SANS DFIR Cheat Sheet

HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer

- \ComDlg32
  - \LastVistedPidlMRU
  - \OpenSavePidlMRU
- \RecentDocs
- \RunMRU
- \TypedPaths
- \UserAssist

HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\RunOnce

HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\RunOnce

HKCU\SOFTWARE\Microsoft\Windows\Shell < Shellbags

- \BagMRU
- \Bags

HKCU\SOFTWARE\Classes

- **Insert %USERPROFILE%\AppData\Local\Microsoft\Windows\UsrClass.dat**

HKLM\SYSTEM\*CurrentControlSet*\Enum\USBSTOR < Class ID / Serial #
HKLM\SYSTEM\*CurrentControlSet*\Enum\USB < VID / PID

HKLM\SOFTWARE\Microsoft\Windows Portable Devices\Devices

- Find Serial # and then look for *FriendlyName* to obtain the **Volume Name** of the USB device

HKLM\SYSTEM\MountedDevices

- Find Serial # to obtain the **Drive Letter** of the USB device
- Find Serial # to obtain the **Volume GUID** of the USB device

HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\EMDMgmt

- Key will ONLY be present if system drive is NOT SSD
- Traditionally used for ReadyBoost
- Find Serial # to obtain the **Volume Serial Number** of the USB device
  - The Volume Serial Number will be in decimal – convert to hex
  - You can find complete history of Volume Serial Numbers here, even if the device has been formatted multiple times. The USB device's Serial # will appear multiple times, each with a different Volume Serial Number generated on each format.

Using the **Volume GUID** found in **SYSTEM\MountedDevices**, you can find the **user** that actually mounted the USB device:

NTUSER.DAT\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Mountpoints2

USB Times:

- First time device is connected
- Last time device is connected
- Removal time

HKLM\SYSTEM\*CurrentControlSet*\Enum\USBSTOR\*Ven_Prod_VersionUSB iSerial #*\Properties\{83da6326-97a6-4088-9453-a1923f573b29}\####**

- **0064** = First Install (Win7 / 8)
  - Also found in setupapi.log / setupapi.dev.log
- **0066** = Last Connected (Win8+ only)
  - Also \Enum\USB\VID_XXXX&PID_YYYY last write time of USB Serial # key
  - Also \MountPoints2\{GUID} last write time of key
- **0067** = Last Removal (Win8+ only)

USB First Time Device Connected Logs:

XP:	C:\Windows\setupapi.log
Vista+:	C:\Windows\inf\setupapi.dev.log

Search for the device's Serial # within these logs and you can discover the first time a device was plugged in to a computer.

USBDeviceForensics is an application by WoanWare that can help automate all of these things.

Miscellaneous Info:

HKLM\SYSTEM\*CurrentControlSet*\Control\TimeZoneInformation
HKLM\SYSTEM\*CurrentControlSet*\Control\ComputerName\ComputerName

---

HKLM\SYSTEM\*CurrentControlSet*\services\LanmanServer\Shares

- Display all open shares on a system

HKLM\SYSTEM\*CurrentControlSet*\Control\FileSystem

- Look for **NtfsDisableLastAccessUpdate**, which is set to 0x1 by default, which means that access time stamps are turned OFF by default

HKLM\SYSTEM\*CurrentControlSet*\services\Tcpip\Parameters\Interfaces

- Display interfaces and their associated IP address configuration (record the interface GUID!)

Network Location Awareness (NLA) was included in Vista+, and aggregates the network information for a PC and generates a GUID to identify each network (a "network profile", if you will). The Windows Firewall uses that information to apply firewall rules to the appropriate profile. You can find evidence of every network a machine has connected to using NLA registry keys.

Check the last write time of a key to determine the last time a PC connected to a particular network.

HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\NetworkList

- \Signatures
  - \Unmanaged (record **DefaultGatewayMac**, **DnsSuffix**, **FirstNetwork** (SSID), **ProfileGuid**)
  - \Managed
- \Nla
  - \Cache
- Profiles

Most info regarding NLA will be stored under the **NetworkList** key above, and also:
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\HomeGroup

Network Type, and First / Last Connected Times (find using the **ProfileGuid** key harvested from Signatures\Unmanaged):

HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\NetworkList\Profiles\{GUID}

HKLM\SOFTWARE\Microsoft\WZCSVC\Parameters\Interfaces\{GUID} → (XP only, use last write time of the key to determine the last time the network was connected)

0x06	=	Wired
0x17	=	Broadband
0x47	=	Wireless

You will also find **DateCreated** and **DateLastConnected** under this key. It's 128-bit Windows System Time, and is stored in UTC.

LNK File Analysis:

C:\*username*\AppData\Roaming\Microsoft\Windows\Recent
**Use TZWorks lp.exe utility!*
Jump Lists (like LNK files on steroids):

C:\*username*\AppData\Roaming\Microsoft\Windows\Recent\AutomaticDestinations
C:\*username*\AppData\Roaming\Microsoft\Windows\Recent\CustomDestinations

**Use TZWorks jmp.exe utility!*
…remember, LNK files are actually embedded in the database structure in AutomaticDestinations

---

Prefetcher and SuperFetch:

- Prefetcher and SuperFetch are part of Windows' memory manager
- **Prefetcher** is the less capable version included in **Windows XP**
- Prefetcher was extended by **SuperFetch** and **ReadyBoost** in **Windows Vista+**
- **ReadyBoot** replaces Prefetcher for the boot process if > 700MB RAM
- Tries to make sure often-accessed data can be read from the fast RAM instead of slow HDD
- Can speed up boot and shorten amount of time to start programs

C:\Windows\Prefetch

filename-hash(xxxxxxxx).pf
Example: CALC.EXE-AC08706A.pf
The hash is a hash of the file's path. In this example, CALC.EXE is located in C:\Windows\System32. If it were copied to another location (like the Desktop) and executed, a new .pf file would be created reflecting a hash of the new path.

HKLM\SYSTEM\CurrentControlSet\Control\Session Manager\Memory Management\PrefetchParameters

EnablePrefetcher Key:

0 = Disabled
1 = Application prefetching enabled
2 = Boot prefetching enabled (default on Windows 2003 only)
3 = Application and Boot prefetching enabled (default)

- Task Scheduler calls Windows Disk Defragmenter every three (3) days
- When idle, lists of files and directories referenced during boot process and application startups is processed
- The processed result is stored in **Layout.ini** in the Prefetch directory, and is subsequently passed to the Disk Defragmenter, instructing it to re-order those files into sequential positions on the physical hard drive