Windows Forensic Analysis

Dedicated to incident response and computer forensic analysis topics, with respect to Windows 2000, XP, 2003, and Vista operating systems

WFA: Locating MAC addresses in a Windows Image

It’s not unusual to the see the question of locating media access control (MAC) addresses within an acquired image of a Windows system. The MAC address[1] is commonly known as being a unique identifying number “burned into” a network interface card (NIC). As such, when a forensic analyst is incorporating network traffic captures from within a subnet into her investigation, she may need to determine the MAC address used by a NIC installed in a particular system. The question then becomes, does the operating system make use of the MAC address in such a way that leaves discernable and identifiable artifacts that can be found within the system image?

The first 6 bytes of a MAC address constitute the “organizationally unique identifier” (OUI[2]) that can be used to identify the manufacturer of the NIC. A list of manufacturers to which OUIs are assigned is maintained[3] by the IEEE.

The short answer is…maybe. By default, the Windows NT family of operating systems (specifically, Windows 2000, XP, 2003, and Vista) does not maintain MAC addresses within the Registry. However, drivers for some NICs or even perhaps application software may maintain or record MAC addresses in the Registry, or in other files. Further, it may be possible to locate the MAC address within other files found within an acquired image of a Windows system.

Live Response

When performing live response, obtaining the MAC address(es) from a Windows system is relatively trivial, using a command such as `ipconfig /all`. The MAC address appears in the output of the command as “Physical Address” for each interface in use on the system. Such commands can be included in a batch file, allowing for the automated collection of a variety of information, including the MAC address(es).

You can also use the `net config rdr` command to list the MAC addresses for each of the network interfaces. In the command output, under the heading “Workstation active on”, you will see entries that start with “NetBT_Tcpip_{*GUID*}”, followed by the MAC address in parentheses.

Network Connection Settings

The first place to look for a MAC address on a system is the following Registry key:

HKLM\SYSTEM\CurrentControlSet\
  Control\Class\{4D36E972-E325-11CE-BFC1-08002bE10318}

Within an image, this key would be found in the System Registry hive file, usually located in the path %WinDir%\system32\config\system.

Locating the CurrentControlSet

Within an image, the “CurrentControlSet” key is not visible, as it is volatile (populated by the system at boot), and the analyst will need to mount or access the System Registry

[1] http://en.wikipedia.org/wiki/MAC_address

[2] http://en.wikipedia.org/wiki/Organizationally_Unique_Identifier

[3] http://standards.ieee.org/regauth/oui/oui.txt

hive file within the forensic analysis application being used (i.e., ProDiscover, EnCase, etc.) and locate the SYSTEM\Select Registry key. The "Current" value identifies which ControlSet00*n* was loaded as the CurrentControlSet when the system is booted.

This key contains driver-specific settings for each network interface, as seen through the network connection properties (via the Control Panel). Each of the subkeys beneath this key (i.e., "0000", "0001", etc.) contains a "DriverDesc" value that corresponds to the adapter name found in the Description field in the output of `ipconfig /all`. One of the values found within this key may be named "NetworkAddress". In the case of the Dell Wireless 1390 WLAN Mini-Card properties settings on one of my Windows XP systems, there is a value named "Locally Administered MAC Address" which appears in the network interface properties. This setting does not have a value by default, but when I choose to set a value, and set that value to "DEADBEEFCAFE" and click "OK" (closing the dialog and setting the value), that string appears in the Registry subkey associated with the interface, as the data for a new value named "NetworkAddress".

If the "NetworkAddress" value is located beneath one of the above listed Registry keys, then this may be the result of specific attempts by a user to change or spoof the MAC address used by their NIC. Methods for doing this have been widely circulated on the Internet[4], and tools have been produced to make this Registry modification relatively trivial.

Additional artifacts that you may look for if you suspect that the MAC address has been modified include the existence of MAC address modification tools (see the Resources section at the end of this article for a list of such tools), or indications that they may have been run (i.e., Prefetch files on Windows XP systems, entries in the UserAssist Registry key, etc.).

Network Adapters

Information about network adapters in use on a Windows system is maintained in the following Registry key:

HKLM\SOFTWARE\Microsoft\
  Windows NT\CurrentVersion\
  NetworkCards

Navigating to this key in RegEdit on a live system, you should see subkeys named with various numbers, as illustrated in figure 1.

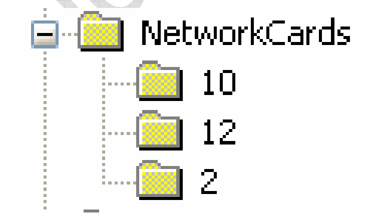
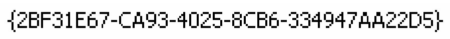


Figure 1

Figure 1 illustrates the contents of the NetworkCards[5] key on a Windows XP SP2 system. Each of the listed subkeys contains the values named "Description" and "ServiceName", and the "ServiceName" value contains data that looks like a globally unique identifier, or GUID, as illustrated in figure 2.



{2BF31E67-CA93-4025-8CB6-334947AA22D5}

Figure 2

With the GUIDs from the numbered network adapter subkeys, you can then navigate to the following key:

HKLM\SYSTEM\CurrentControlSet\

[4] http://www.irongeek.com/i.php?page=security/changemac

[5] http://support.microsoft.com/kb/146333

Services\Tcpip\Parameters\Interfaces

This key contains a list of subkeys, each of which is the GUID that references a particular interface or adapter. For each of the listed GUIDs from the NetworkCards key, the Interfaces subkey with the same name will contain the values that apply to various settings, such as `EnableDHCP`, `IPAddress`, etc.

What you should *not* see within these Registry keys is a value named “NetworkAddress”. By default, the Windows OS does not maintain the MAC address for an interface within the Registry keys that apply to that interface or adapter.

A final Registry location you can check is the following key:

HKLM\SOFTWARE\Microsoft\Windows
Genuine Advantage

This key appears to be an artifact of the installation of Window Genuine Advantage validation on Windows XP. While not found on a consistent basis during testing, some instances of Windows XP may have this key, and may have a string value named “MAC” beneath this key. The “MAC” value may contain a semi-colon-delimited list of MAC addresses in use on the system.

Shortcut Files
Another potential location for MAC address artifacts is Windows shortcut (*.lnk) files. There is very little official documentation published by MS regarding the binary format of Windows shortcut files, but Jesse Hager produced a fairly thorough description of the format through reverse engineering the binary format[6]. Fortunately, tools such as the Windows File Analyzer (WFA) from MiTec[7] make parsing Windows shortcut files a relatively simple task. If a directory within an image (such as the Recent folder within a user profile) contains several Windows shortcut files, you can either mount the image as a read-only drive letter on your analysis system using Mount Image Pro[8] or VDKWin[9], or export all or a number of the files from the image. WFA will automatically parse the binary contents of the shortcut files within the selected directory, showing you the MAC address, as illustrated in figure 3.

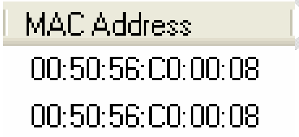

MAC Address
00:50:56:C0:00:08
00:50:56:C0:00:08

Figure 3

Note
The MAC address illustrated in figure 3 is the MAC address assigned to the “VMWare Virtual Ethernet Adapter for VMnet8”. As yet, I have been unable to locate any credible documentation that states how this part of a Windows shortcut file is populated.

Summary
By default, Windows does not maintain MAC addresses for NICs within the Registry. This, of course, does not take into account specific drivers that are written to record the MAC address. However, a forensic analyst may be able to locate artifacts within an acquired image of a Windows system that may provide contain the MAC address for a NIC that was used on the live system.

Check the following three Registry keys for values named “NetworkAddress”:

[6] The location of Jesse’s PDF file can vary, so search on Google for “Jesse Hager” + “Windows shortcut format”

[7] http://www.mitec.cz/wfa.html
[8] http://www.mountimage.com/
[9] http://petruska.stardock.net/software/

HKLM\SYSTEM\CurrentControlSet\
Control\Class\{4D36E972-E325-11CE-BFC1-08002bE10318}*nnnn*

HKLM\SYSTEM\CurrentControlSet\
Services\Tcpip\Parameters\Interfaces\
{GUID}

Look for a value name "MAC" within the following Registry key:

HKLM\SOFTWARE\Microsoft\Windows
Genuine Advantage

Also be sure to check Windows shortcut files for MAC addresses.

Resources

- Lance Mueller's EnScript[10] for retrieving the MAC address from Windows shortcut files
- TMAC[11] MAC address changer from Technitium
- MAC Makeup[12] MAC address changer
- MadMACs[13] MAC address changer from IronGeek
- SMAC[14] MAC address changer
- EtherChange[15] from NTSecurity.nu

Harlan is a forensic analyst located in the metro DC area, and is the author of three books related to live response and forensic analysis of Windows systems. He also presents at conferences and provides training on those same topics, as well. He can be reached at keydet89@yahoo.com.

[10] http://www.forensickb.com/2007/07/obtaining-mac-address-of-machine-from.html

[11] http://tmac.technitium.com/tmac/index.html

[12] http://www.gorlani.com/publicprj/macmakeup/macmakeup.asp

[13] http://www.irongeek.com/i.php?page=security/madmacs-mac-spoofer

[14] http://www.klcconsulting.net/smac/

[15] http://ntsecurity.nu/toolbox/etherchange/