# Windows Registry analysis

# Objective:-

Analyzing the windows registry in the provided encase image. To facilitate this, following tasks will be performed.

* Converting and Mounting an E0? (EnCase image) on-the-fly over a Linux box using xmount
* Using RegRipper to analyze Windows registry hives for the purpose of extracting evidence
* Comprehending the different ‘plugins’ available for RegRipper and the purpose that they serve
* Using RegRipper in command-line mode and grasping the available options

# Why Registry Analysis

Virtually everything done in Windows refers to or is recorded into the Registry. The Registry is referenced in one way or another with every action taken by the user.

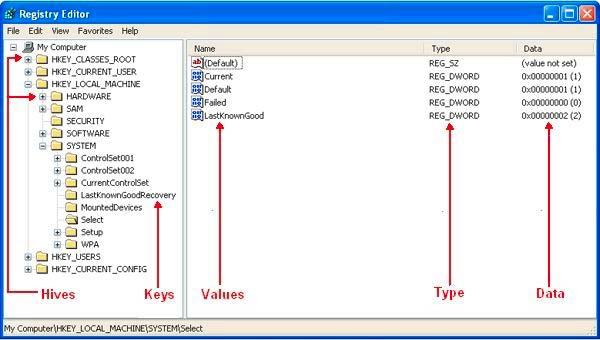
The registry is viewed as the central hierarchical database used in Microsoft Windows 9x, Windows CE, Windows NT, and Windows 2000 used to store information necessary to configure the system for one or more users, applications and hardware devices.

# Structure of the Windows Registry

By opening the Registry Editor (by typing 'regedit' in the run window), the Registry can be seen as one unified 'file system'.

The left-hand pane, also known as the key pane contains an organized listing of what appear to be folders. The five most hierarchal folders are called 'hives' and begin with 'HKEY'.

Although five hives can be seen, only two of these are actually 'real', HKEY\_USERS (HKU) and HKEY\_LOCAL\_MACHINE (HKLM). The other three are shortcuts or aliases to branches within one of the two hives. Each of these five hives is composed of keys, which contain values and subkeys. Values are the names of certain items within a key, which uniquely identify specific values pertaining to the operating system, or to applications that depend upon that value.



1.HKEY\_CLASSES\_ROOT (HKCR)

Information stored here ensures that the correct program opens when it is executed in Windows Explorer. It also contains further details on drag-and-drop rules, shortcuts, and information on the user interface. Alias for: HKLM\Software\Classes

2.HKEY\_CURRENT\_USER (HKCU)

Contains configuration information for the user who is currently logged into the system, including user's folders, screen colors, and Control Panel settings. Alias for a user specific branch in HKEY\_USERS. The generic information usually applies to all users and is HKU\.DEFAULT.

3.HKEY\_LOCAL\_MACHINE (HKLM)

Contains machine hardware-specific information that the operating system runs on. It includes a list of drives mounted on the system and generic configurations of installed hardware and applications.

4.HKEY\_USERS (HKU)

Contains configuration information of all user profiles on the system, which concerns application configurations, and visual settings.

5.HKEY\_CURRENT\_CONFIG (HCU)

Stores information about the systems current configuration. Alias for: HKLM\Config\profile

# Registry Examination

The Registry as a Log

All Registry keys contain a value associated with them called the 'LastWrite' time, which is very similar to the last modification time of a file. This value is stored as a FILETIME structure and indicates when the Registry Key was last modified.

Knowing the LastWrite time of a key can allow a forensic analyst to infer the approximate date or time an event occurred. And although one may know the last time a Registry key was modified, it still remains difficult to determine what value was actually changed. Using the Registry as a log is most helpful in the correlation between the LastWrite time of a Registry key and other sources of information, such as MAC (modified, accessed, or created) times found within the file system

# Autorun Locations

Autorun locations are Registry keys that launch programs or applications during the boot process. It is generally a good practice to look here depending on the case of examination. For instance, if a computer is suspected to have been involved in a system intrusion case, autorun locations should be looked at. If the user denies their involvement then it's possible their own system was compromised and used to initiate the attack. In a case such as this, the autorun locations could prove that the system had a trojan backdoor installed leaving it vulnerable for an attacker to use at their discretion.

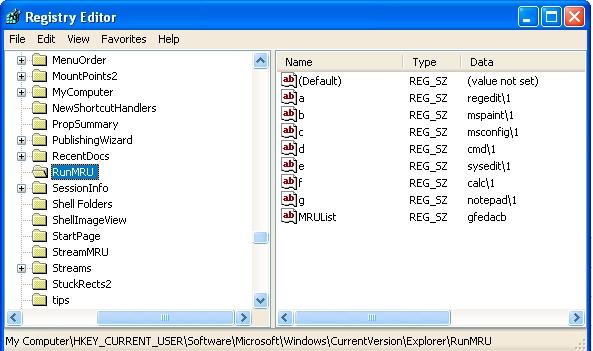
List of common autorun locations:

HKLM\Software\Microsoft\Windows\CurrentVersion\Runonce  
HKLM\Software\Microsoft\Windows\CurrentVersion\policies\Explorer\Run  
HKLM\Software\Microsoft\Windows\CurrentVersion\Run  
HKCU\Software\Microsoft\Windows NT\CurrentVersion\Windows\Run  
HKCU\Software\Microsoft\Windows\CurrentVersion\Run  
HKCU\Software\Microsoft\Windows\CurrentVersion\RunOnce  
(ProfilePath)\Start Menu\Programs\Startup

# MRU lists

MRU, or 'most recently used' lists contain entries made due to specific actions performed by the user. There are numerous MRU lists located throughout various Registry keys. The Registry maintains these lists of items in case the user returns to them in the future. It is basically similar to how the history and cookies act to a web browser. One example of an MRU list located in the Windows Registry is the RunMRU key. When a user types a command into the 'Run' box via the Start menu, the entry is added to this Registry key. The location of this key is HKCU\Software\Microsoft\Windows\ CurrentVersion\Explorer\RunMRU and its contents can be seen in Figure 2. The chronological order of applications executed via 'Run' can be determined by looking at the Data column of the 'MRUList' value. The first letter of this is 'g', which tells us that the last command typed in the 'Run' window was to execute notepad. Also, the LastWrite time of the RunMRU key will correlate with the last application executed in 'Run', or in this case application 'g'.

With the information provided from the RunMRU key, an examiner can gain a better understanding of the user they are investigating and the applications that are being used.



# UserAssist

The UserAssist key, HCU\Software\Microsoft\Windows\CurrentVersion \Explorer\UserAssist, contains two or more subkeys which have long hexadecimal names that appear as globally unique identifiers (GUIDs). Each subkey records values that pertain to specific objects the user has accessed on the system, such as Control Panel applets, shortcut files, programs, etc. These values however, are encoded using a ROT- 13 encryption algorithm, sometimes known as a Caesar cipher. This particular encryption technique is quite easy to decipher, as each character is substituted with the character 13 spaces away from it in the ASCII table. A much faster and easier method to decipher this code is with the use of an online ROT-13 decoder, such as http://www.edoceo.com/utilis/rot13.php.

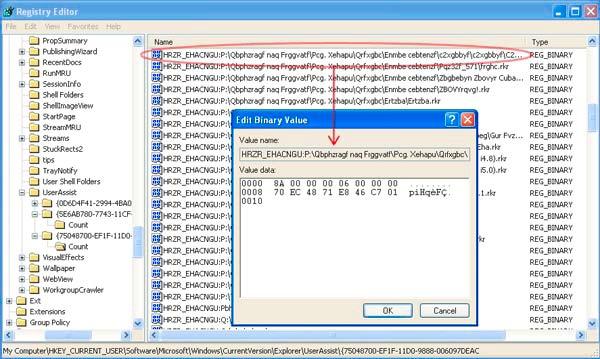
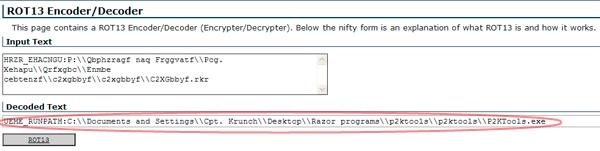


Figure UserAssist Key



ROT-13 cipher decoded

With the UserAssist key, a forensic examiner can gain a better understanding of what types of files or applications have been accessed on a particular system. Even though these entries are not definitive, for they cannot be associated with a specific date and time, it may still indicate a specific action by the user.

For instance, in the example of Figure the decoded value can show a potential amount of information. First, it tells the name of the user profile - 'Cpt. Krunch' - from which the .exe was executed from. Cpt. Krunch could also indicate a handle or an alias of some sort. Second, by researching 'p2ktools.exe', it tells that it is a program used for editing and managing Motorola cell phones. Finally, it shows the user has the p2ktools folder in a parent directory called 'Razor programs', which is located on their desktop. Not only does this give the location of where similar programs may reside, but the name of this directory is a good indicator that the suspect has a Motorola Razor cell phone. If so, that too should be seized for further analysis.

# Wireless Networks

Wireless networks today are popular and are only becoming more popular. A wireless ethernet card picks up wireless access points within its range, which are identified by their SSID or service set identifier. When an individual connects to a network or hotspot the SSID is logged within Windows XP as a preferred network connection. Unsurprisingly, this can be found in the Registry in the HKLM\SOFTWARE\ Microsoft\WZCSVC\Parameters\Interfaces key.

When opening this Registry key there may be subkeys beneath it, like UserAssist, that look like GUIDs. The contents of these should contain the values 'ActiveSettings' and 'Static#0000'. There may be additional values that begin with 'Static#' and are sequentially numbered. In the binary data of these 'Static#' values are the network SSIDs of all the wireless access points that system has connected to. This can be seen by right clicking the value and selecting 'modify', as shown in figure below

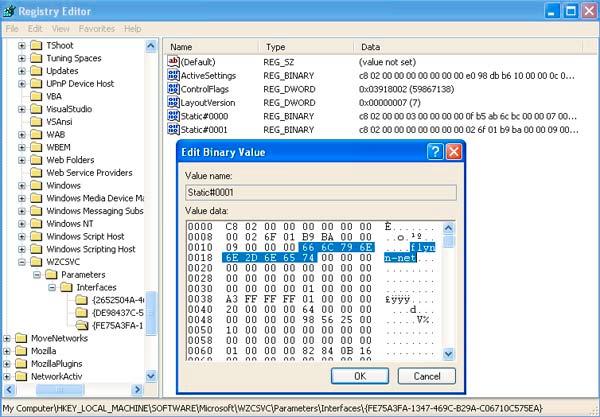


Figure - SSID 'flynn-net'

In addition to logging the name of the SSID, Windows also logs the network settings of that particular connection - such as the IP address, DHCP domain, subnet mask, etc. The Registry key in which this can be found is HKLM\SYSTEM\ControlSet001\ Services\Tcpip\Parameters\Interfaces\, which is illustrated in Figure 4a.

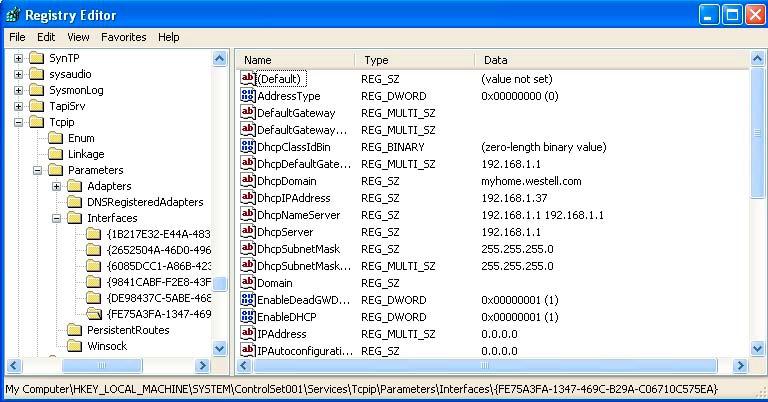


Figure - Network settings of SSID 'flynn-net'

Based on this wireless network information, a Forensic examiner can determine if a user connected to specific wireless access point, the timeframe, and their IP address they were assigned by the DHCP server. For instance, if it were a case about a child pornography suspect that was war-driving to various network connections and using them illegally, these methods would be very useful. Given the suspect's computer to run an analysis on would make it possible to see what network connections they were using and the IP address that was assigned to further support a subpoena of the ISP.

# LAN Computers

Windows XP implements a network mapping tool called My Network Place, which allows users to easily find other users within a LAN or Local Area Network. A computer on a properly configured LAN should be able to display all the users on that network through My Network Place. This list of users or computers, like many other things, is stored in the Registry. Therefore, even after the user is no longer connected to the LAN, the list of devices still remain, including desktop computers, laptops, and printers. The Registry key where this information is stored is HKCU\Software\Microsoft\ Windows\CurrentVersion\Explorer\ComputerDescriptions.

The ComputerDescriptions key is useful in determining whether or not a user was connected to certain computers or belonged to a specific LAN. Figure below displays the output of this key.

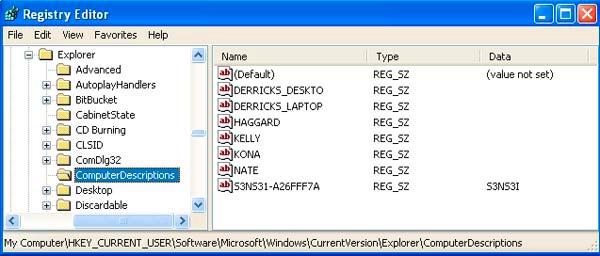


Figure 5 - List of computers associated with on a LAN

# USB Devices

There is sufficient information on this topic to write an entire research paper on, however, for the scope of this paper only the basics will be discussed to show the most relevant Registry keys.

Anytime a device is connected to the Universal Serial Bus (USB), drivers are queried and the device's information is stored into the Registry (i.e., thumb drives). The first important key is HKLM\SYSTEM\ControlSet00x\Enum\USBSTOR. This key stores the contents of the product and device ID values of any USB device that has ever been connected to the system. Figure below reveals the contents of this key. All of which can be interpreted - there lists an ipod, two external hard drives, a digital video camcorder, and several different thumb drives.

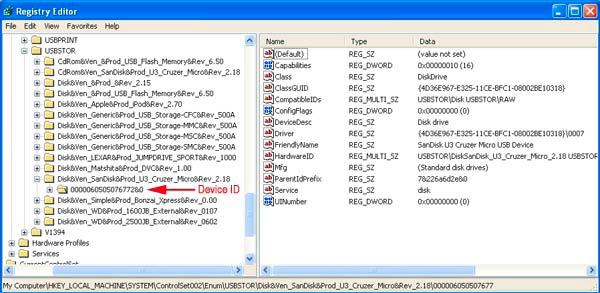


Figure - Contents of USBSTOR key

Beneath each device is the Device ID, which is also a serial number. The serial numbers of these devices are a unique value assigned by the manufacturer, much like the MAC address of a network interface card. Therefore, a particular USB device can be identified to determine whether or not it has been connected to other Windows systems. Carvey mentions in his article The Windows Registry as a Forensic Resource, an important consideration to keep in mind regarding USB device IDs. Not every thumb drive will have a serial number. Particularly, those that have an '&' symbol for the second character of the device ID. In reference to Figure 6, the Device ID that is pointed out has a serial number. However, if the '0' was an '&' that would indicate to an examiner that the device doesn't have a designative serial number. An example of a device that doesn't have an assigned serial number can be seen in Figure below, a Western Digital 250GB external hard drive.

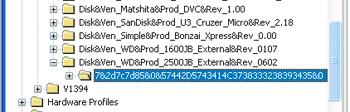


Figure USB device without a designated Device ID

Knowing what USB devices have been connected to a system can assist an examiner in collecting additional evidence that may be crucial to the investigation.

# Mounted Devices

There is a key in the Registry that makes it possible to view each drive associated with the system. The key is HKLM\SYSTEM\MountedDevices and it stores a database of mounted volumes that is used by the NTFS file system. The binary data for each \DosDevices\x: value contains information for identifying each volume. This is demonstrated in Figure below, where \DosDevice\F: is a mounted volume and listed as 'STORAGE Removable Media'.

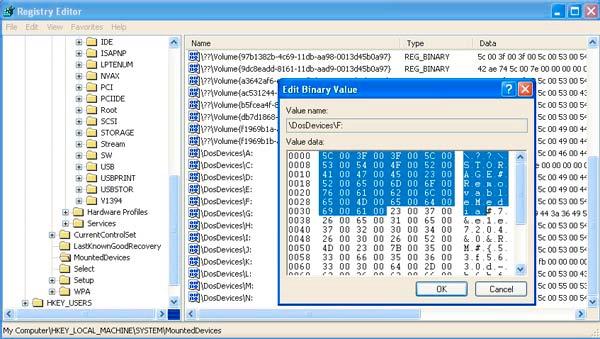


Figure - Identification of volume \DosDevice\F:

This information can be useful to a digital forensics examiner as it shows the hardware devices that should be connected to the system. Therefore, if a device is shown in the list of MountedDevices and that device isn't physically in the system, it may indicate that the user removed the drive in attempt to conceal the evidence. In this case, the examiner would know they have additional evidence that needs to be seized.

# Internet Explorer

Internet Explorer is the native web browser in Windows operating systems. It utilizes the Registry extensively in storage of data, like many applications discussed thus far. Internet Explorer stores its data in the HKCU\Software\Microsoft\Internet Explorer key. There are three subkeys within the Internet Explorer key that are most important to the forensic examiner. The first is HKCU\Software\Microsoft\ Internet Explorer\Main. This key stores the user's settings in Internet Explorer. It contains information like search bars, start page, form settings, etc. The second and most important key to a forensic examiner is HKCU\Software\Microsoft\ Internet Explorer\TypedURLs. Figure below demonstrates the content of what the TypedURLs key displays.

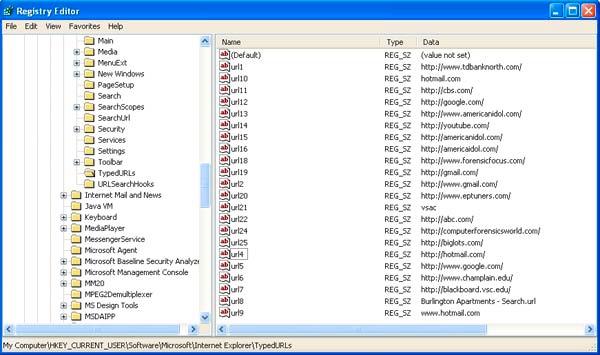


Figure TypedURLs key

From this data an examiner could conclude that the user possibly has a gmail and hotmail email address, they engage in online banking at tdbanknorth, is interested in digital forensic websites, and that they perhaps go to college at Champlain and have been researching apartments in the area.

The third subkey that may interest an examiner is HKCU\Software\Microsoft\ Internet Explorer\Download Directory. This key reveals the last directory used to store a downloaded file from Internet Explorer, giving the examiner an idea as to the location of where the user stores their files.

# Why RegRipper?

RegRipper is a flexible open source tool that can facilitate registry analysis with ease. It contains pre-written Perl scripts for the purpose of fetching frequently needed information during an investigation involving a Windows box.

We are using RegRipper because of the simplicity of the tool and the availability of numerous plugins that capture specific information from the registry.

# Where are these Windows registry hive files located?

Before we start our analysis, it is important that you are familiar with the locations of the Windows registry hive files. In this case, we have analyzed registry hives from a Windows XP box. However, we are mentioning the location of registry hive files both, on Windows XP box, and a Windows 7 box. On a Window XP system, you can find them at:

* WINDOWS/system32/config/software
* /WINDOWS/system32/config/system
* /WINDOWS/system32/config/SAM
* /WINDOWS/system32/config/SECURITY
* /Documents and Settings/Mr. Evil/NTUSER.DAT

On Windows 7 box, you can find them at:

* /Users/intellikid/NTUSER.DAT
* /Windows/System32/config/SOFTWARE
* /Windows/System32/config/SAM
* /Windows/System32/config/SECURITY
* /Windows/System32/config/SYSTEM

Highlighted portion depends on the user profile.

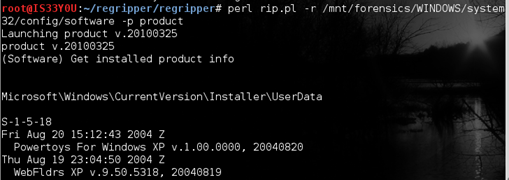
**Beginning Windows Registry Forensics with RegRipper**

Now, we can begin analyzing the registry hives located in the dd image that we have just mounted. We will explore specific registry keys for information one at a time using relevant RegRipper plugins.

**Determining installed product information**

To get information about the Operating System installed on this computer, we use the ‘product’ plugin as follows:

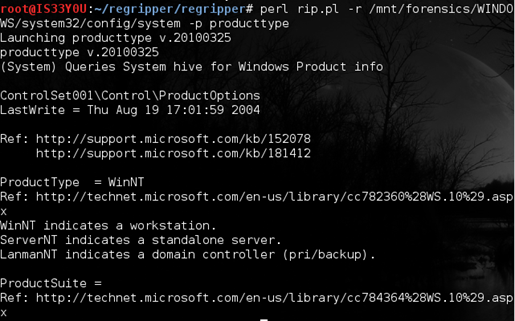
perl rip.pl -r /mnt/forensics/WINDOWS/system32/config/software -p product



**Determining the product type**

The suspect’s Windows box could have been a workstation, a standalone server or a domain controller. To determine this information, we use the ‘producttype’ plugin as follows:

perl rip.pl -r /mnt/forensics/WINDOWS/system32/config/system -p producttype

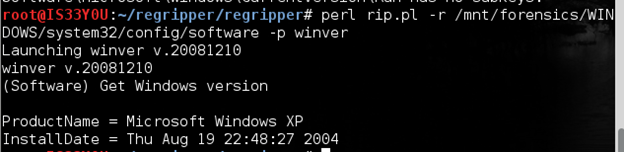


The result indicates that this was a WinNT workstation

**Determining the Windows version**

In order to determine the version of Windows installed on this computer, we use the ‘winver’ plugin:

perl rip.pl -r /mnt/forensics/WINDOWS/system32/config/software -p winver



Notice that this computer has Microsoft Window XP installed and the date of installation is Aug 19, 2004

**Determining the network cards used**

Here, we pull information regarding the network cards that were in use on this computer using the ‘networkcards’ plugin. The command used for this purpose is:

perl rip.pl -r /mnt/forensics/WINDOWS/system32/config/software -p networkcards

The result shows the two network cards that were in use over the Ethernet (wired) and the wireless interface. It also indicates the date and time when these cards were last used

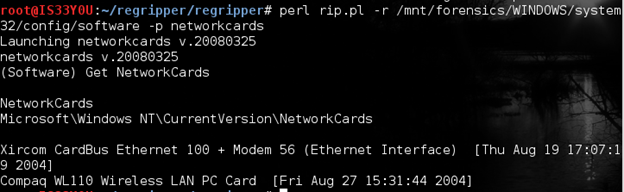


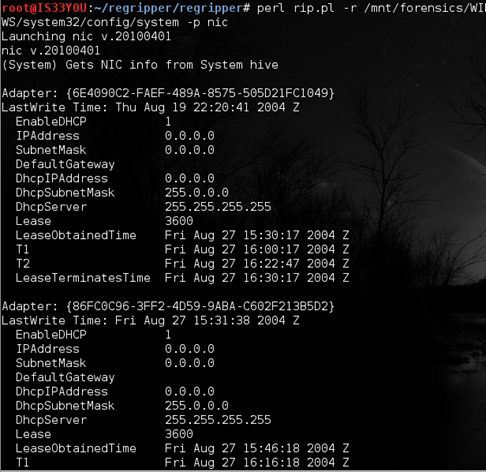
Figure 8

**Note:** During experimentation, it is best that you do not run RegRipper on a live registry hive file.

**Determining the DHCP information**

DHCP is the Dynamic Host Control Protocol that is responsible for allocating IP addresses to computers on a network. We use the ‘nic’ plugin to get the DHCP information from the Network Interface Card, as follows:

perl rip.pl -r /mnt/forensics/WINDOWS/system32/config/system -p nic

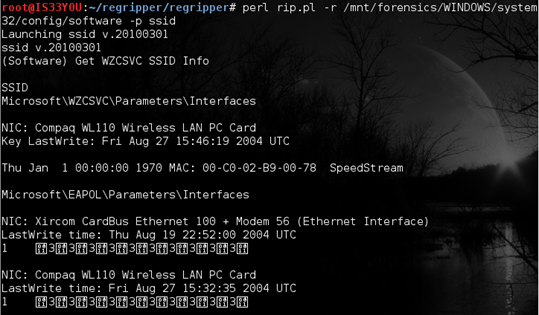


Notice that DCHP is enabled on both the network adapters and time when the lease was obtained from the DHCP server. Also notice that lease lasted for 3600 seconds or 60 minutes (expires after one hour).

**Determining the wireless access points information**

Since this is a wireless hacking case , we are interested in determining the SSIDs that this computer authenticated to. We can pull this information using the ‘ssid’ plugin as follows:

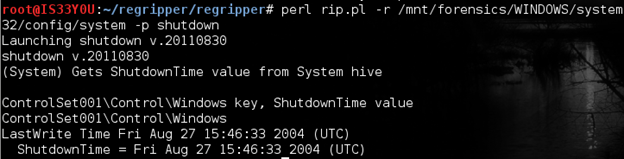
perl rip.pl -r /mnt/forensics/WINDOWS/system32/config/software -p ssid



**Determining the shutdown time**

To get the time this computer was last shutdown, we use the ‘shutdown’ plugin as follows:

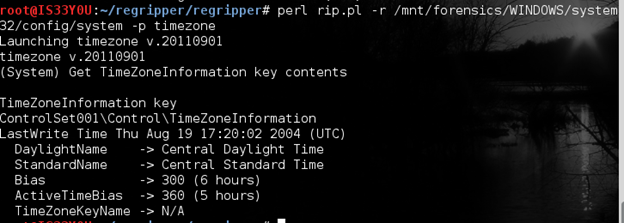
perl rip.pl -r /mnt/forensics/WINDOWS/system32/config/system -p shutdown



**Determining the time zone**

To get information about the time zone that this computer was in, we use the ‘timezone’ plugin as follows:

perl rip.pl -r /mnt/forensics/WINDOWS/system32/config/system -p timezone



We notice that this computer was in the Central Daylight Time or CDT.

**Determining all installed applications**

To get information about the applications that are installed on this computer, the ‘uninstall’ plugin loops through the keys stored under ‘uninstall’ in the registry. We obtain this information as follows:

perl rip.pl -r /mnt/forensics/WINDOWS/system32/config/software -p uninstall

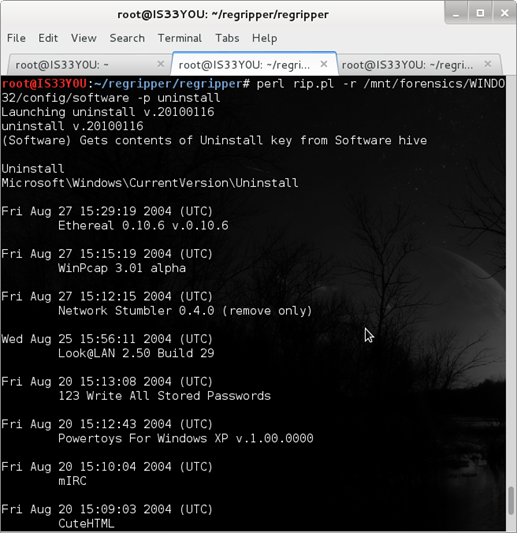


Figure 13

Notice that this plugin sorts the list of installed applications according to date and time (latest first). Also notice the various hacking tools installed on the suspect’s computer such as ‘Ethereal’, ‘Network Stumbler’, ‘WinPcap’, etc [Figure 13].

**Determining User SIDs**

The ‘ProfileList‘ key in the registry can be used to resolve SIDs to users on the machine. If a user has logged onto this machine, a subkey with that user’s SID as its name will be created under ProfileList. Use the ‘profilelist’ plugin in RegRipper to pull this information:

perl rip.pl -r /mnt/forensics/WINDOWS/system32/config/software -p profilelist

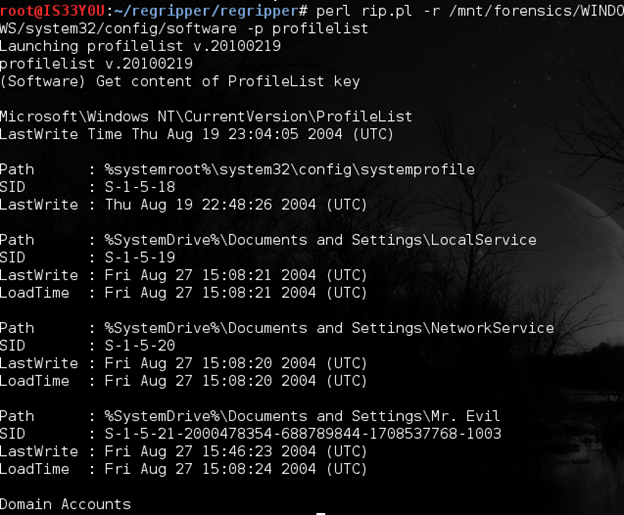


Figure 14

**Determining the recent documents used**

While performing a forensics investigation, recent documents used on the suspect’s computer are of special interest to the investigator. To take a look at what recent documents the suspect has used, we use the ‘recentdocs’ plugin:

perl rip.pl -r /mnt/forensics/Documents\ and\ Settings/Mr.\ Evil/NTUSER.DAT -p recentdocs

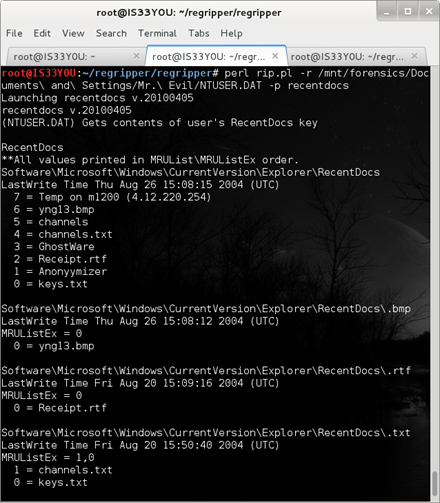


Figure 15

**Note:** By default, Windows stores 15 items in the My Recent Documents option.

**Extracting information from the ‘winlogon’ key**

Winlogon is a valuable key that can help in determining autostart information for the machine. This key is responsible for controlling the events that occur after you logon to a Windows machine. The ‘winlogon’ plugin is a comprehensive plugin that is capable of extracting information from the winlogon key, as follows:

perl rip.pl -r /mnt/forensics/WINDOWS/system32/config/software -p winlogon

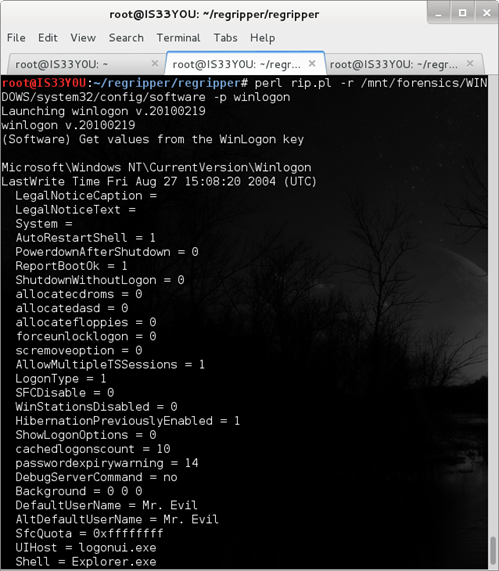


Figure 16

**Determining suspect’s web-browsing history**

The ‘TypedURLs‘ key in the registry holds information about the web addresses that this specific user has previously typed in the browser. The purpose of this key is to populate the drop down menu pertaining to URLs in order to improve user experience. We use the ‘typedurls’ plugin in RegRipper to access the information stored in this key, as follows:

perl rip.pl -r /mnt/forensics/Documents\ and\ Settings/Mr.\ Evil/NTUSER.DAT -p typedurls

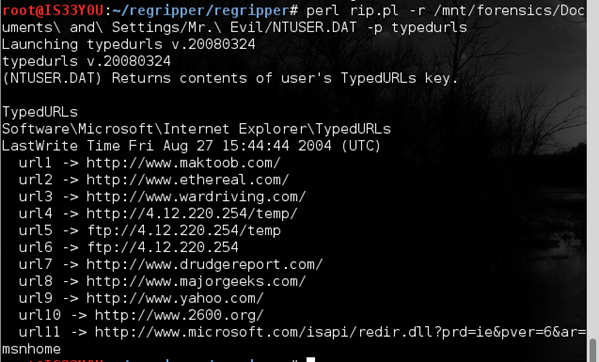


Figure 17

Notice that the suspect has been visiting websites related to ‘hacking’ tools. [Figure 17]

**Collecting information about unread emails**

Windows registry stores information about the unread emails of the outlook user. We use the ‘unreadmail’ plugin to extract this information:

perl rip.pl -r /mnt/forensics/Documents\ and\ Settings/Mr.\ Evil/NTUSER.DAT -p unreadmail

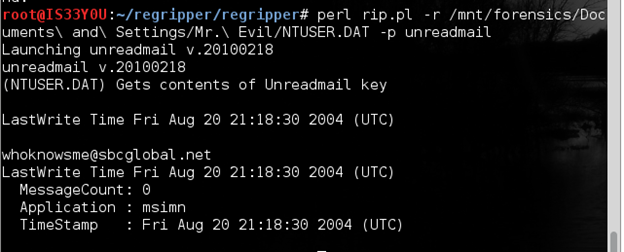


Figure 18

Notice that the registered email account of the suspect is ‘whoknowsme@sbcglobal.net’. There are no unread emails [Figure 18].

**Determining applications set to auto start**

The ‘Run’ key in the NTUSER.DAT file contains the locations of the programs that are set to autostart once this specific user logs into the machine. We capture the contents of the ‘Run’ key using the plugin ‘user\_run’:

perl rip.pl -r /mnt/forensics/Documents\ and\ Settings/Mr.\ Evil/NTUSER.DAT -p user\_run

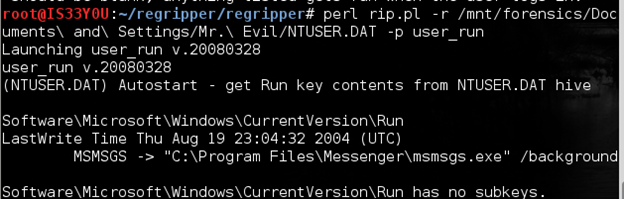


Figure 19

**Determining the value of the ‘userinit’ registry key**

The registry key ‘UserInit‘ is used to indicate what applications can be launched once a user logs onto the computer. The default application stored in this key is: ‘C:\WINDOWS\system32\userinit.exe‘. We use the plugin ‘userinit’ to pull information from this key:

perl rip.pl -r /mnt/forensics/WINDOWS/system32/config/software -p userinit

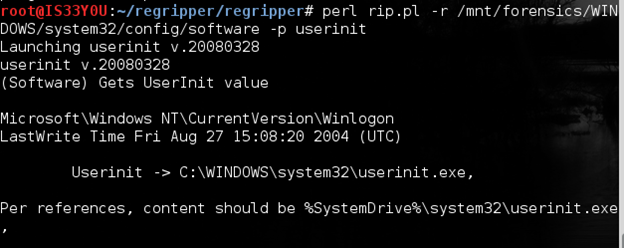


Figure 20

**Determining the user’s printers**

Printers are of interest during investigations since they may have been used to print sensitive documents that can help shed further light on the case. The investigator determines the printers used by the suspect on this computer using the ‘printers’ plugin:

perl rip.pl -r /mnt/forensics/Documents\ and\ Settings/Mr.\ Evil/NTUSER.DAT -p printers

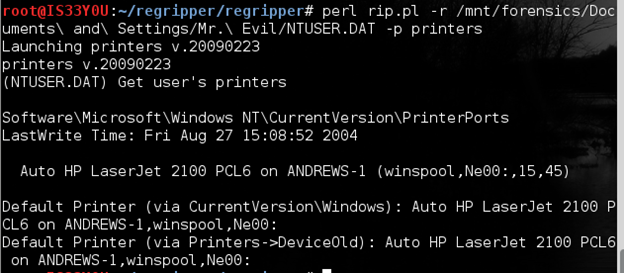


Figure 21

Notice that the default printer used on this device is an ‘Auto HP LaserJet 2100 PCL6’ and the last write time was Aug 27, 2004 at 15:08:52 [Figure 21].

**Collecting information about ‘Cain & Able’**

We have noticed the presence of Cain & Able sniffing and password cracking utility installed on this system. We use the ‘cain’ plugin to pull more information about this tool:

perl rip.pl -r /mnt/forensics/Documents\ and\ Settings/Mr.\ Evil/NTUSER.DAT -p cain

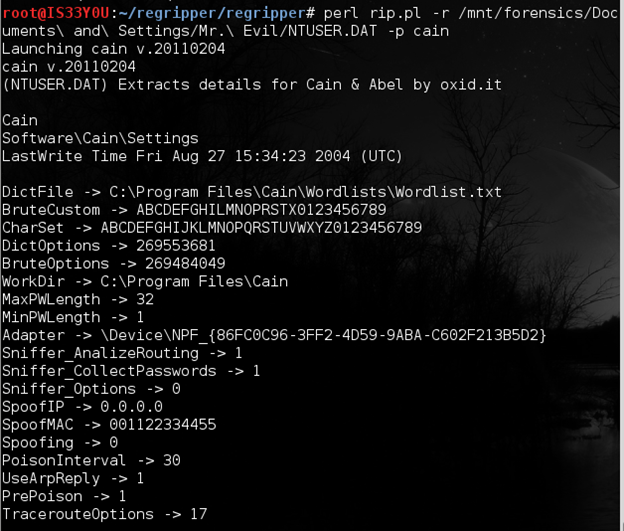


Figure 22

Notice the location of the Dictionary file set as: ‘C:\Program Files\Cain\Wordlists\Wordlist.txt’ [Figure 22]. Also notice other settings like the Sniffer is set to collect passwords and analyze routing protocols. The spoofed MAC (hardware) address of the computer is set to be ‘001122334455‘ and the password character length is set to range from 1 to 32. The working directory is set as ‘C:\Program Files\Cain\‘.

# Installing xmount

# apt-get install xmount

# Installing parse::win32Registry

# perl –MCPAN –e shell

# install Parse::Win32Registry

# Converting ewf file to dd image using xmount

# xmount –in ewf /media/sf\_Downloads/4Dell\ Latitude\ CPi.E?? /mnt/xmount

# Obtain the partition details of image

# fdisk–l /mnt/xmount/4Dell\ Latitude\ CPi.dd

# Mounting the dd image

# mount –t ntfs –o ro,offset=32256 /mnt/xmount/4Dell\ Latitude\ CPi.dd /mnt/Forensics