overhead costs. The cost differential alone is enough for businesses to give serious consideration to virtualization capabilities. From power consumption, to heating and cooling cost, the savings can be extensive. Other benefits of virtualization include the ability to rapidly scale enterprise resources to meet consumer needs, test software on many different operating systems, and provide a cost-effective way to achieve fault tolerance for your enterprise services. With benefits like these, it is easy to understand why virtualization is being adopted faster than ever. The annual report from RightScale outlines cloud adoption trends from the previous year. The 2017 report surveyed 1,002 respondents and determined that 95% of organizations surveyed are experimenting with Infrastructure as a Service (IaaS). In addition, the use of multiple clouds per organization increased from 82% to 85% since 2016. In addition, 23% of enterprises with more than 1,000 employees have over 1,000 virtual machines in VMware [40]. This adoption underscores the need for the ability to perform thorough digital forensic investigations on virtualized computers.

Digital evidence is created anytime a user takes any action – criminal or not – on a computer. When the actions performed on a computer are criminal, or aid in a criminal act, digital forensics is performed to collect this evidence. The National Institute of Justice defines digital evidence as:

> "*... information stored or transmitted in binary form that may be relied on in court. It can be found on a computer hard drive, a mobile phone, a personal digital assistant (PDA), a CD, and a flash card in a digital camera, among other places. Digital evidence is commonly associated with electronic crime, or e-crime, such as child pornography or credit card fraud.*" [15].

When searching for digital evidence on a windows machine, the focus should be on gathering information pertaining to file uploads/downloads, files/folder created, opened, and removed, programs installed, executed, and removed, usage of various accounts on the machine, external device usage, and usage of the browsers by each account on the machine [30]. For the purposes of this research, we will be focusing on similar user activities as listed above.

With the popularity of virtualization, it is critical that there be documented processes and procedures on how to perform digital forensics investigations in this environment; however, it is not that simple. There are several concerns when it comes to digital forensics in a virtualized environment, several of which I will present in the final section of this paper as proposed future research topics. One of the primary concerns is the lack of documented forensic artifacts that can be recovered from a virtual machines disk file. In a traditional forensics investigation, the investigator has access to the physical media and can create images as needed for their investigation. These forensic images provide a wealth of data. This research aims to identify the forensic artifacts that can be recovered from a virtualized computer running Windows 7 by using AccessData's Forensic Toolkit to investigate the supporting files used by the VM. This research aims to strengthen the digital forensics field and associated techniques to keep up with the ever-changing technology landscape.

In the next section, we will discuss related works, their strengths and opportunities, and ultimately the driving force for this research. In Sects. 3 and 4, we will then discuss the steps taken to ensure a sound research environment, the steps taken to recover artifacts of interest and detailed findings. In the last section, we will finish with our conclusions and proposed future research efforts.

2 Related Literature

Digital forensics, in some form or another, has been around since Cliff Stoll famously investigated a mere seventy-five cent discrepancy between two accounting systems at the Lawrence Berkley National Laboratory in 1986 [45]. While only a few decades have passed, the advances in technology have been immense. As such, there has been an increased need for the ability to perform digital forensics investigations against these new technologies. Digital forensics is not an old practice, and the forensic artifacts recoverable from physical media have been well documented [14, 30]. However, emphasis on applying these processes to virtualized computing environments is limited [28, 29, 56]. Virtualization has seen a significant increase in popularity over the past few years, requiring that virtual machine forensic be researched just as heavily, if not more, than traditional physical media [16, 40].

Shavers [43] discussed the process of acquiring a forensic image from a virtual machine disk (VMDK) file and subsequently using it to create a new virtual machine. This allowed investigators to safely examine the VM and its contents. However, he did not go into detail about the specific artifacts that could be recovered, or their relevance to a digital forensics investigation.

Martini and Choo [4] proposed a six-step process for the remote programmatic collection of evidential data from virtual machines and demonstrated the utility of their process using VMware vCloud as a case study.

Cruz and Atkison [2] focused on the process of recovering a fragmented or corrupted VMDK file from a hypervisor by using a write blocker and the physical hard drive of the host. They explained the possibilities of creating a forensic image from the recovered VMDK file and using common forensic tools to analyze this image. They did not go into detail regarding the artifacts that could be recovered from this image.

While several research efforts have focused on identifying the difficulties involved with cloud forensics and the processes of performing forensic investigation on both cloud servers and client devices [9–13, 32–34], few have focused on identifying and recovering artifacts from the guest system [3, 6, 18, 49]. This research sought to answer the question, “What are the forensic artifacts and their locations that can be recovered from a VM running Windows 7 x64?”.

3 Experiment Setup

In this section, we describe the tools used, the configuration of the lab environment, and the process of seeding the VM for the investigation.

- VMware Workstation 12 Pro, version 12.1.0 build-3272444
- Windows 7 x64 ISO file for creating the VM (configured with 4 GB Memory, 2 Processors, and 80 GB of hard disk)
- AccessData’s Forensic Toolkit (FTK), version 6.0.3.5
- FTK Imager, version 3.4.2.6
- FTK Registry Viewer, version 1.8.3.0

VMware Workstation serves as the type II hypervisor for this research, allowing the creation of a VM from the Windows 7 x64 ISO file. AccessData’s Forensic Toolkit is a commonly used commercial digital forensics tool. FTK allows for quick mounting of forensic images, which can then be searched for forensic artifacts such as deleted files, configuration changes, and internet history [20]. For this research, the forensic image was created from the Virtual Machine Disk File (VMDK). FTK Imager converts the VMDK file to raw (dd) format which is needed to mount the image into FTK as evidence [26]. FTK Registry Viewer, also from AccessData, is packaged with FTK and provides the ability to view registry hives contained within the mounted evidence [37]. We will discuss in Sect. 4.1 how registry hives can be used in a forensic investigation.

The environment for this research was contained to a single physical host. The physical host ran VMware Workstation Pro, enabling me to create a virtual machine from the Windows 7 x64 ISO. The forensic tools from AccessData were also installed on the same physical host. Once the VM was configured and powered on, transactions required for subsequent investigation were then generated. This consisted of normal user activity: browsing the internet, uploading and downloading files, accessing, creating and deleting files, and creating and removing users. Transactions were conducted over a two-week period between 06122017 and 06222017 to ensure enough activity took place allowing for a thorough investigation.

To perform the investigation of the VMDK file, we first had to create a forensic image that could be mounted into FTK. To do this, FTK Imager was used, which takes a VMDK file and converts it to the raw image format needed by FTK. A forensic image is a bit-for bit replication of either an entire disk or a single partition and is like a "snapshot"; it captures the full state of the disk or partition [50]. This is done so that investigators do not modify or alter the original evidence in any way throughout their investigation (this is one of the forensic principles emphasized by McKemmish [41]). If the forensic image gets corrupted, then the image can be discarded and a new image restored. Hashing algorithms are commonly used to prove the integrity of the evidence.

Figure 1 shows that creating a forensic image with FTK Imager is as easy as pointing to the file location for the VMDK file. Figure 2 shows the completed image summary. In order to maintain proper chain of custody, this information must be recorded and maintained for the life of the evidence. Figure 3 is used to prove the integrity of the forensic image that was created. Two hashing algorithms are used to prove integrity throughout the creation of the image. First, MD5 and SHA1 hashes are calculated for the original evidence. Then, once processing has completed, additional hashes are calculated using the same algorithms as before. The comparison of these hashes can be used to validate evidence integrity during the case and potential subsequent trial [26].

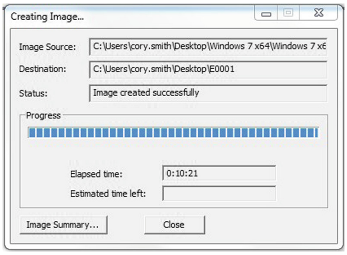
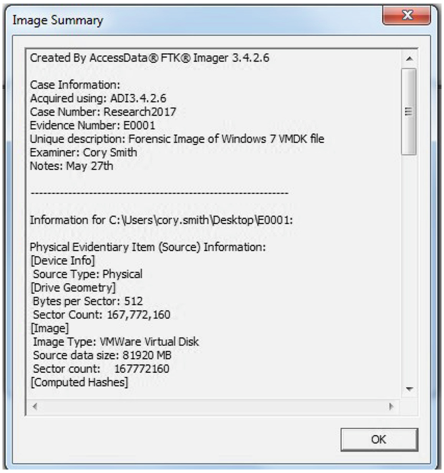


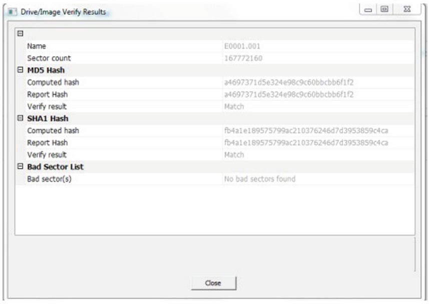
Image Source: C:\Users\cory.smith\Desktop\Windows 7 x64\Windows 7 x6
Destination: C:\Users\cory.smith\Desktop\E0001
Status: Image created successfully
Elapsed time: 0:10:21
Estimated time left:

Fig. 1. Creating an image in FTK Imager.



```
Created By AccessData® FTK® Imager 3.4.2.6

Case Information:
Acquired using: ADI3.4.2.6
Case Number: Research2017
Evidence Number: E0001
Unique description: Forensic Image of Windows 7 VMDK file
Examiner: Cory Smith
Notes: May 27th

--------------------------------------------------------------

Information for C:\Users\cory.smith\Desktop\E0001:

Physical Evidentiary Item (Source) Information:
[Device Info]
 Source Type: Physical
[Drive Geometry]
 Bytes per Sector: 512
 Sector Count: 167,772,160
[Image]
 Image Type: VMWare Virtual Disk
 Source data size: 81920 MB
 Sector count:    167772160
[Computed Hashes]
```

Fig. 2. Completed image summary in FTK Imager.



⊟	
Name	E0001.001
Sector count	167772160
⊟ **MD5 Hash**	
Computed hash	a4697371d5e324e98c9c60bbcbb6f1f2
Report Hash	a4697371d5e324e98c9c60bbcbb6f1f2
Verify result	Match
⊟ **SHA1 Hash**	
Computed hash	fb4a1e189575799ac210376246d7d3953859c4ca
Report Hash	fb4a1e189575799ac210376246d7d3953859c4ca
Verify result	Match
⊟ **Bad Sector List**	
Bad sector(s)	No bad sectors found

Fig. 3. FTK Imager integrity hashes.

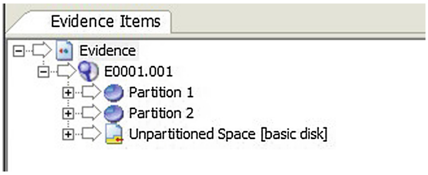


Fig. 4. Available partitions in FTK.

Once a forensic image has been created, it can easily be mounted into FTK as evidence. The FTK user guide easily outlines this process [27]. Figure 4 shows the accessible partitions in FTK, once the forensic image has been mounted as evidence.

4 Findings: Forensic Artifacts

Any time an action is taken on a computer, clues from that action are left behind, regardless of whether that action was taken by a human or a program. Digital artifacts are the pieces of information that can be gathered by recovering what is left behind. Any time a digital crime is committed, the investigation of these artifacts can provide a wealth of information as to what really happened, who did it, and the event's timeline (35). This section details the key digital artifacts that were recovered along with their respective location within the VM.

4.1 Registry Hive Artifacts

The windows registry is the authoritative source for configuration settings in the Windows operating system; every configuration change manipulates a key kept in the registry. These keys are separated out into special groupings, known as "registry hives". There are four main registry hives, as explained by Microsoft: the HKEY_-CURRENT_CONFIG, HKEY_CURRENT_USER, HKEY_LOCAL_MACHINE, and HKEY_USERS (39). Each hive maintains its own tree structure with several sub-directories containing different key value pairs, as well as its own ".log" file which maintains a history of all changes made to that registry hive [39].

HKEY_CURRENT_USER provides configuration settings for the current logged on user, and is located in "C:\Users\%USERNAME%\NTUSER.dat" [25]. This hive is critical to a forensics investigation as it maintains all the unique settings for a certain user. It also keeps track of several pieces of metadata to provide enhanced functionality to the user. By accessing the NTUSER.dat file through FTK, and opening it with FTK Registry Viewer, the investigator can retrieve critical artifacts such as URLs typed into a browser, recently accessed documents, commands typed into the Windows “Run” utility, searches from the start menu, user assist keys for application execution, and the user’s shell bags.

Within the HKEY_LOCAL_MACHINE hive, there are several sub-directories of interest; they are the Software, Security, SAM, and System folders. This hive is located in the “C:\Windows\System32\config” folder [39]. All hives present in the NTFS file system can be viewed through FTK Registry Viewer. This hive provides configurations and settings for the machine itself, and can be incredibly useful to an investigator. By interrogating the HKLM\System registry file, an investigator could recover the machine name, system time and time offset, profiles for recently connected USB drives and the shim cache. In this same directory, there is a “.alt” file, which maintains the registry key value pairs for the SYSTEM hive. By investigating the HKLM\Software registry hive, a list of installed applications was recovered. The HKLM\Security registry hive is where all system policy information is maintained, while the HKLM\SAM registry file is a collection of user credentials [1].

Per Microsoft, the HKEY_CURRENT_CONFIG contains the hardware state for the local machine and is used to compare the current configuration to the standard configuration. This can prove useful in order to determine hardware changes during a certain period. HKEY_USERS contain the standard configurations that are assigned to all new users to the system. These configurations are assigned at creation of the user [36]. An unapproved change to this key could indicate a user trying to manipulate configurations and privileges for any users created in the future.

4.2 NTFS File System Artifacts

The NTFS file system is based on a hierarchical file system, therefore, much of a digital forensics investigation is focused on the recovery of files. There are three main locations that an investigator needs to pay special attention to, namely: the Master File Table, Recycle Bin, and orphaned files.

The Master File Table relates to files like the Registry relates to configurations; it is an authoritative source for all files on the system, as well as their attributes. For every file on the NTFS file system, there is at least one entry in the MFT. This entry contains all metadata about the file such as file name, location, timestamps, permissions and content [31]. Traditionally a user assumes that deleting a file from the computer removes it permanently. In reality, deletion of the file only sets the “Active/Inactive” field to “Inactive”. Mark Stam explains the methods of extracting data from the master file table to recreate files, even if they have been marked as “Inactive” [44]. In addition to being able to recreate the file, additional metadata can be carved out of the MFT Attributes; these include the file creation time, last accessed time, as well as the last time the file was modified. There are several tools that can be used to easily parse and display the contents of the MFT, such as MFT Ripper and Analyze MFT [17, 25, 54]. Once a file has been marked “inactive” it is only a matter of time until the OS reuses the MFT entry for that file. Because of this, it can be concluded that the longer the time between deletion of files and the investigation, the greater the chances are that the file will not be able to be recovered.

Recycle Bin artifacts can be located using FTK in the "$Recycle.Bin" folder in the root directory (\$Recycle.Bin). The recycle bin is a holding place for files and folder that have been deleted by the user (Right Click > Delete). Files in this location can easily be recovered using FTK.

In the recycle bin, a folder is created for each user that has logged into the system. These folders are named using the user's SID. It is rare that any files should exist outside of a particular user's SID; this may represent a user attempting to hide a file on the file system since the root of the recycle bin is not viewable by any user in Windows Explorer [21]. Each time a user deletes a file, there are two entries created in their $Recycle.Bin. The first entry begins with "$I" and contains the metadata of the file: the date recycled, original file path, name of the file, and the size of the file. The second entry begins with "$R" and contains the actual data of the file. When files are deleted, the name of the file is converted to an ID string; the matching of the "$I" and "$R" entries provide the needed information to the forensic investigator.

The recycle bin artifacts allow forensic investigators to recover files that the user believes they have deleted. The findings suggested that deleted files could still be recoverable after the recycle bin had been emptied; given the investigation occurred in a timely manner. This echoed the findings of Quick and Choo [9–12], who demonstrated that data that had been removed using CCleaner and Eraser could still be recovered. Microsoft implements a wiping technique in Windows 7 for files that have been deleted from the recycle bin, this often occurs at the next restart of the machine, and indicates that the longer the time is between the deletion from the recycle bin and the investigation, the less likely it is to recover a file [21]. These recovered files can provide a wealth of information as to what the user was doing. Along with logon events, discussed in Sect. 4.4, it is simple to tie a user to the action of deleting a file. We refer the interested reader to [7] for a recent survey on Windows 7 anti-forensics approaches and countermeasures.

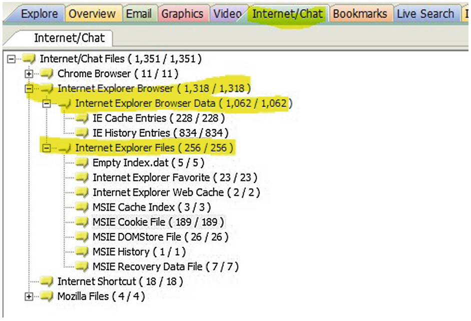
Orphaned files are a special type of file in the NTFS file system that get created when the files parent folder gets deleted from the file system. FTK provides an easy way to search through orphaned files. It is possible to recover a file that has been orphaned, given that the files MFT entry has not been entirely overwritten [21].

4.3 Web Browser Artifacts

It is no secret that we are living in a digital age, and the amount of time that we spend online is increasing. What many people do not realize is the amount of information that your computer records while you are browsing the internet [42]. When it comes to inappropriate or malicious actions, the information is stored in the same manner. Extensive browser forensics was completed throughout this research for Internet Explorer, resulting in the recovery of several key artifacts that painted the picture of what the user was doing during their online sessions.

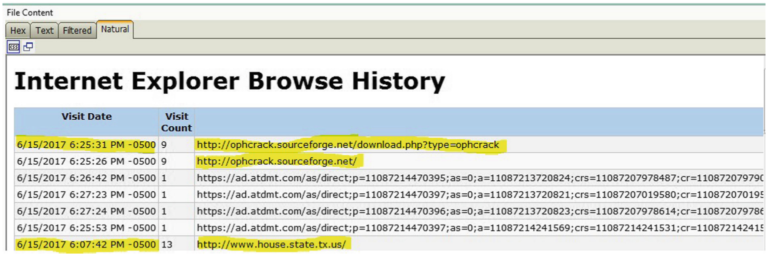
Outside of the registry, there are three major artifacts that can be recovered regarding internet usage: the browser cache, user created bookmarks, and browser cookies. The browser cache is not only a history of the sites that a user had visited, but also a cached copy of that site. The browser creates an entry in the browser cache every time a user visits a site to decrease the load time for any future visits to that site [23].

The browser cache files exist in, “C:\Users\%USERNAME%\AppData\Local\Microosoft\Windows\WebCache”. Using FTK, an investigator can view a listing of the sites visited as well as recreate them in the “html” tab of the tool (Figs. 5 and 6). This artifact provides a great deal of information about the active user’s activity during a certain period.



```
⊟ Internet/Chat Files ( 1,351 / 1,351 )
  ⊞ Chrome Browser ( 11 / 11 )
  ⊟ Internet Explorer Browser ( 1,318 / 1,318 )
    ⊟ Internet Explorer Browser Data ( 1,062 / 1,062 )
        IE Cache Entries ( 228 / 228 )
        IE History Entries ( 834 / 834 )
    ⊟ Internet Explorer Files ( 256 / 256 )
        Empty Index.dat ( 5 / 5 )
        Internet Explorer Favorite ( 23 / 23 )
        Internet Explorer Web Cache ( 2 / 2 )
        MSIE Cache Index ( 3 / 3 )
        MSIE Cookie File ( 189 / 189 )
        MSIE DOMStore File ( 26 / 26 )
        MSIE History ( 1 / 1 )
        MSIE Recovery Data File ( 7 / 7 )
    Internet Shortcut ( 18 / 18 )
  ⊞ Mozilla Files ( 4 / 4 )
```

Fig. 5. Internet tab in FTK.



Internet Explorer Browse History

Visit Date	Visit Count	
6/15/2017 6:25:31 PM -0500	9	http://ophcrack.sourceforge.net/download.php?type=ophcrack
6/15/2017 6:25:26 PM -0500	9	http://ophcrack.sourceforge.net/
6/15/2017 6:26:42 PM -0500	1	https://ad.atdmt.com/as/direct;p=11087214470395;as=0;a=11087213720824;crs=11087207978487;cr=110872079790
6/15/2017 6:27:23 PM -0500	1	https://ad.atdmt.com/as/direct;p=11087214470397;as=0;a=11087213720821;crs=11087207019580;cr=110872070195
6/15/2017 6:27:24 PM -0500	1	https://ad.atdmt.com/as/direct;p=11087214470396;as=0;a=11087213720823;crs=11087207978614;cr=110872079786
6/15/2017 6:25:53 PM -0500	1	https://ad.atdmt.com/as/direct;p=11087214470397;as=0;a=11087214241569;crs=11087214241531;cr=110872142415
6/15/2017 6:07:42 PM -0500	13	http://www.house.state.tx.us/

Fig. 6. WebCache for admin user.

User created bookmarks are saved links to sites that have been visited in the past and marked for quick access. This artifact lives in, “C:\Users\%USERNAME%\Favorites”, and can provide additional information regarding past browsing history [25]. No favorites were created during this research.

Cookies are created by visited sites and attached to the user for future use. These cookies are saved in a file on the local machine and referenced any time a user visits a site that requires them. They can be used to store authentication tokens, location data, and other general user information [23]. Cookies files are stored in “C:\users\Administrator\AppData\Roaming\Microsoft\Windows\Cookies\”. They are difficult to interpret because the data is only meaningful to the site that created it. However, the existence of a cookie indicates that the user visited the referenced site it at least once, and the recovery of the cookies can provide additional information to the activities of the user.

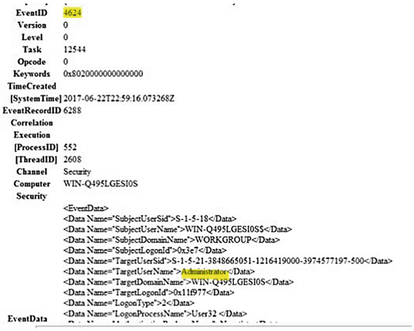
There are several artifacts of interest regarding internet usage stored in the registry; such as URL's typed by the user, form auto complete information, and browser preferences.

4.4 Windows System Log Artifacts

The windows operating system keeps thorough logs regarding what occurred on, and to, a machine. These logs provide a wealth of information to an investigator [38]. As such, they should be paid special attention to. Certain logs of interest are the Windows Event Logs and the Windows Change Logs.

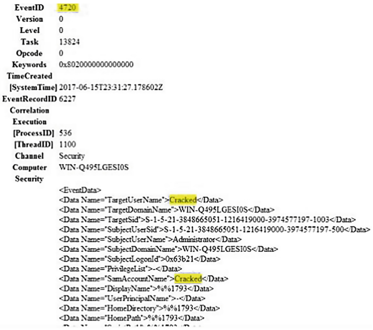
Windows event logs provide a wealth of data about what occurred on a computer. There are three main event logs on the Windows OS, namely: the Security, System, and Application. These three logs are all located in the same directory, "C:\Windows\System32\Winevt\Logs". When viewed in FTK, one would see each log with a ".evtx" file extension. The information contained in these logs are invaluable to an investigator when recreating a timeline of events.

One of the most informative logs is the security log, as it tracks all security events that occur on the computer, such as logon events, failed logons, creation of users, permission changes for users, removing users, and execution of the processes. This information paints a clear picture of what occurred while a specific user was logged in. It also records who attempted to logon to a machine. Figure 7 shows the information that gets logged when a user authenticates onto the machine. This information can be used to recreate a timeline of events. Figure 8 shows the information that is logged when a new user is created. Often attackers will create additional users on a compromised system to create alternate access methods should they ever lose access. In addition to creating new users, an attacker may also attempt to change the password for an existing account, information for this type of event can also be recovered in the Security log (Fig. 9). Examining the creation of new users can lead investigators to another account that needs to be scoped into their investigation. Figure 10 shows the information that is logged when a user is removed from the system.



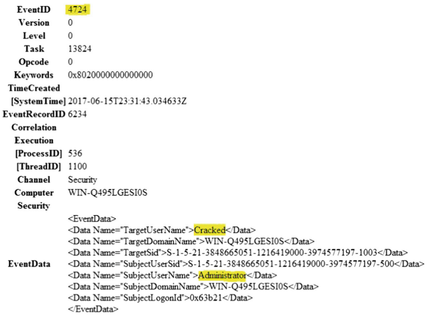
```
        EventID  4624
        Version  0
          Level  0
           Task  12544
         Opcode  0
       Keywords  0x8020000000000000
    TimeCreated
   [SystemTime]  2017-06-22T22:59:16.073268Z
  EventRecordID  6288
    Correlation
      Execution
    [ProcessID]  552
     [ThreadID]  2608
        Channel  Security
       Computer  WIN-Q495LGESI0S
       Security
                 <EventData>
                 <Data Name="SubjectUserSid">S-1-5-18</Data>
                 <Data Name="SubjectUserName">WIN-Q495LGESI0S$</Data>
                 <Data Name="SubjectDomainName">WORKGROUP</Data>
                 <Data Name="SubjectLogonId">0x3e7</Data>
                 <Data Name="TargetUserSid">S-1-5-21-3848665051-1216419000-3974577197-500</Data>
                 <Data Name="TargetUserName">Administrator</Data>
                 <Data Name="TargetDomainName">WIN-Q495LGESI0S</Data>
                 <Data Name="TargetLogonId">0x11f977</Data>
                 <Data Name="LogonType">2</Data>
      EventData  <Data Name="LogonProcessName">User32 </Data>
```

Fig. 7. Admin logon event.



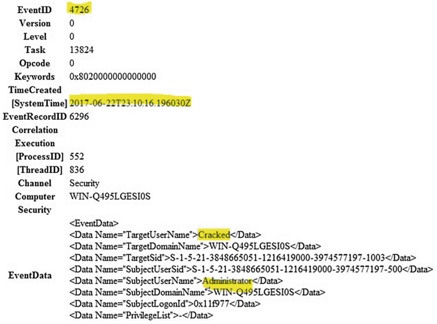
```
        EventID  4720
        Version  0
          Level  0
           Task  13824
         Opcode  0
       Keywords  0x8020000000000000
    TimeCreated
   [SystemTime]  2017-06-15T23:31:27.178602Z
  EventRecordID  6227
    Correlation
      Execution
    [ProcessID]  536
     [ThreadID]  1100
        Channel  Security
       Computer  WIN-Q495LGESI0S
       Security
                 <EventData>
                 <Data Name="TargetUserName">Cracked</Data>
                 <Data Name="TargetDomainName">WIN-Q495LGESI0S</Data>
                 <Data Name="TargetSid">S-1-5-21-3848665051-1216419000-3974577197-1003</Data>
                 <Data Name="SubjectUserSid">S-1-5-21-3848665051-1216419000-3974577197-500</Data>
                 <Data Name="SubjectUserName">Administrator</Data>
                 <Data Name="SubjectDomainName">WIN-Q495LGESI0S</Data>
                 <Data Name="SubjectLogonId">0x63b21</Data>
                 <Data Name="PrivilegeList">-</Data>
                 <Data Name="SamAccountName">Cracked</Data>
                 <Data Name="DisplayName">%%1793</Data>
                 <Data Name="UserPrincipalName">-</Data>
                 <Data Name="HomeDirectory">%%1793</Data>
                 <Data Name="HomePath">%%1793</Data>
```

Fig. 8. User creation event.



```
         EventID  4724
         Version  0
           Level  0
            Task  13824
          Opcode  0
        Keywords  0x8020000000000000
     TimeCreated
    [SystemTime]  2017-06-15T23:31:43.034633Z
   EventRecordID  6234
     Correlation
       Execution
     [ProcessID]  536
      [ThreadID]  1100
         Channel  Security
        Computer  WIN-Q495LGESI0S
        Security
                  <EventData>
                  <Data Name="TargetUserName">Cracked</Data>
                  <Data Name="TargetDomainName">WIN-Q495LGESI0S</Data>
                  <Data Name="TargetSid">S-1-5-21-3848665051-1216419000-3974577197-1003</Data>
       EventData  <Data Name="SubjectUserSid">S-1-5-21-3848665051-1216419000-3974577197-500</Data>
                  <Data Name="SubjectUserName">Administrator</Data>
                  <Data Name="SubjectDomainName">WIN-Q495LGESI0S</Data>
                  <Data Name="SubjectLogonId">0x63b21</Data>
                  </EventData>
```

Fig. 9. Password change attempt event.



```
         EventID  4726
         Version  0
           Level  0
            Task  13824
          Opcode  0
        Keywords  0x8020000000000000
     TimeCreated
    [SystemTime]  2017-06-22T23:10:16.196030Z
   EventRecordID  6296
     Correlation
       Execution
     [ProcessID]  552
      [ThreadID]  836
         Channel  Security
        Computer  WIN-Q495LGESI0S
        Security
                  <EventData>
                  <Data Name="TargetUserName">Cracked</Data>
                  <Data Name="TargetDomainName">WIN-Q495LGESI0S</Data>
                  <Data Name="TargetSid">S-1-5-21-3848665051-1216419000-3974577197-1003</Data>
       EventData  <Data Name="SubjectUserSid">S-1-5-21-3848665051-1216419000-3974577197-500</Data>
                  <Data Name="SubjectUserName">Administrator</Data>
                  <Data Name="SubjectDomainName">WIN-Q495LGESI0S</Data>
                  <Data Name="SubjectLogonId">0x11f977</Data>
                  <Data Name="PrivilegeList">-</Data>
```

Fig. 10. User deletion event.

Events generated by running applications are logged in the application event log. According to Microsoft, the application event log includes errors, warnings, and informational messages [22]. This information can be useful to a forensic investigator in many ways. Once a user's timeline has been established, through logon event, the investigator can then determine which applications were run during that time. Analyzing the application event log can assist in determining what that user was doing inside of each application.

The system event log maintains the starting and stopping of processes on the machine. This log is similar to the application event log as it contains errors, warnings, and informational messages pertaining to processes running on the machine. The log can be very useful to an investigator as many attacks utilize malware with known process names. The system event logs can be monitored and analyzed for these processes and identify malicious programs running on the machine [24].

Just like volume shadow copies can be restored to reverse changes to the operating system, change logs maintained by the Windows operating system can be rolled back to revert previously made changes to the file system. These artifacts include the "$LogFile" and "$UsnJrnl". Both include information regarding changes to the system. The $LogFile is much more detailed than the $UsnJrnl and is located in the root directory "\$LogFile". This file contains changes made to the file system such as creation and deletion of files and directories. The $UsnJrnl stores much less information regarding system changes than the $LogFile and is located in, "\$Extend\$UsnJrnl" off of the root directory [IR Book]. Once the file system has been mounted into FTK, as discussed earlier, the investigator can easily navigate to it and view or extract these files. There are several open source tools that can be used to intelligently analyze and display the contents of these files, such as LogFileParser [25]. Upon analysis of these artifacts, an investigator will better be able to determine which changes were made to a system over a certain time period.

4.5 Prefetch File Artifacts

Prefetch files are a critical piece of a forensic investigation. If available, they can provide investigators with a list of applications that were executed during a certain period of time. Located in "C:\Windows\Prefetch", this directory contains a ".pf" entry for every application that was executed on the machine [19]. Microsoft designed the ".pf" as a performance optimization file. The Microsoft OS tracks the first 10 s of every application's start up process and creates a ".pf" file with this information. This file is then referenced every time the application runs in order to decrease the application start time [25]. According to Luttgens, Pepe, and Mandia, the existence of a prefetch provides critical information about which programs were executed on a machine; such as name, number of executions, execution path, and when it was executed [25]. This information helps build out the attack timeline and points investigators to further evidence locations. Most importantly, even if a program has been uninstalled from the machine, the existence of a prefetch file is proof that the program existed and was utilized.

Prefetch file artifacts are crucial to a digital forensics investigation, but what if they are missing? The absence of an artifact could be an artifact in itself (e.g. signs of antiforensic activities). If the prefetch files are missing, or only exist up to a certain point, then it could indicate that the attacker was more knowledgeable than the average user. The foresight to disable the creation of prefetch files indicates a potentially skilled attacker. Disabling the creation of prefetch files modifies a key in the registry hive HKEY_LOCAL_MACHINE, this key can be viewed using FTK Registry Viewer, as discussed in Sect. 4.1. The value of the "EnablePrefetcher" registry key set to "0" which indicates that the value is disabled [16].

When available, the prefetch files can easily be recovered using FTK. Investigators can simply navigate to the Prefetch directory and view these records. There are also several open source tools that can be used to view the prefetch files and parse the data in many ways [19, 55].

4.6 LNK File Artifacts

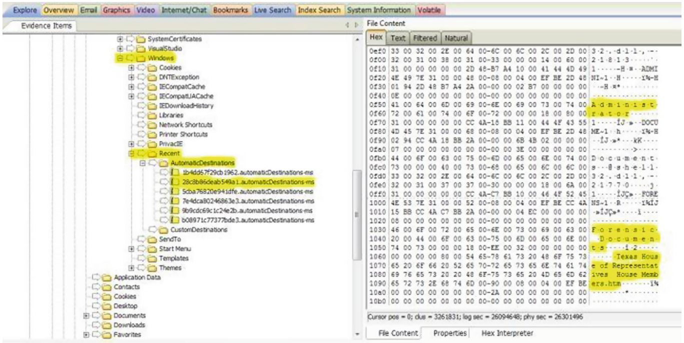
LNK files are another name for shortcut files. These types of files are the result of user action (Right Click > Create Shortcut) or program execution/install. Any time a user or program creates a shortcut, a LNK file is created in, "C:\Users\%USERNAME%\AppData\Roaming\Microsoft\Windows\Recent\".

Luttgens et al. [25] provide detailed steps for recovering the LNK files for Windows 7. Using FTK to recover the LNK files, one could recover the local path, modified, and created timestamp, as well as the file size and volume serial number.

LNK files can provide a wealth of data to forensic investigators and contribute to the re-creation of an attack timeline. These files can provide the "what," "when" and "where" of an attacker's activity while they were on a system [25].

4.7 Jump List Artifacts

Microsoft Jump Lists keep a running history of the recently used items for an application. For example, when Microsoft Word is pinned to the task bar, one could right click and choose from several of the recently opened Word documents. A user can also select to pin certain options to the jump list menu for future use. There are two types of jump lists, namely: "automatic destinations" and "custom destinations". The automatic destinations jump list is populated with recently used programs, while the custom destinations jump list is populated with the options that a user has 'pinned' to the jump list [25].



```
0ef0  33 00 32 00 2E 00 64 00-6C 00 6C 00 2C 00 2D 00  3·2·.·d·l·l·,·-·
0f00  32 00 31 00 38 00 31 00-33 00 00 00 14 00 60 00  2·1·8·1·3·····`·
0f10  31 00 00 00 00 00 2D 48-B7 A4 10 00 41 44 4D 49  1·····-H·¤··ADMI
0f20  4E 49 7E 31 00 00 48 00-08 00 04 00 EF BE 2D 48  NI~1··H·····ï¾-H
0f30  01 94 2D 48 B7 A4 2A 00-00 00 02 B7 00 00 00 00  ··-H·¤*·········
0f40  0E 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ················
0f50  41 00 64 00 6D 00 69 00-6E 00 69 00 73 00 74 00  A·d·m·i·n·i·s·t·
0f60  72 00 61 00 74 00 6F 00-72 00 00 00 18 00 80 00  r·a·t·o·r·······
0f70  31 00 00 00 00 00 CC 4A-18 BB 11 00 44 4F 43 55  1·····ÌJ·»··DOCU
0f80  4D 45 7E 31 00 00 68 00-08 00 04 00 EF BE 2D 48  ME~1··h·····ï¾-H
0f90  02 94 CC 4A 18 BB 2A 00-00 00 6B 4B 02 00 00 00  ··ÌJ·»*···kK····
0fa0  07 00 00 00 00 00 00 00-00 00 3E 00 00 00 00 00  ··········>·····
0fb0  44 00 6F 00 63 00 75 00-6D 00 65 00 6E 00 74 00  D·o·c·u·m·e·n·t·
0fc0  73 00 00 00 40 00 73 00-68 00 65 00 6C 00 6C 00  s···@·s·h·e·l·l·
0fd0  33 00 32 00 2E 00 64 00-6C 00 6C 00 2C 00 2D 00  3·2·.·d·l·l·,·-·
0fe0  32 00 31 00 37 00 37 00-30 00 00 00 18 00 6A 00  2·1·7·7·0·····j·
0ff0  31 00 00 00 00 00 CC 4A-C7 BB 10 00 46 4F 52 45  1·····ÌJÇ»··FORE
1000  4E 53 7E 31 00 00 52 00-08 00 04 00 EF BE CC 4A  NS~1··R·····ï¾ÌJ
1010  15 BB CC 4A C7 BB 2A 00-00 00 04 EC 00 00 00 00  ·»ÌJÇ»*····ì····
1020  08 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ················
1030  46 00 6F 00 72 00 65 00-6E 00 73 00 69 00 63 00  F·o·r·e·n·s·i·c·
1040  20 00 44 00 6F 00 63 00-75 00 6D 00 65 00 6E 00   ·D·o·c·u·m·e·n·
1050  74 00 73 00 00 00 18 00-EE 00 32 00 00 00 00 00  t·s·····î·2·····
1060  00 00 00 00 80 00 54 65-78 61 73 20 48 6F 75 73  ······Texas Hous
1070  65 20 6F 66 20 52 65 70-72 65 73 65 6E 74 61 74  e of Representat
1080  69 76 65 73 20 20 48 6F-75 73 65 20 4D 65 6D 62  ives  House Memb
1090  65 72 73 2E 68 74 6D 00-90 00 08 00 04 00 EF BE  ers.htm·······ï¾
10a0  00 00 00 00 00 00 00 00-2A 00 00 00 00 00 00 00  ········*·······
10b0  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ················
Cursor pos = 0; clus = 3261831; log sec = 26094648; phy sec = 26301496
```

Fig. 11. Administrators jump list entry for "Texas House of Representatives.html".

Figure 11 shows that from a jump list entry, an investigator can uncover the user who accessed the file, the file name and the original path. Each jump list entry has a corresponding application ID, these ID's remain fairly static and can easily be looked up. In this research, a jump list entry for the network reconnaissance tool NMAP was discovered, which pointed to a file titled "Research.xml". It was determined that this file had been deleted from the file system, hence the need to interrogate the $Recycle. Bin directory. The deleted file was recovered and an analysis of this files showed that the active user launched NMAP and carried out a network scan of "scanme.Nmap.org". This is piece of the entire puzzle that allows a forensic investigator to determine the "who," "what," "when," "where," and "how" of an incident.

4.8 Installed Application Artifacts

When beginning an investigation, one thing the investigator will want to do is gather a list of all installed applications. This will provide several pieces of information that assist with tooling decisions. Certain forensic tools are used for certain types of applications and artifacts. LUTTGENS, PEPE and MANDIA provide an overview of the directories in which you can recover artifacts from installed applications; these include the default application installation directory, default application data directories, registry uninstall information, and default registry configuration data locations [25]. Using FTK, the "Uninstall" registry key was recovered. The recovered key lists the currently installed application, by expanding one of these entries an investigator can view several pieces of information about the application, such as: instance ID, help link, install source, and display name.

Such information can be used by the investigator to better select the tools they will use throughout the remainder of the investigation. The list of installed applications, combined with the timeline recovered from the Windows Security Event logs and the Prefetch file artifacts, will assist in defining what the attacker did during the refined timeline, as well as what other activity may have been taking place on that machine. For example, using FTK, it was determined that both NMAP and OphCrack had been installed and run. Nmap is a popular network reconnaissance tool used for mapping out target networks, and OphCrack is a popular password cracking tool. This provides insight into the user's activities and can point to other artifacts.

4.9 Windows Task Scheduler Artifacts

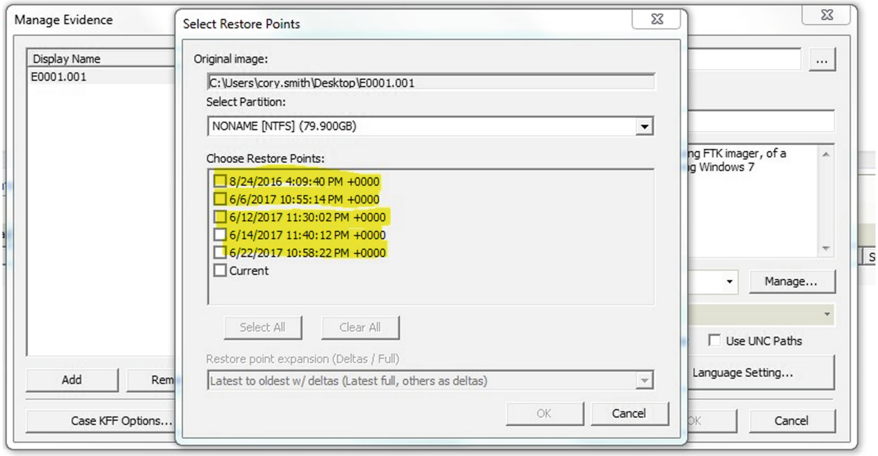
Similar to the way the UNIX Operating system schedules reoccurring tasks through the Crontab, Windows can schedule reoccurring tasks using ".job" files; this is done through the Windows Task Scheduler [48]. These tasks can be rule-based, time-based, or state-based [46]. These ".job" files can be found easily using FTK in "C:\Windows \System32\Tasks" [25].

An attacker can use scheduled tasks for many things: creating backdoors, adding and removing users, modifying accesses, or cleaning up files and directories after the attack. Recovering these files can provide information as to what was ran after the attack, what may have been scheduled in the past to enable the attack, and any jobs that have yet to run and should be prevented.

4.10 Windows Restoration Point Artifacts

Microsoft introduced the Volume Shadow Copy Service (VSS) in Windows Server 2003. This service is used to generate backups of application and operating system data, known as restoration points [53]. VSS provides the ability to generate reoccurring backups of data to protect against the potential loss of data in the future. Should a system be compromised, it can be restored from the latest shadow copy created. There are several activities that can trigger a shadow copy to be created; such as an update, program installation, or scheduled task [25].

During an investigation, an investigator may come across the remnants of a critical file or application that has since been deleted. By restoring a shadow copy from around the time the file or application was created or installed, they may be able to recover the full contents of the file. In my research, it was proven that volume shadow copy recovery with FTK was simple, allowing an investigator to interrogate the shadow copy file system as if it was the true hard disk. Figure 12 shows the number of shadow copies that were available to restore from a single VMDK file. Once restored, an investigator could then interrogate the file system for that shadow copy.



Manage Evidence

Display Name: E0001.001

Select Restore Points

Original image: C:\Users\cory.smith\Desktop\E0001.001

Select Partition: NONAME [NTFS] (79.900GB)

Choose Restore Points:

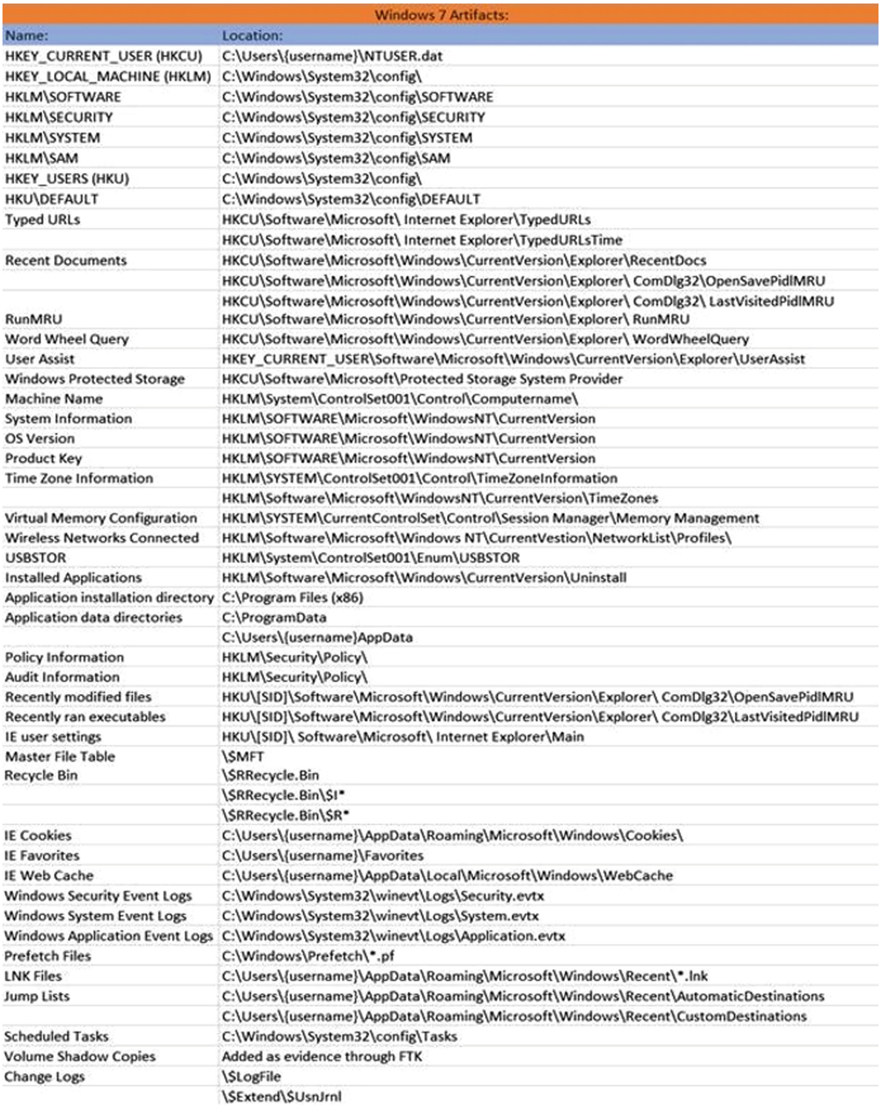
□ 8/24/2016 4:09:40 PM +0000
□ 6/6/2017 10:55:14 PM +0000
□ 6/12/2017 11:30:02 PM +0000
□ 6/14/2017 11:40:12 PM +0000
□ 6/22/2017 10:58:22 PM +0000
□ Current

Restore point expansion (Deltas / Full): Latest to oldest w/ deltas (Latest full, others as deltas)

ng FTK imager, of a
g Windows 7

□ Use UNC Paths

Fig. 12. Mountable volume shadow copies from virtual machine disk file.



Windows 7 Artifacts:	
Name:	Location:
HKEY_CURRENT_USER (HKCU)	C:\Users\{username}\NTUSER.dat
HKEY_LOCAL_MACHINE (HKLM)	C:\Windows\System32\config\
HKLM\SOFTWARE	C:\Windows\System32\config\SOFTWARE
HKLM\SECURITY	C:\Windows\System32\config\SECURITY
HKLM\SYSTEM	C:\Windows\System32\config\SYSTEM
HKLM\SAM	C:\Windows\System32\config\SAM
HKEY_USERS (HKU)	C:\Windows\System32\config\
HKU\DEFAULT	C:\Windows\System32\config\DEFAULT
Typed URLs	HKCU\Software\Microsoft\ Internet Explorer\TypedURLs
	HKCU\Software\Microsoft\ Internet Explorer\TypedURLsTime
Recent Documents	HKCU\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs
	HKCU\Software\Microsoft\Windows\CurrentVersion\Explorer\ ComDlg32\OpenSavePidlMRU
	HKCU\Software\Microsoft\Windows\CurrentVersion\Explorer\ ComDlg32\ LastVisitedPidlMRU
RunMRU	HKCU\Software\Microsoft\Windows\CurrentVersion\Explorer\ RunMRU
Word Wheel Query	HKCU\Software\Microsoft\Windows\CurrentVersion\Explorer\ WordWheelQuery
User Assist	HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\UserAssist
Windows Protected Storage	HKCU\Software\Microsoft\Protected Storage System Provider
Machine Name	HKLM\System\ControlSet001\Control\Computername\
System Information	HKLM\SOFTWARE\Microsoft\WindowsNT\CurrentVersion
OS Version	HKLM\SOFTWARE\Microsoft\WindowsNT\CurrentVersion
Product Key	HKLM\SOFTWARE\Microsoft\WindowsNT\CurrentVersion
Time Zone Information	HKLM\SYSTEM\ControlSet001\Control\TimeZoneInformation
	HKLM\Software\Microsoft\WindowsNT\CurrentVersion\TimeZones
Virtual Memory Configuration	HKLM\SYSTEM\CurrentControlSet\Control\Session Manager\Memory Management
Wireless Networks Connected	HKLM\Software\Microsoft\Windows NT\CurrentVestion\NetworkList\Profiles\
USBSTOR	HKLM\System\ControlSet001\Enum\USBSTOR
Installed Applications	HKLM\Software\Microsoft\Windows\CurrentVersion\Uninstall
Application installation directory	C:\Program Files (x86)
Application data directories	C:\ProgramData
	C:\Users\{username}AppData
Policy Information	HKLM\Security\Policy\
Audit Information	HKLM\Security\Policy\
Recently modified files	HKU\[SID]\Software\Microsoft\Windows\CurrentVersion\Explorer\ ComDlg32\OpenSavePidlMRU
Recently ran executables	HKU\[SID]\Software\Microsoft\Windows\CurrentVersion\Explorer\ ComDlg32\LastVisitedPidlMRU
IE user settings	HKU\[SID]\ Software\Microsoft\ Internet Explorer\Main
Master File Table	\$MFT
Recycle Bin	\$RRecycle.Bin
	\$RRecycle.Bin\$I*
	\$RRecycle.Bin\$R*
IE Cookies	C:\Users\{username}\AppData\Roaming\Microsoft\Windows\Cookies\
IE Favorites	C:\Users\{username}\Favorites
IE Web Cache	C:\Users\{username}\AppData\Local\Microsoft\Windows\WebCache
Windows Security Event Logs	C:\Windows\System32\winevt\Logs\Security.evtx
Windows System Event Logs	C:\Windows\System32\winevt\Logs\System.evtx
Windows Application Event Logs	C:\Windows\System32\winevt\Logs\Application.evtx
Prefetch Files	C:\Windows\Prefetch\*.pf
LNK Files	C:\Users\{username}\AppData\Roaming\Microsoft\Windows\Recent\*.lnk
Jump Lists	C:\Users\{username}\AppData\Roaming\Microsoft\Windows\Recent\AutomaticDestinations
	C:\Users\{username}\AppData\Roaming\Microsoft\Windows\Recent\CustomDestinations
Scheduled Tasks	C:\Windows\System32\config\Tasks
Volume Shadow Copies	Added as evidence through FTK
Change Logs	\$LogFile
	\$Extend\$UsnJrnl

Fig. 13. Summary of Windows 7 artifacts.

5 Conclusion

The fast-paced changes in computing require a continual increase in the knowledge of digital forensics and the ability to apply this knowledge to new technologies. Virtualized computing has become the new norm, and the ability to perform a forensics investigation against a VM is critical in today's incident response process. During an investigation, those artifacts that identify who acted on a system, what they carried out, and when it was done must be recovered to develop an activity timeline. Understanding what was done, and how it was done is crucial to beginning the next steps of the incident response process. Once all the pertinent artifacts have been recovered, responders can begin developing a remediation plan as well as designing security controls to protect them from similar events in the future.

This research focused on the identification of forensic artifacts, and their locations, in virtualized computing to provide foundational knowledge to future digital forensic investigations. Specifically, this research described the process of gathering digital evidence from the virtual machine disk file, creating forensic images, and interrogating the NTFS file system. A detailed list of artifacts recovered within the timeline of this research was also presented.

Figure 13 documents forensic artifacts along with their locations that were recovered throughout this research. This is not a definitive list of artifacts that can be recovered from the Windows OS, rather a listing of artifacts that were recoverable within the strict timeline of this research, and should serve as the foundation for a digital forensics investigation against a VM running Windows 7.

As forensic tools progress, more artifacts will potentially be recovered. There is still much research that can be done pertaining to digital forensics in virtualized environments, such as the ability to recover a deleted VMDK file from the physical host to recreate a VM and provide a forensic image for investigation and the identification of forensic artifacts from virtual machines running different operating systems.

References

1. Admin. Password Recovery. Password Recovery RSS. Top Password Software, Inc., 31 May 2013. https://www.top-password.com/blog/tag/windows-samregistry-file/. Accessed 11 July 2017
2. Atkison, T., Cruz, J.C.F.: Digital Forensics on a Virtual Machine. Rep. (n.d.). http://atkison.cs.ua.edu/papers/ACMSE11_JF.pdf. Accessed 18 July 2017
3. Aziz, A.S.A., Fouad, M.M., Hassanien, A.E.: Cloud computing forensic analysis: trends and challenges. In: Hassanien, A.E., Fouad, M.M., Manaf, A.A., Zamani, M., Ahmad, R., Kacprzyk, J. (eds.) Multimedia Forensics and Security. ISRL, vol. 115, pp. 3–23. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-44270-9_1
4. Martini, B., Choo, K.-K.R.: Remote programmatic vCloud forensics: a six-step collection process and a proof of concept. In: Proceedings of 13th IEEE International Conference on Trust, Security and Privacy in Computing and Communications, TrustCom 2014, pp. 935–942 (2014)
5. Kleyman, B.: Hypervisor 101: Understanding the Virtualization Market. Data Center Knowledge. Penton, 03 August 2012. http://www.datacenterknowledge.com/archives/2012/08/01/hypervisor-101-a-lookhypervisor-market/. Accessed 14 June 2017
6. Birk, D., Christoph, W.: Technical Issues of Forensic Investigations in Cloud Computing Environments. Rep. (n.d.)
7. Eterovic-Soric, B., Choo, K.-K.R., Mubarak, S., Ashman, H.: Windows 7 antiforensics: a review and a novel approach. J. Forensic Sci. **62**(4), 1054–1070 (2017)

8. Esposito, C., Castiglione, A., Pop, F., Choo, K.-K.R.: Challenges of connecting edge and cloud computing: a security and forensic perspective. IEEE Cloud Comput. **4**(2), 13–17 (2017)
9. Quick, D., Choo, K.-K.R.: Digital droplets: microsoft SkyDrive forensic data remnants. Future Gener. Comput. Syst. **29**(6), 1378–1394 (2013)
10. Quick, D., Choo, K.-K.R.: Dropbox analysis: data remnants on user machines. Digit. Invest. **10**(1), 3–18 (2013)
11. Quick, D., Choo, K.-K.R.: Forensic collection of cloud storage data: does the act of collection result in changes to the data or its metadata? Digit. Invest. **10**(3), 266–277 (2013)
12. Quick, D., Choo, K.-K.R.: Google drive: forensic analysis of data remnants. J. Netw. Comput. Appl. **40**, 179–193 (2014)
13. Quick, D., Choo, K.-K.R.: Pervasive social networking forensics: intelligence and evidence from mobile device extracts. J. Netw. Comput. Appl. **86**, 24–33 (2017)
14. Dean, B.: Best Practices in Browser Forensics. IANS. IANS (n.d.). https://www.iansresearch.com/insights/reports/best-practices-in-browser-forensics. Accessed 15 June 2017
15. Digital Evidence and Forensics. National Institute of Justice, 14 April 2016. https://www.nij.gov/topics/forensics/evidence/digital/Pages/welcome.aspx. Accessed 23 July 2017
16. Disabling Prefetch. Microsoft Developer Network. Microsoft (n.d.). https://msdn.microsoft.com/en-us/library/ms940847(v=winembedded.5).aspx. Accessed 18 July 2017
17. Dkovar. Dkovar/analyzeMFT. GitHub. GitHub, Inc., 16 July 2017. https://github.com/dkovar/analyzeMFT. Accessed 13 July 2017
18. Dykstra, J., Sherman, A.T.: Understanding Issues in Cloud Forensics: Two Hypothetical Case Studies. Rep. (2011)
19. Forensic Analysis of Prefetch Files in Windows. Magnet Forensics Inc. Magnet Forensics, 6 August 2014. https://www.magnetforensics.com/computerforensics/forensic-analysis-of-prefetch-files-in-windows/. Accessed 15 July 2017
20. Forensic Toolkit (FTK). AccessData (n.d.). http://accessdata.com/products-services/forensic-toolkit-ftk. Accessed 15 July 2017
21. FTK BootCamp Windows 7 Forensics - Recycle Bin. AccessData (n.d.). http://accessdata.com/. Accessed 16 July 2017
22. How To: Access the Application Event Log. Microsoft TechNet. Microsoft (n.d.). https://technet.microsoft.com/en-us/library/ms166507(v=sql.90).aspx. Accessed 19 July 2017
23. How to Clear Cache, Cookies and History. What Is Cache, Cookies, and History and How Do You Clear Them… Content (n.d.). http://www.pgcconline.com/technicalSupport/clearCache/clearCache.html. Accessed 17 July 2017
24. How to View the System Log in Event Viewer. Microsoft TechNet. Microsoft (n.d.). https://technet.microsoft.com/en-us/library/aa996634(v=exchg.65).aspx. Accessed 19 July 2017
25. Luttgens, J., Pepe, M., Mandia, K.: Incident Response & Computer Forensics, 3rd edn. McGraw-Hill/Osborne, New York (2014)
26. Jensen, C.: FTK Imager User Guide. AccessData, Lindon, 21 March 2012
27. Jensen, C.: FTK User Guide. AccessData, Lindon, 21 January 2015
28. Choo, K.-K.R., Esposito, C., Castiglione, A.: Evidence and forensics in the cloud: challenges and future research directions. IEEE Cloud Comput. **4**(3), 14–19 (2017)
29. Choo, K.-K.R., Herman, M., Iorga, M., Martini, B.: Cloud forensics: state-of-the-art and future directions. Digit. Invest. **18**, 77–78 (2016)
30. Lee, R.: SANS Digital Forensics and Incident Response Blog. SANS Digital Forensics and Incident Response Blog | New Windows Forensics Evidence of Poster Released | SANS Institute. SANS Institute, 04 June 2015. https://digitalforensics.sans.org/blog/2015/06/04/new-windows-forensics-evidence-of-poster-released. Accessed 18 June 2017

31. Master File Table. Master File Table (Windows). Microsoft (n.d.). https://msdn.microsoft.com/en-us/library/windows/desktop/aa365230(v=vs.85).aspx. Accessed 12 July 2017
32. Cahyani, N.D.W., Martini, B., Choo, K.-K.R., Muhammad Nuh Al-Azhar, A.K.B.P.: Forensic data acquisition from cloud-of-things devices: windows smartphones as a case study. Concurr. Comput.: Pract. Exp. **29**(14) (2017)
33. Cahyani, N.D.W., Ab Rahman, N.H., Glisson, W.B., Choo, K.-K.R.: The role of mobile forensics in terrorism investigations involving the use of cloud storage service and communication apps. MONET **22**(2), 240–254 (2017)
34. Ab Rahman, N.H., Cahyani, N.D.W., Choo, K.-K.R.: Cloud incident handling and forensic-by-design: cloud storage as a case study. Concurr. Comput.: Pract. Exp. **29**(14) (2017)
35. Ab Rahman, N.H., Glisson, W.B., Yang, Y., Choo, K.-K.R.: Forensic-by-design framework for cyber-physical cloud systems. IEEE Cloud Comput. **3**(1), 50–59 (2016)
36. Predefined Keys. Predefined Keys (Windows). Microsoft (n.d.). https://msdn.microsoft.com/en-us/library/windows/desktop/ms724836(v=vs.85).aspx. Accessed 11 July 2017
37. Product Downloads. AccessData (n.d.). http://accessdata.com/product-download/registry-viewer-1.8.1.3. Accessed 14 July 2017
38. Do, Q., Martini, B., Looi, J., Wang, Y., Choo, K.-K.R.: Windows event forensic process. In: Peterson, G., Shenoi, S. (eds.) DigitalForensics 2014. IAICT, vol. 433, pp. 87–100. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-44952-3_7
39. Registry Hives. Registry Hives (Windows). Microsoft (n.d.). https://msdn.microsoft.com/en-us/library/windows/desktop/ms724877(v=vs.85).aspx. Accessed 16 July 2017
40. RightScale 2017 State of the Cloud Report. Rep. RightScale, Inc (n.d.). http://assets.rightscale.com/uploads/pdfs/RightScale-2017-State-of-the-Cloud-Report.pdf?mkt_tok=eyJpIjoiTjJOaE1qTm1aRFJoTm1ZeSIsInQiOiJGQlB2WklLRWp4OFU1Mm1FS1dzRW9DOFQwaXhuT0lPYVlzcktCMmdUeEVaRk84dTlGQnFIaWNxM0k0WnNIaUgyS2ZRdGs3Nk9hUFZNeXFJVU94ZmFRdU55ZVB5NzF5WjNRQXUrbW1INlhLTUtYdEY5bmdtbFJ3VVFQbXV0YWczNCJ9. Accessed 10 June 2017
41. McKemmish, R.: What is forensic computing? Trends Issues Crime Crim. Justice **118**, 1–6 (1999)
42. Pokharel, S., Choo, K.-K.R., Liu, J.: Mobile cloud security: an adversary model for lightweight browser security. Comput. Stand. Interfaces **49**, 71–78 (2017)
43. Shavers, B.: Virtual Forensics: A Discussion of Virtual Machines Related to Forensic Analysis. Rep. Virtual Forensics (n.d.). https://www.forensicfocus.com/downloads/virtual-machines-forensics-analysis.pdf. Accessed 24 June 2017
44. Stam, M.: Lab FTK Imager: File Carving Using the MFT. 8 Bits. Techblog, 09 October 2009. http://stam.blogs.com/8bits/2009/10/lab-ftk-imager-file-carvingusing-the-mft-.html. Accessed 10 July 2017
45. Stoll, C.: The Cuckoo's Egg: Tracking a Spy Through the Maze of Computer Espionage. Pocket, New York (2005)
46. Task Scheduler. Task Scheduler (Windows). Microsoft (n.d.). https://msdn.microsoft.com/en-us/library/windows/desktop/aa383614(v=vs.85).aspx. Accessed 14 July 2017
47. Tholeti, B.P.: Learn about Hypervisors, System Virtualization, and How It Works in a Cloud Environment. Hypervisors, Virtualization, and the Cloud, 23 September 2011. https://www.ibm.com/developerworks/cloud/library/clhypervisorcompare/. Accessed 10 June 2017
48. 2.4 .JOB File Format. [MS-TSCH]: .JOB File Format. Microsoft (n.d.). https://msdn.microsoft.com/en-us/library/cc248285.aspx. Accessed 19 July 2017
49. Urias, V.E., Young, J.W.: Hypervisor assisted forensics and incident response in the cloud. Publication no. 10.1109. IEEE (2016)
50. Vandeven, S.: Forensic Images: For Your Viewing Pleasure. Publication. SANS Institute (2014)

51. Virtualization Technology & Virtual Machine Software. VMWare. VMware, Inc., 20 July 2017. https://www.vmware.com/solutions/virtualization.html. Accessed 22 July 2017
52. VMware Workstation 5.5. What Files Make Up a Virtual Machine? VMware, Inc (n.d.). https://www.vmware.com/support/ws55/doc/ws_learning_files_in_a_vm.html. Accessed 12 June 2017
53. Volume Shadow Copy Service. Windows Server. Microsoft (n.d.). https://technet.microsoft.com/en-us/library/ee923636(v=ws.10).aspx. Accessed 15 July 2017
54. Welcome to MyKey Technology. MFT Ripper. MyKey Technology Inc (n.d.). http://mftripper.com/. Accessed 18 July 2017
55. WinPrefetchView V1.35. View the Content of Windows Prefetch (.pf) Files. Nir Sofer (n.d.). http://www.nirsoft.net/utils/win_prefetch_view.html. Accessed 16 July 2017
56. Teing, Y.-Y., Dehghantanha, A., Choo, K.-K.R., Yang, L.T.: Forensic investigation of P2P cloud storage services and backbone for IoT networks: BitTorrent Sync as a case study. Comput. Electr. Eng. **58**, 350–363 (2017)