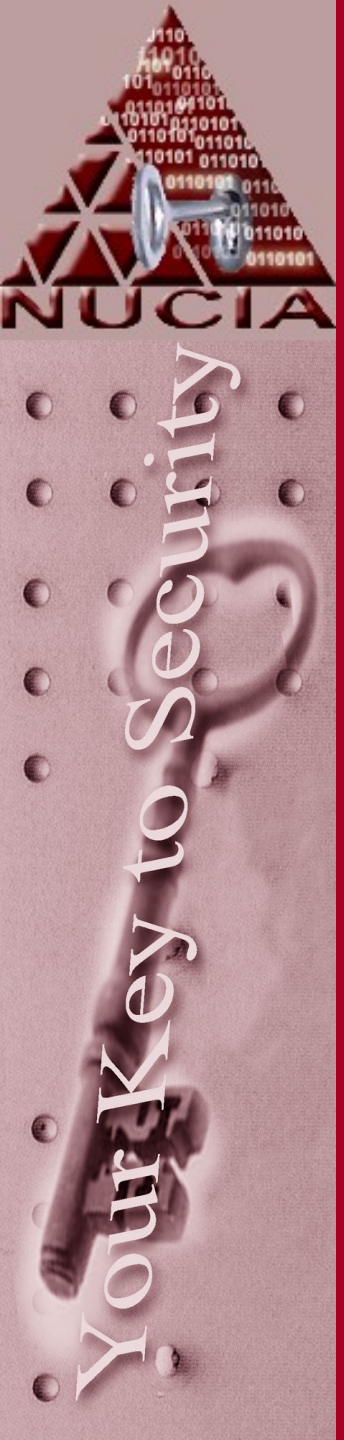


Cyber-Forensics Intermediate Topics

CERTConf2006

Tim Vidas

Analysis is an Art

Who am I?

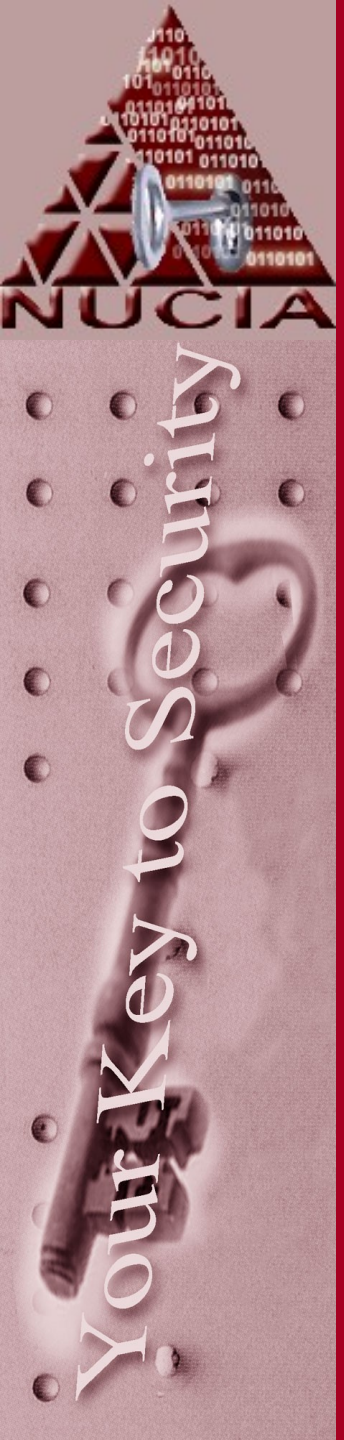
- Tim Vidas
  - Sr. Tech. Research Fellow
  - UNO/PKI/NUCIA
  - Certs: CISSP, 40xx, Guidance, AccessData etc.
  - Instructor: UNO, Guidance, LM RRCF

NUCIA

- Nebraska University Consortium on Information Assurance
- IA full time
- Traditional university coursework in IA, Crypto, Forensics, Secure Administration, Certification and Accreditation, etc
- STEAL Labs
- “Other work”
- Most of us are ‘around’ CERTconf.

Who are you?

- Who are you?
- Where do you work?
- What do you do?
- How many of you are planning on attending all “Forensics” sessions?
- What are you expecting to get out of them? (I’ll try to be accommodating)



Disclaimer

- Even though this class touches on quite a few legal topics – nothing should be construed as advice or legal instruction
- Before performing many of the skills learned this week on a computer other than your own, you may need to seek permission (possibly written) and or seek advice from your own legal counsel.

Analysis

Generic

Mounting an Acquired Image

- Mount read only
  - Software write blocker
  - Sort of like write protect tab
  - Example:

mount –t vfat –o ro,loop,noexec,noatime image directory

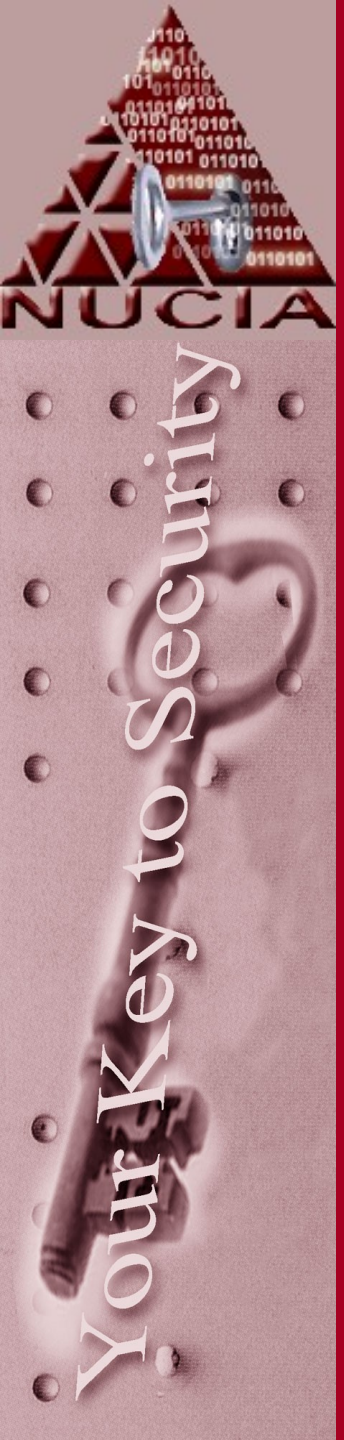
-t type (also can auto-guess)

-o options

ro	read only
loop	loopback device – we're really mounting a file
noexec	don't execute anything from this mount point
noatime	don't attempt to update MAC times

Flags for Linux mount Command

- **-a**
  - All the filesystems described in **fstab(5)** are mounted.
- **-d**
  - Causes everything to be done except for the actual system call.
  - This option is useful in conjunction with the **-v** flag to determine what the **mount** command is trying to do.
- **-f**
  - Forces the revocation of write access when trying to downgrade a filesystem mount status from read-write to read-only.



Flags for Linux mount Command

- **-o**
  - Options are specified with a **-o** flag followed by a comma separated string of options.
  - The following options are available:
    - async
      - All I/O to the file system should be done asynchronously. This is a *dangerous* flag to set, and should not be used unless you are prepared to recreate the file system should your system crash.
    - nodev
      - Do not interpret character or block special devices on the file system. This option is useful for a server that has file systems containing special devices for architectures other than its own.

Mounting Read-Only in Linux

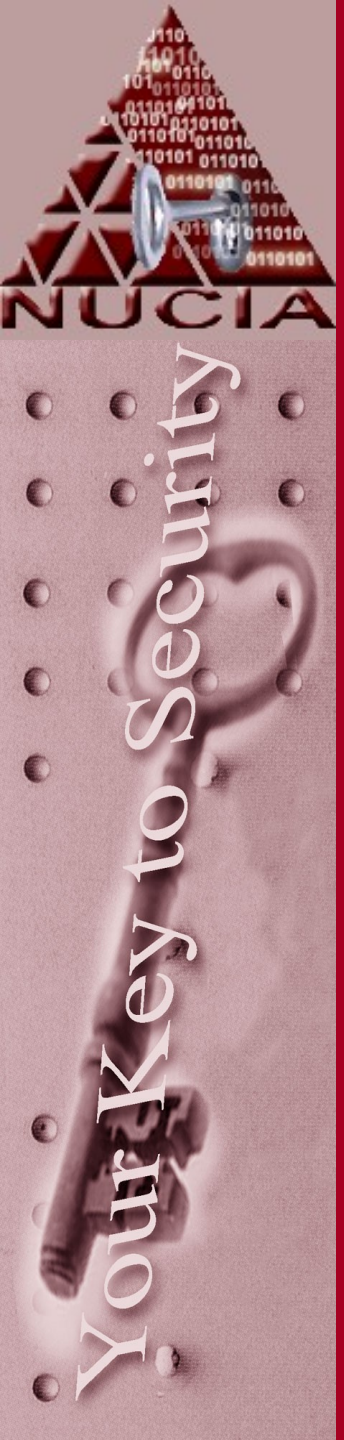
- **-o**
  - Options are specified with a **-o** flag followed by a comma sepa- rated string of options.
  - The following options are available:
    - noexec
      - Do not allow execution of any binaries on the mounted file system. This option is useful for a server that has file systems containing binaries for architectures other than its own. nosuid Do not allow set-user-identifier or set-group-identifier bits to take effect.
    - rdonly
      - The same as **-r**; mount the file system read-only (even the super-user may not write it). sync All I/O to the file system should be done synchronously. update The same as **-u**; indicate that the status of an already mounted file system should be changed.
    - loop
      - Treat the image as a file system

Mount

- **-r**
  - The file system is to be mounted read-only.
  - Mount the file system read-only (even the super-user may not write it).
  - The same as the ``rdonly'' argument to the **-o** option.
- **-t** *ufs* | *lfs* | *external type*
  - The argument following the **-t** is used to indicate the file system type. The type *ufs* is the default.
  - The *-t* option can be used to indicate that the actions should only be taken on filesystems of the specified type.
  - More than one type may be specified in a comma separated list.

When you're done

- If something can be mounted, can it be unmounted?
- Yes, but the command is actually: **umount** (NOT unmount)
  - # mount –t linux-ext3 –o ro,loop floppy.dd /mnt/evidence
  - # umount /mnt/evidence
- If you mounted an actual device either directory works:
  - # mount /dev/sda1 /mnt/usb
  - # umount /dev/sda1 OR #umount /mnt/usb



Slight diversion

- A GREAT way to hide stuff…
  - Create a directory
    - mkdir /mnt/hide
  - Copy ‘stuff’ to /mnt/hide
    - cp badstuff.* /mnt/hide
  - ‘ls’ to make sure documents are there
    - ls /mnt/hide
  - Next mount an image on that directory
    - mount –t vfat image.dd /mnt/hide
  - ‘ls’ again

Slight diversion…

- You should see the contents of the mounted drive
  - Where did the evidence go?
  - Still there, but there is a file mounted over it.
  - It still physically exists on the HD
- As an investigator how do I tell if someone is hiding information this way?
  - Several ways
  - Execute mount command by itselft to determine if anything looks suspicious
  - Umount the directories and see if anything physically resides in them.

Strings

- The sleuthkit
- Underneath Autopsy
- *Strings* looks for ASCII strings in a binary file or standard input. *Strings* is useful for identifying random object files and many other things. A string is any sequence of 4 (the default) or more printable character followed by whitespace or null.
- Windows versions available, some tools performs strings analysis “for free”

Strings

- Options:
  - f report filename
  - d report in decimal
  - x report in hex
  - n # where # is a number, instead of using the default of 4, specify string length

grep

- **Grep** searches the named input ***files*** (or standard input if no files are named, or the file name **-** is given) for lines containing a match to the given ***pattern***. By default, **grep** prints the matching lines.
- Options:
  - H show filename
  - e “string” where string is a search term
  - There are TONS more
- Example:
  - grep –He “bomb” ./*
- Note: careful when using STDIN – you may not get filenames for example

Together?

- strings –f ./* > stringfile.dat
- cat stringfile.dat | grep “bomb”
- Now I just said be careful when using with cat? WTF?
  - Why is it ok here?
- Why would we want to do it that way and not:
  - strings –f ./* | grep “bomb”

    or just
  - grep –e “bomb” ./*

Analysis

Micro$oft Centric

Legend

- If the item only pertains to Windows 9x it will appear in green
- If the item only pertains to Windows 2000 / XP it will appear in blue
- Documents and Settings is abbreviated at DaS

Internet Explorer Remnants

- IE uses a caching system to make frequent visits to the same pages quicker
- Remnants can be found in
  - \windows\Temporary Internet Files
  - \DaS\[username]\Local Settings\Temporary Internet Files

Internet Explorer Remnants

- …also ‘lives’ in the Temp Internet Directories:

\temporary internet files\Content.IE5\[XXXXXXXX]

The last directory is a series of ASCII characters (sim to 6YQ2GSWF)

There may be many of these, as the user surfs content is divided into each directory

These folders plus a directory for cookies exist in the Temp Internet Files directory

Internet Remnants

- History files
  - \Windows\History
  - \DaS\[username]\Local Settings\History
- Favorites
  - \Windows\Favorites
  - \DaS\[username]\Local Settings\Favorites
- Cookies
  - \Windows\Cookies
  - \DaS\[username]\Local Settings\Cookies

Cookies

- Each cookie is typically it's own file
- Cookies are different for each site but could contain a wealth of information

History

- The amount of history stored is user-defined – the hex value may be found at:
- Local_Machine\Software\Microsoft\Windows\Current\InternetHistory\URLHist

History

- \history\history\IE5\<mshist200206 25…..0724>\index.dat
- Date range for history shows 6.25.2005 to 7.24.2005
- The index.dat is updated after a page is visited and thus added to the cache

Index.dat

- IE hashes webpages into the index.dat (traditional OS flavor “hashing”)
- If the visited pages hash value doesn’t exist in index.dat, then the page must be retrieved from the internet

Index.dat

- IE 4+
  - Cached files:
    - ..\Temporary Internet Files\Content.IE5\index.dat
  - History
    - ..\History\index.dat
  - Cookies
    - ..\Cookies\index.dat
- IE 3
  - MM256.dat - web addresses lest that 256 chars
  - MM2048.dat –web address from 257-2048 chars long

Typed URLs

- IE 5+ keeps track of urls you've typed, in order to attempt to predict the web address as you type it
- These URLs live at:
  - Hkey_current_user/software/Microsoft/Internet Explorer/TypedURLs

AutoComplete

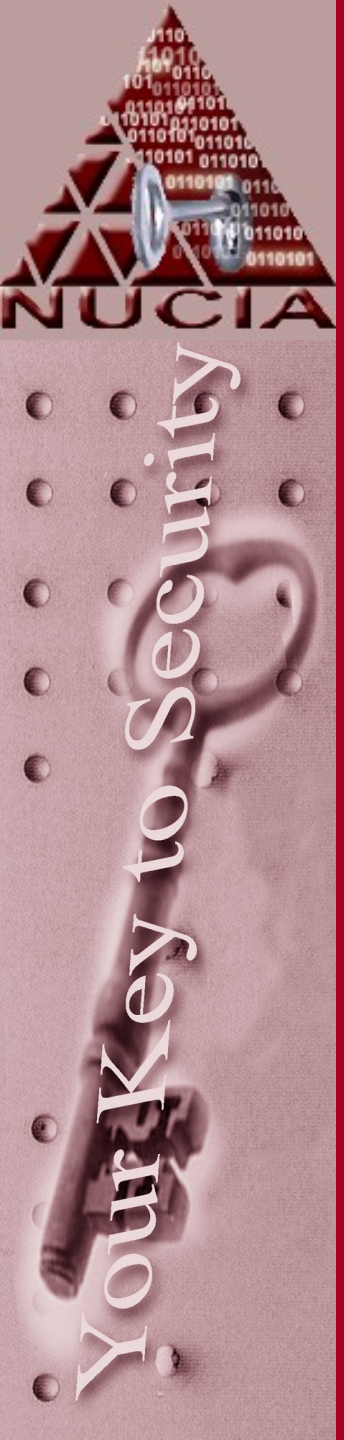
- If a user has enabled form-field completion in IE, their passwords can be recovered…
- Extract these reg keys:
  - Curr_users\Software\Microsoft\InternetExplorer\Intelliforms
  - Local_Machine\Software\Microsoft\Protected Storage System Provider
- Note the URLs of the sites you want to obtain the password for
- Create a new user on the investigation machine matching the users User-id
- Import the above keys in the new registry hive
- Booting the investigation machine and browsing to the URL will fill the appropriate username and password fields

AutoComplete

- Who can thing of another way to recover auto complete information?
- Hint “less forensicy”

Win NT

- In WinNT 4.x the location for profile information is:
  - \winnt\Profiles\[username]

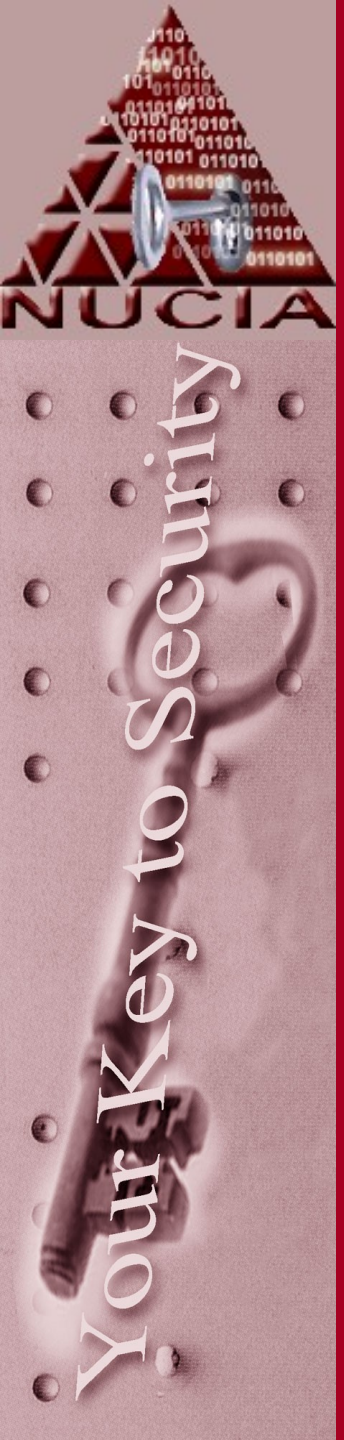


Other “good” registry keys

- Dial-up Accounts:
  - HKEY_CURRENT_USER\RemoteAccess\Addresses
- Dial-up Account Usernames:
  - HKEY_CURRENT_USER\RemoteAccess\Profile\[isp_name]
- RegisteredOwner/Organization, Version, VersionNumber, ProductKey, ProductID, ProductName
  - HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion
- MSN Messenger Info:
  - HKEY_CURRENT_USER\Identities\{string}\Software\Microsoft\MessengerService
  - HKEY_CURRENT_USER\Software\Microsoft\MessengerService
- MS NetMeeting Information:
  - HKEY_CURRENT_USER\Software\Microsoft\User Location Service\Client

Other “good” registry keys

- **McAfee user registration info:**
  - LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\ElectronicCommerce\UserInfo
  - LOCAL_MACHINE\Software\Network AssociatesECare\UserInfo
- **Outlook Express User Info (e-mail, newsgroups, etc):**
  - HKEY_LOCAL_MACHINE\Software\Microsoft\Internet Account Manager\Accounts
  - HKEY_LOCAL_MACHINE\Software\Microsoft\Internet Account Manager\Accounts\0000000x
- **Yahoo! Messenger User Info (last username logged in):**
  - HKEY_CURRENT_USER\Software\Yahoo\Pager
- **Yahoo! Account Info (possible other accounts)**
  - HKEY_CURRENT_USER\Software\Yahoo\Pager\Aim\UserSerring
- **Instant Messenger Screen Name**
  - HKEY_CURRENT_USER\Software\America Online\AOL Instant Messenger(TM)\CurrentVersion\Login



Other “good” registry keys

- **Instant Messenger Users**
  - CURRENT_USER\Software\America Online\AOL Instant Messenger (TM)\CurrentVersion\Users.
- **Internet Explorer History settings length**
  - HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Internet Settings\URLHistory
- **View Recent Network Shares**
  - HKEY_CURRENT_USER\Network\Recent
  - HKEY_USERS\.Default\Network\Recent
- **Network Information**
  - HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\VXD\DHCP\DhcpInfo00
- **YAHOO! Chat – (ID’S CHATTING TOGETHER)**
  - CURRENT_USER\Software\Yahoo\Pager\Profiles\ <ID> \ Imvironment\Recent

Other “good” registry keys

- **DHCP information :**
  - DHCP IP Address (hex)
  - DHCP Server (hex)
  - Subnet Mask
  - Hardware Address (MAC Address)

  HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\VXD\MSTCP

- **Information entered in TCP/IP Properties, DNS Configuration:**
  - Host Name
  - Domain Name
  - DNS Server IP Addresses

  HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Class\NetTrans\0000

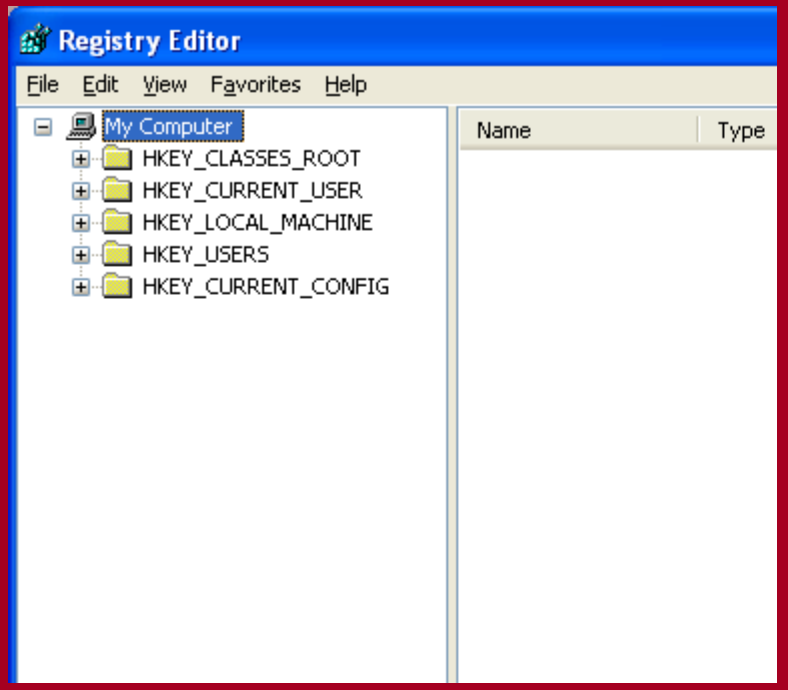
- **Information entered in TCP/IP Properties, IP Address:**
  - Static IP Address
  - Subnet Mask
  - Typed URLs (Internet Explorer)

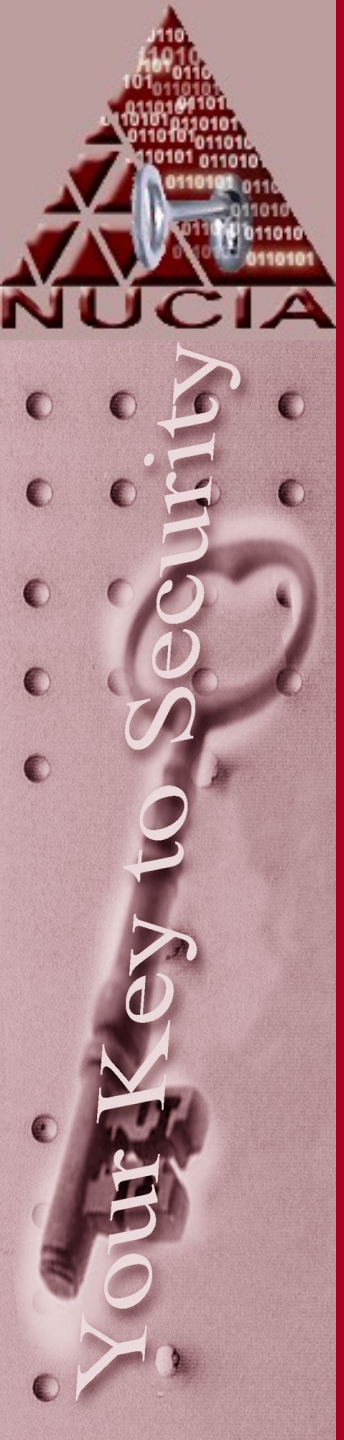
  HKEY_CURRENT_USER/Software/Microsoft/Internet Explorer/TypedURLs

- **Other information like nameservers and domains.**
  - HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\VxD\MSTCP

The Registry

- Bob McCoy is running around the conference, see if you can coerce him into giving you a copy of his “registry slides” they are a great explanation of how the registry works assuming little/no prior knowledge





Print Spooler

- Since win 9x printed documents are actually spooled before any data is actually sent to the printer
- This spooling process created files temporarily that contain information about what is to be sent to the printer
- Typically this is done in either RAW or EMF (enhanced metafile format)

Print Spooler

- SPL and SHD files are created for each printing process
  - SHD is a “shadow file” with metadata about the print job…owner, printer, print method
  - SPL
    - RAW – the actual data sent to the printer
    - EMF – Name of the file, data
      - Each printed page is going to have one EMF contained inside of the SPL

Print Spooler

- On a typical workstation:
  - \windows\system32\spool\printers
  - \windows\temp
  - Note: in windows 9x, the method is a little different. The spool files point to .tmp files where the filename starts with EMF. One tmp file per page.
    - Ie ~EMFXXXX.TMP where XXXX are ascii chars

Print Spooler

- EMF files contain essentially thumbnails of what was printed
- Simply extract the preceding 41 bytes before the string “EMF” in an SPL file using a hex editor.

Print spooler

- File Headers
  - 01 00 00 00 is the header for EMF
  - Further bytes can help you decode what version of Windows the EMF was created on:
  - 01 00 00 00 58 00 - win 9x
  - 01 00 00 00 58 6e - win 9x
  - 01 00 00 00 18 17 - win 2000
  - 01 00 00 00 d8 18 - win 2000
  - 01 00 00 00 c4 36 - win 2000
  - 01 00 00 00 5a 01 - win xp
  - 01 00 00 00 5c 01 - win xp

Print Spooler

- Keep in mind, that once the print has “succeeded” the files are deleted
- Irfanview, Encase and other viewers can view EMF files

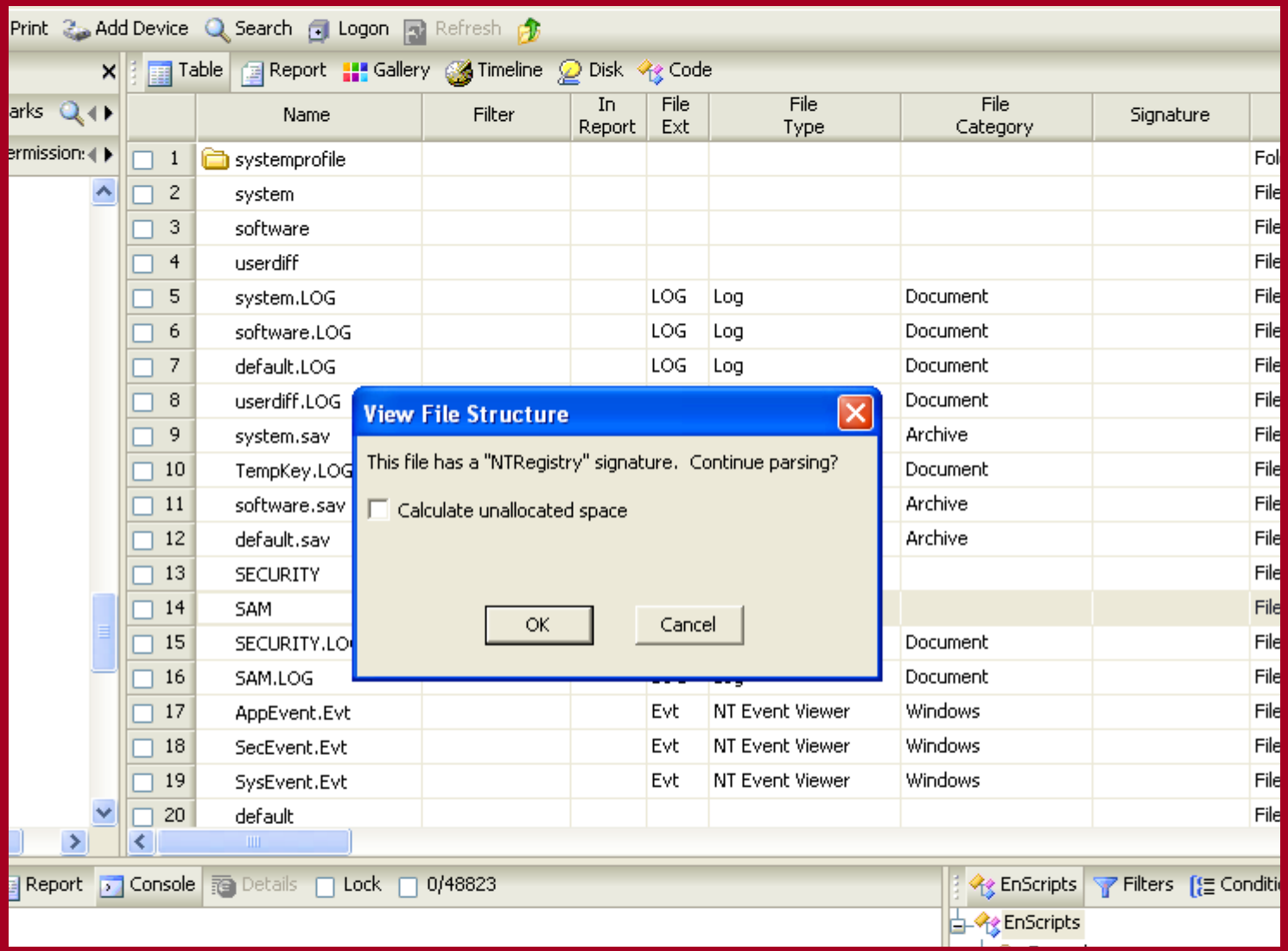
The SAM file

- Security Accounts Manager
- In a typical windows domain, this is going to be on the PDC, or in %systemroot%\system32\config\sam for local accounts
- The SAM is locked by system (admins have read) so changes are made through the OS or a pre-boot environment

The SAM

- The sam can be dumped using a utilities like samdump, or pwdump
- Once dumped there are a variety of tools used to parse the sam, not the least popular is L0phtCrack.
- Encase can view a sam natively, with “view file structure”

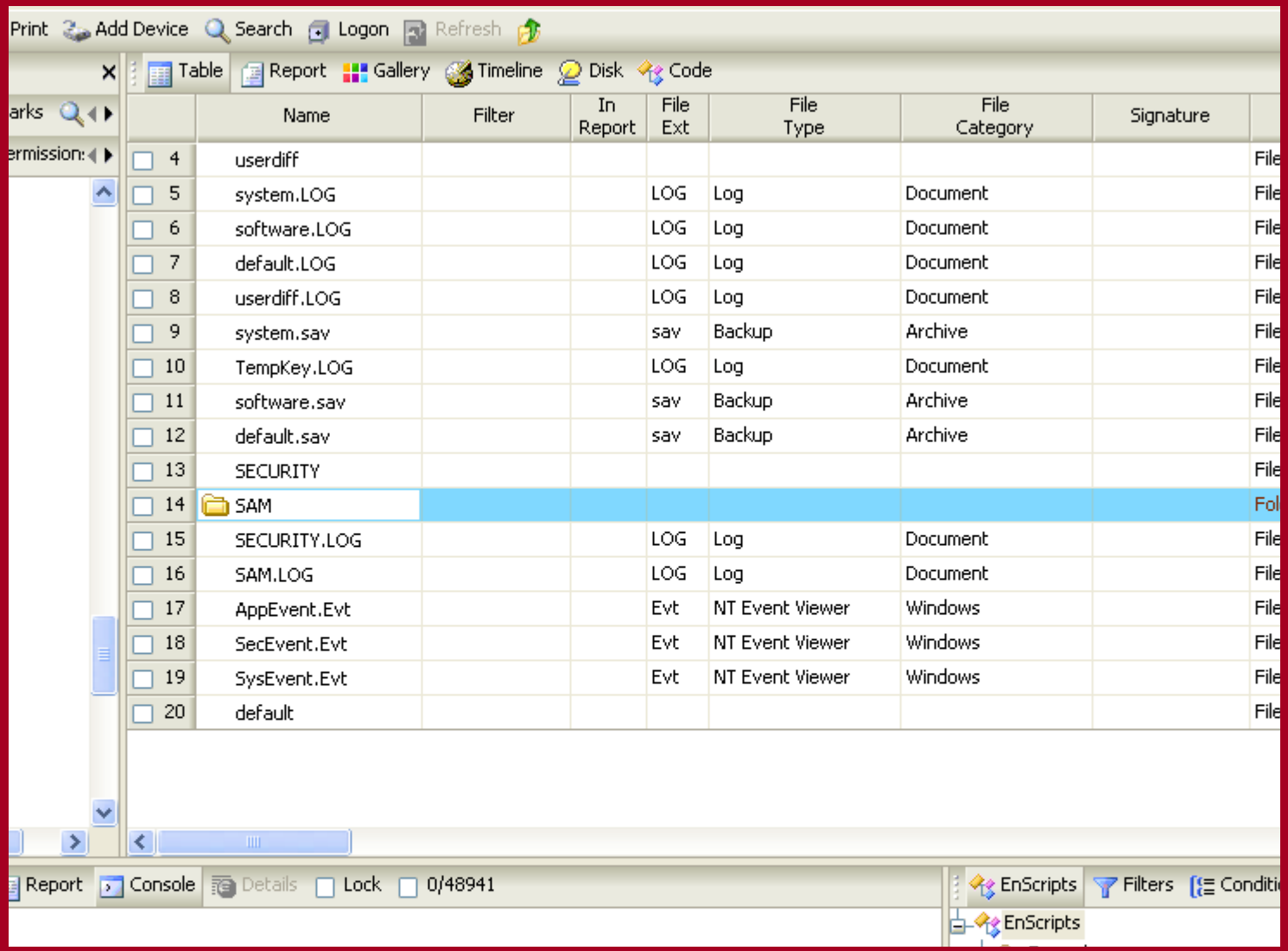
The SAM



NUCIA

Your Key to Security

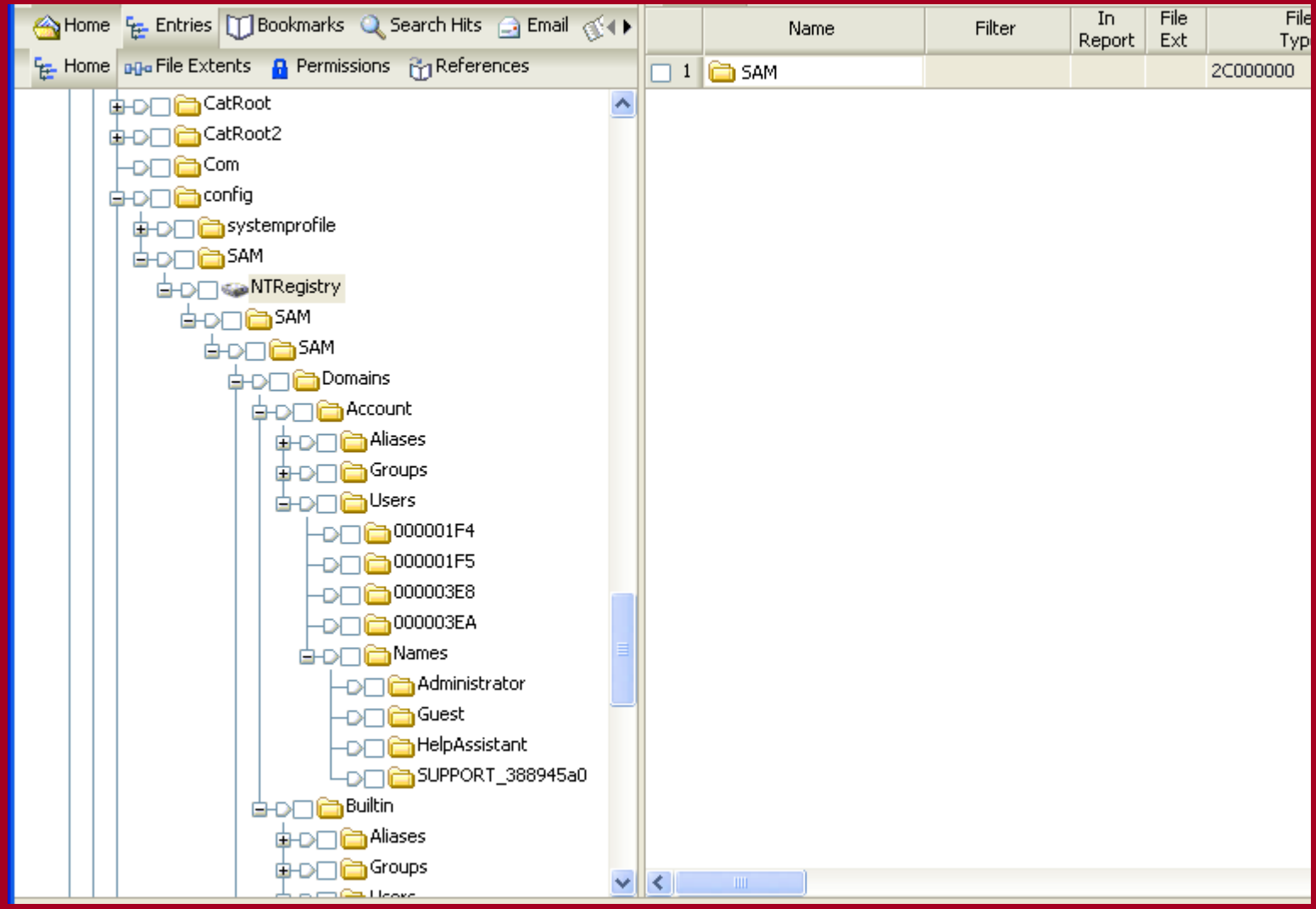
The SAM



NUCIA

Your Key to Security

The SAM



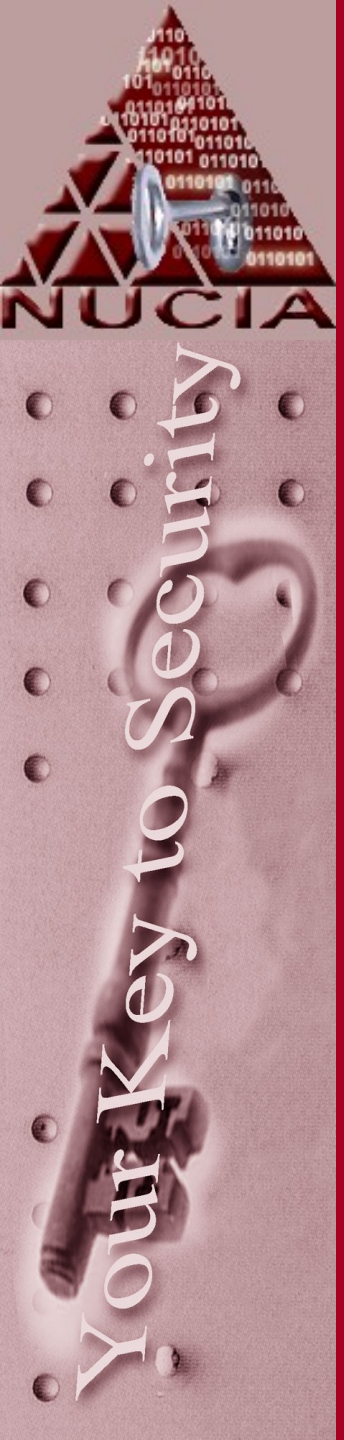
Your Key to Security

The Recycle Bin

- Recycle bin exists for each volume on each drive
- Removable media has no bin
- First time a file is deleted a new directory is created in the \Recycler directory on that volume the string contains the SID and RID of the user:
  - Eg S-1-5-21-17155567821-1682526488-682003330-500

The Recycle Bin

- S-1-5-21-1715567821-1682526488-682003330-500
- 1 – Version 1=NT, 2K, XP
- 5 – Auth: 1 everyone, 5 specific user
- 21 – part of Domain DID
- 500, last 3 or 4 #s are UID,
  500= admin, 1000+ are users
- Rest of the #'s are the rest of the DID
- The SID also appears in other places in BINARY form…like access_token data structures



The Recycle Bin

- This new directory has two files:
  - Desktop.ini
    - Contains class ID for directory – tells explorer how to display the directory
    - (SOFTWARE\classes\clsid\{XXXX})
  - INFO2
    - Each 800 byte (280 byte) record contains information for a single deleted file

The Recycle Bin

- INFO2
  - Begins with:
    - 4 bytes: Header
    - 4 bytes: allocated records
    - 4 bytes: total records
    - 4 bytes: record size (800/280)
    - 4 bytes: total logical file size

The Recycle Bin

- INFO2
  - Since there were 20 bytes already used, the first record starts at byte 21:
    - 260 bytes: path and original file name
    - 4 bytes: index #
    - 4 bytes: Drive #
    - 8 bytes: date/time deleted
    - 4 bytes: physical size
    - 520 bytes: path and original file name (UNI)

C:\Documents\doc.doc | 01 00 00 00 | 02 00 00 00 | timestamp|….

The Recycle Bin

- When docs are placed in the Recycle Bin, MFT changes occur:
  - Modified / Last Accessed updated
  - Long filename deleted
  - Short filename changed to Dc1.doc or sim:
    - D – deleted
    - C – original volume letter
    - 1 – bin index number
    - .doc – original extension

The Recycle Bin

- Emptying the recycle bin:
  - Resets the first 20 bytes
  - MFT entry for the files are marked as deleted – Modified / accessed, etc are unchanged
  - Index number is NOT changed, the next deleted file index number is sequential
    - TMV note: this seems to depend upon reboot – at least for XP

The Recycle Bin

- Deleting a file from the recycle bin:
  - changes the first byte of the record to 00
  - Change the allocation flag in the MFT to 00 00 (unallocated)

The Recycle Bin

- Remember the SAM?
  - 3 or 4 digit UID is saved in the same in lilendian….so user 1004 is EC030000 in hex
  - In the user ‘directory’ 000030CE locate EC030000 and extract the 18 preceeding bytes. This is a 32 bit integer of the SID for that user.

Dump files

- Dr Watson creates a user.dmp file when a user level program crashes. It's basically a RAM dump.
- Examine using \support\debug\i386\dumpexam.exe or windbg –z user.dump
  - Ei dumpexam –y <symbol file> <dumpfile>
  - This will parse memory information to \windows\MEMORY.txt

MS Office

- Microsoft applications, to say it lightly, generally have a lot of meta data in them
- Documents can contain their own timestamps, creators, editors, etc etc

MS Office

- By default a file is named after the first sentence typed.
- Typically users will alter this name to something more meaningful to them.
- If the default file name found in the data area does not match the first sentence of the document data, the first sentence was altered since creation…

MS Office 97

- Files created in office 97 were tagged with a GUID (Global Unique identifier) this GUID was actually the MAC address of the machine the file was first created on.

MS Office

- rhdtool.exe is an available tool from Microsoft that allows you to save a copy of an office file with all the ‘sensitive’ hidden data removed.

Other “Directories of interest”

- Application Data for a user
- Desktop for users and all users
- My Documents
- Local Settings
- Recent – recently accessed items
- Sendto – remnants of previously attached drives
- Start Menu
- Program Files

Other “Files of Interest”

- Swap file – typically called pagefile.sys and located in the %systemroot% - this is basically virtual RAM
- Thumbs.db – date information and thumbnail data for images

References

- Stephenson, P. (2001). *Investigating Computer-Related Crime*. CRC Press.

Web Sites

- www.lastbit.com
- http://www.garykessler.net/library/fi
- http://www.microsoft.com/downloa

Resources

- Man pages…standard on many distributions or on the web