Chapter 33 – Forensics, Auditing, Logging

Wajih
University of Illinois
ECE 422/CS 461 – Fall 2019

Whoami

- 5th year Ph.D. student working with Prof. Bates
- Research Interests:
  - Threat Detection
  - Forensic Investigation

Digital Forensics

- Forensics –*the use of science or technology to discover evidence for a court of law*
- Digital forensics – forensics relating to digital devices
- Usually trying to recreate a chain of events
- Used in both criminal and civil law

Daubert Standard

- Rules for scientific evidence
  - Admissibility of expert witness testimony.
- Judge is the ultimate “gatekeeper”
- Evidence must be relevant and reliable
  - Is the method subject to testing?
  - Are there established error rates?
  - Is it generally accepted by the community?
  - Has the technique been peer reviewed?

Digital Forensics Phases

Collection/Acquisition

Preservation

Analysis

Presentation

Digital Artifacts

Operating system: event logs, registry data

File system: access times, modification times

Disk: deleted files, hidden partitions

Internet: browser history, email

Media: photos, videos, audio

Documents: Office, PDFs, RTF, XML

Databases: MySQL, Oracle

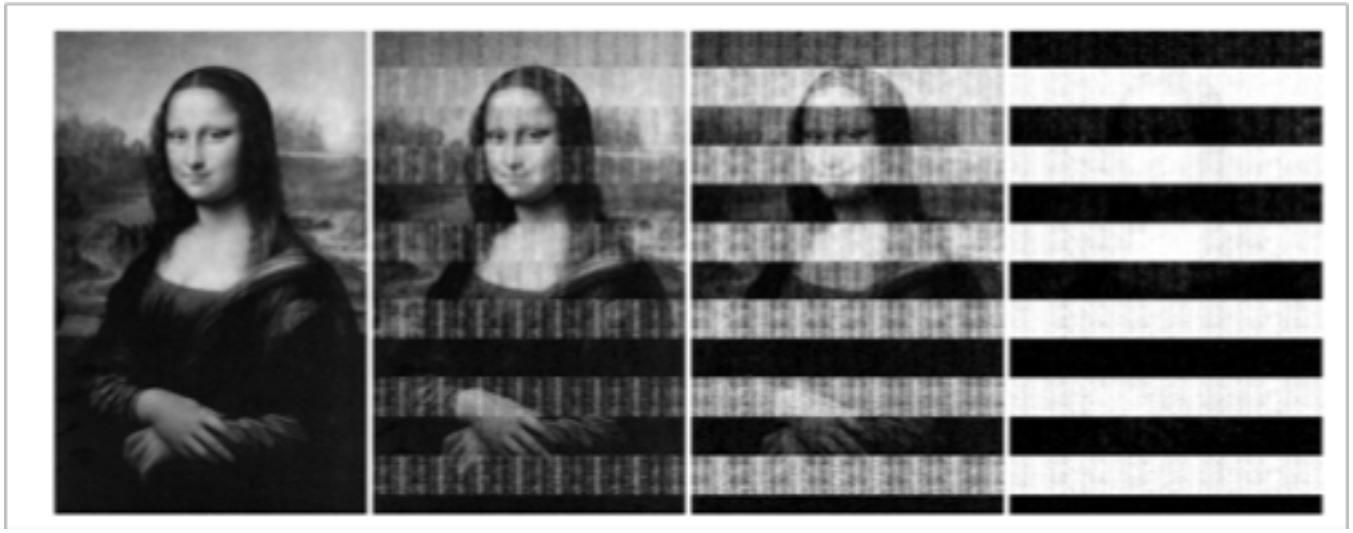
Application data: instant messaging

Preservation Decision - Live Analysis

- When system is running, additional information can be collected
  - active connections
  - active processes
  - loaded kernel modules
  - open files
- Problems
  - analysis modifies state of system –
  - rootkits (or other modifications) can tamper with results

Preservation Decision - Live Analysis

- Physical (volatile) memory is also more persistent than one thinks
  - RAM taken out of computer and scanned for sensitive content



Lest We Remember: Cold Boot Attacks on Encryption Keys Usenix Security 2008

Preservation Decision – Dead Analysis

- Dead analysis – image drives
  - bit-for-bit forensic duplicate using a write blocker
  - analysis can be conducted later
  - identify hidden or deleted files

FILE SYSTEM FORENSICS

Common File Systems

UFS: Unix File System	Ext[2,3,4]: Extended file system	FAT[12,16,32]: File Allocation Table
NTFS: New Technology File System	ReFS: Resilient File System	HFS[+]: Hierarchical File System

Common File Systems

UFS: Unix File System

Ext[2,3,4]: Extended file system

FAT[12,16,32]: File Allocation Table

NTFS: New Technology File System

ReFS: Resilient File System

HFS[+]: Hierarchical File System

NTFS Forensics

- To extract forensically relevant information such as
  - Deleted files
  - Event logs
- NTFS Basics:
  - Everything is a file, even the core file system internals
  - The internal files are always hidden from user view

Hidden Internal Files

Filename	Description
$MFT	Master File Table
$MFTMirr	Backup of first 4 records of MFT
$LogFile	Transaction log file
$Volume	Volume related information, usually empty
$AttrDef	$AttrDef Table listing MFT attribute names and numbers
.	Root folder on NTFS
$Bitmap	Map showing which clusters on volume are in use
$Boot	Boot code used during bootstrap
$BadClus	Map of bad clusters
$Secure	Security descriptors and ACLs are listed here
$Upcase	Keeps all lowercase to uppercase character mappings
$Extend	Optional extensions listed here (This is a folder)

Image taken from https://www.slideshare.net/nullowaspmumbai/ntfs-forensics-66460882

Physical Layout of NTFS Volume

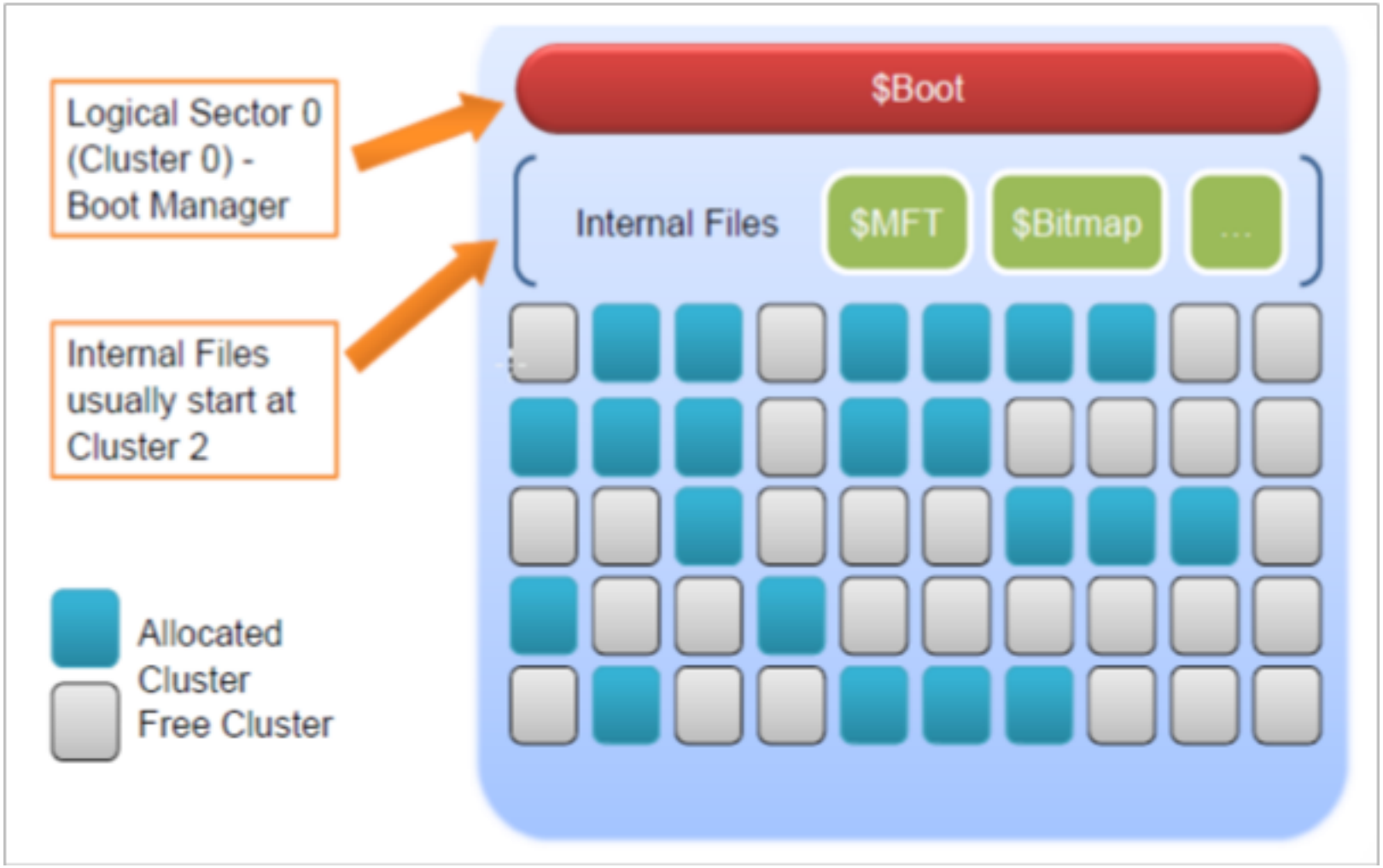


Image taken from https://www.slideshare.net/nullowaspmumbai/ntfs-forensics-66460882

Master Boot Record (MBR)

- Data structure stored on the first sector of the drive.
- Cross-platform industry standard for locating and booting disk partitions
- Used by BIOS/UEFI to boot (start) the OS
- Contains:
  - The partition table describing each partition
  - Boot loader code to fetch and execute a partition

Boot Sector

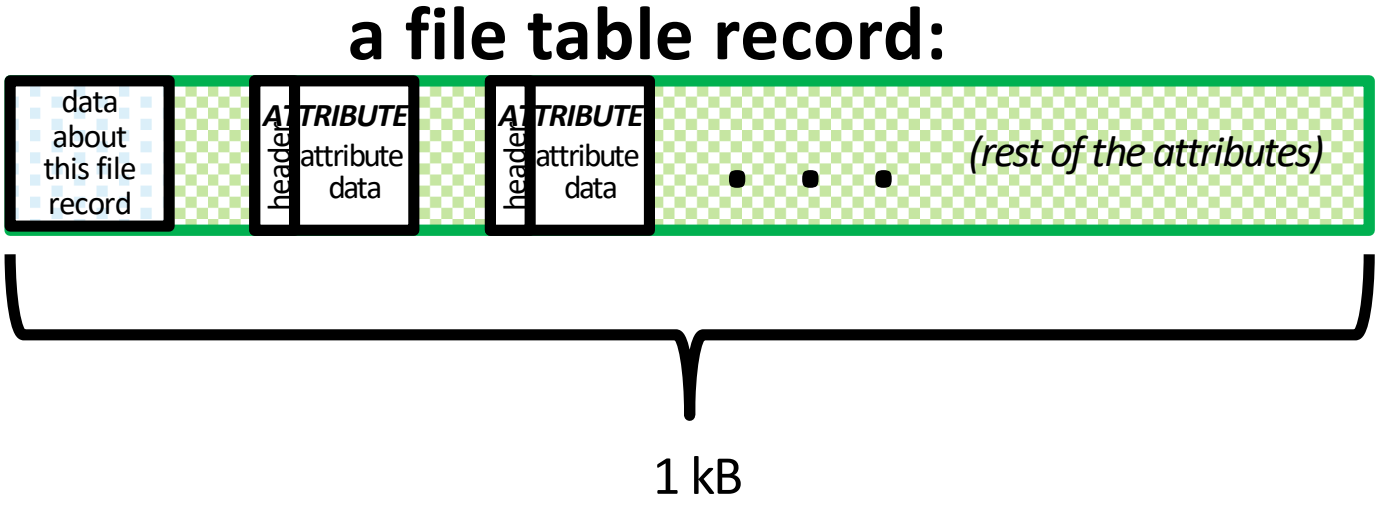
- Jump instruction to bootstrap code
- Data structures describing partition
  - Magic number (for NTFS)
  - Bytes per sector
  - Sectors per cluster
  - Sectors per track
  - Location of master file table ($MFT) and mirror
  - Serial number, checksum
- Bootstrap code (loads operating system)

NTFS: Master File Table

- File/folder records are stored in $MFT
- Records are 1 kb
- Records contain ***attributes*** that define the characteristics of a file, including the data itself

MFT File Record Attributes

- Attributes define the characteristics of a file, including the data
- Resident Attributes vs Non-resident Attributes



File Attributes

- Files in NTFS typically have the following attributes:
  - **$STANDARD_INFORMATION:** Contains MAC times, security ID, Owners ID, permissions in DOS format, and quota data.
  - **$FILE_NAME:** Contains the file name in UNICODE, as well as additional MAC times, and the MFT entry of the parent directory.
  - **$OBJECT_ID:** Identifiers regarding the files original Object ID, its birth Volume ID, and Domain ID.
  - **$DATA:** The raw content data of the file.

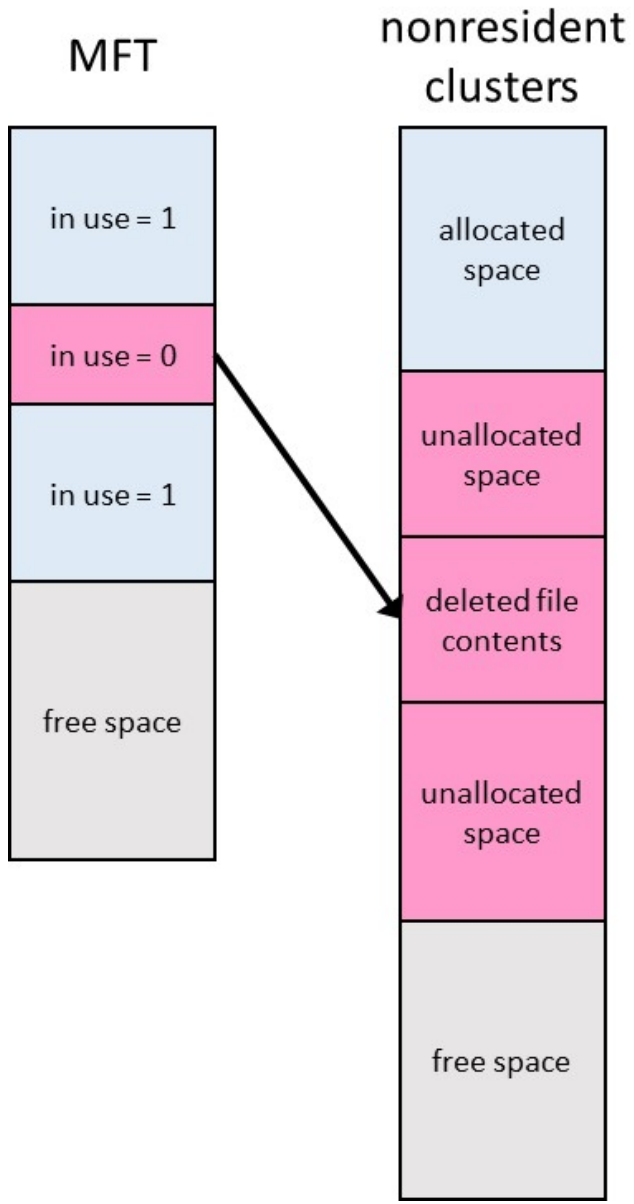
Deleting a File

- When a file is deleted the IN_USE flag is cleared from the MFT entry, but the attribute contents still exist.

- Can be used to recover path of deleted file.

Recovery with File Record

- File record intact.
- File contents intact.



Directories

- Directories in NTFS are indexed to make finding a specific entry in them faster
- They are stored in a B-Tree sorted in alphabetical order
- Some Attributes:
  - **$INDEX_ROOT:** root of the B-Tree.
  - **$INDEX_ALLOCATION:** sub-nodes of the B-Tree.
  - **$BITMAP:** describes which structures in the B-Tree are being used.

FILE CARVING

File Carving

- Modern file systems tend to overwrite metadata for deleted files
- Recovery of files ***without*** file system metadata
- Can be done without ***any*** metadata
- Can carve many different kinds of media

Recovery vs. Carving

- Recovery: file system metadata are intact; use them to find file (“undelete”)
- Carving: pulls the ***raw*** bytes from the media

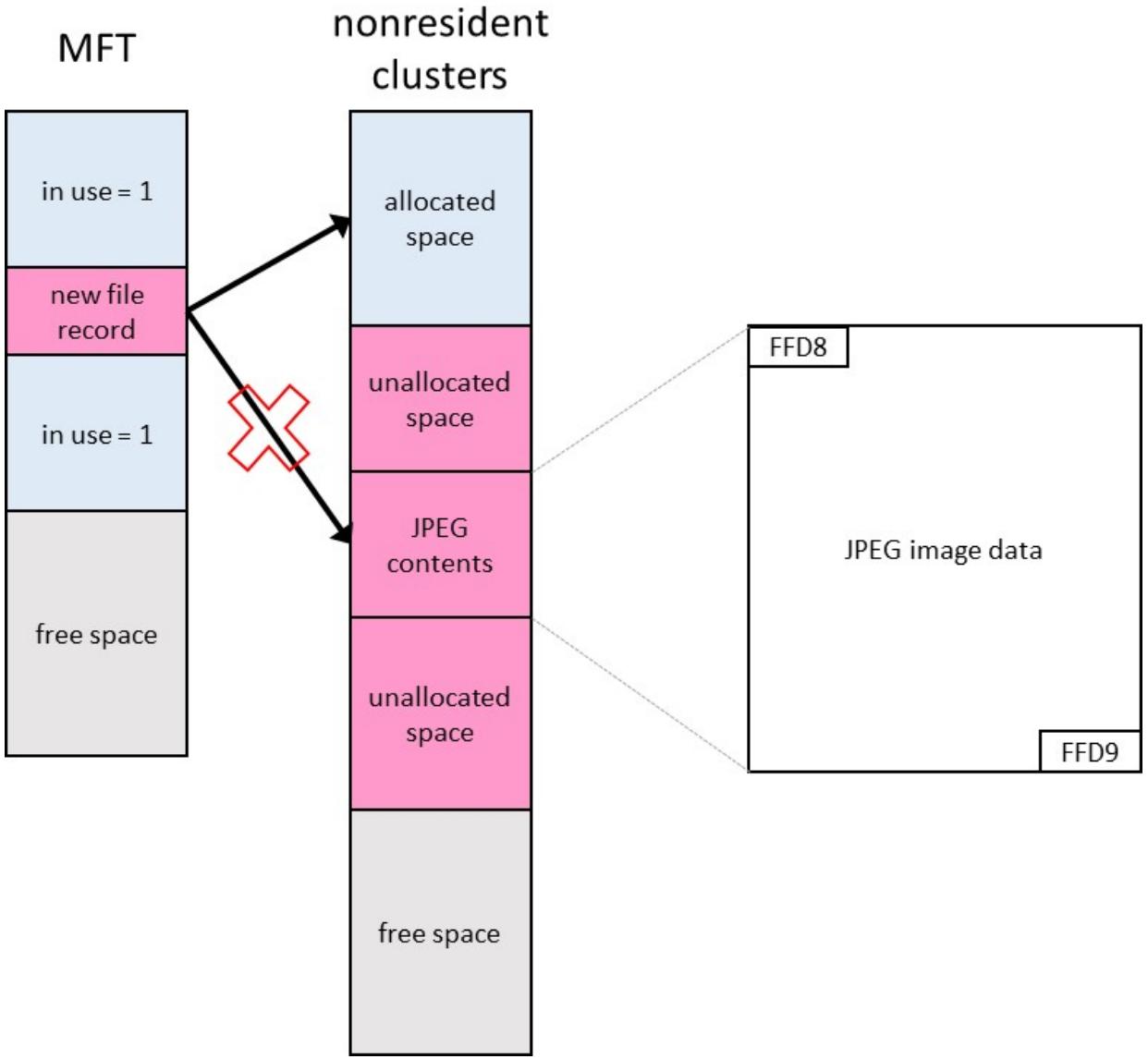
Basic File Carving Techniques

- Header-footer carving
- File-structure-based carving
- Content-based carving

Header-Footer Carving

- Many file types have standard headers and footers stored inside
- Example: A JPEG starts with “Start of image” header 0xFFD8 and ends with “End of image” footer 0xFFD9
- Carve out everything between JPEG header and footer $\rightarrow$ image file

Carving a JPEG



Header-Footer Carving Problems

- Header or footer might be a common string.
  - E.g., the header of an MP3 is “mp”
  - Produces false positives
- The beginning of the file might be missing
  - If file was deleted, might get squashed by a newer file
- The footer might be missing
  - Big output!
- The disk could be fragmented

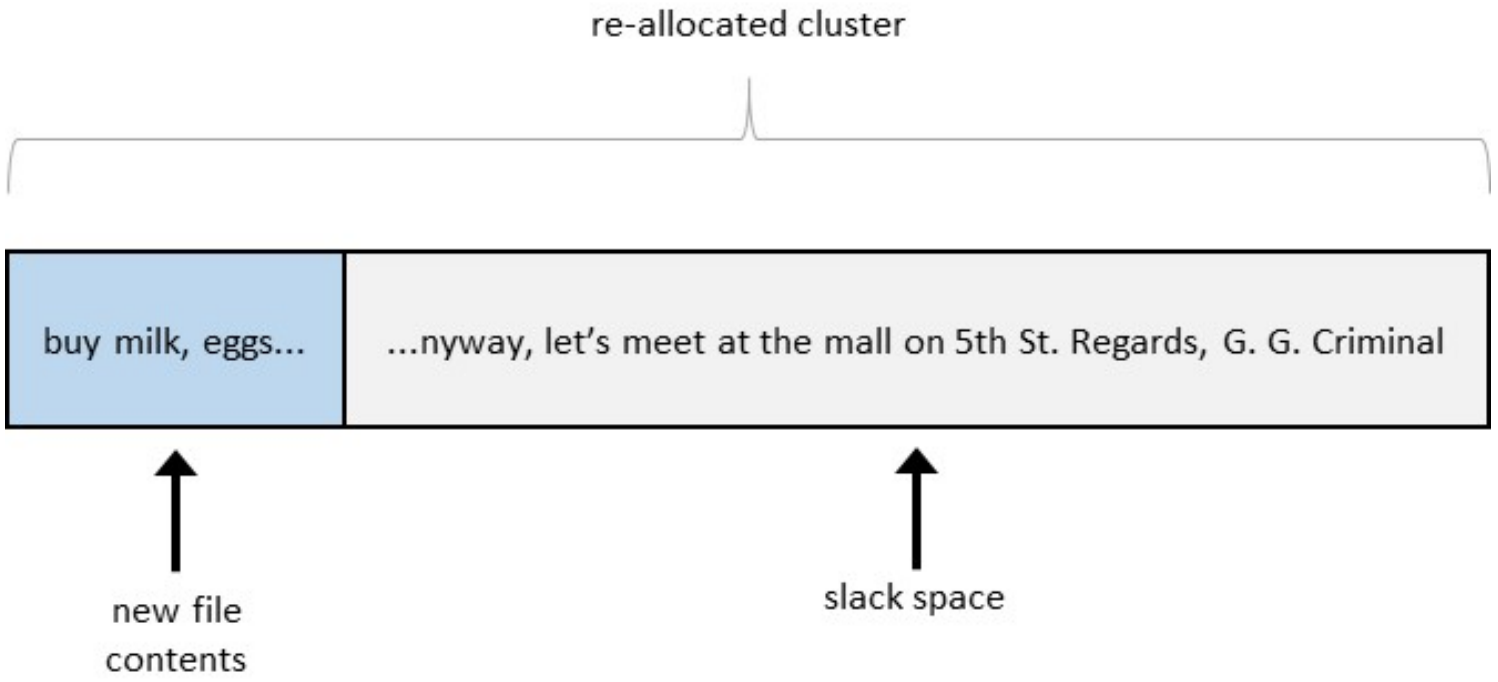
File Structure Carving

- Use internal layout of file if available
- Find cluster size
- Read entire cluster and hunt for internal signatures (in addition to header/footer)
- Example: Foremost, PhotoRec

Content-Based Carvers

- Look for statistical signatures indicating language or file content
- Machine-learning/statistic-based
- Semantic carvers

Slack Space



- *Slack space*: leftover space after file contents in a cluster.
- Old file's data may remain in slack space.

WINDOWS REGISTRY FORENSICS

Registry: A Wealth of Information

Information that can be recovered include:

- System Configuration
- Devices on the System
- User Names
- Personal Settings and Browser Preferences
- Web Browsing Activity
- Files Opened
- Programs Executed

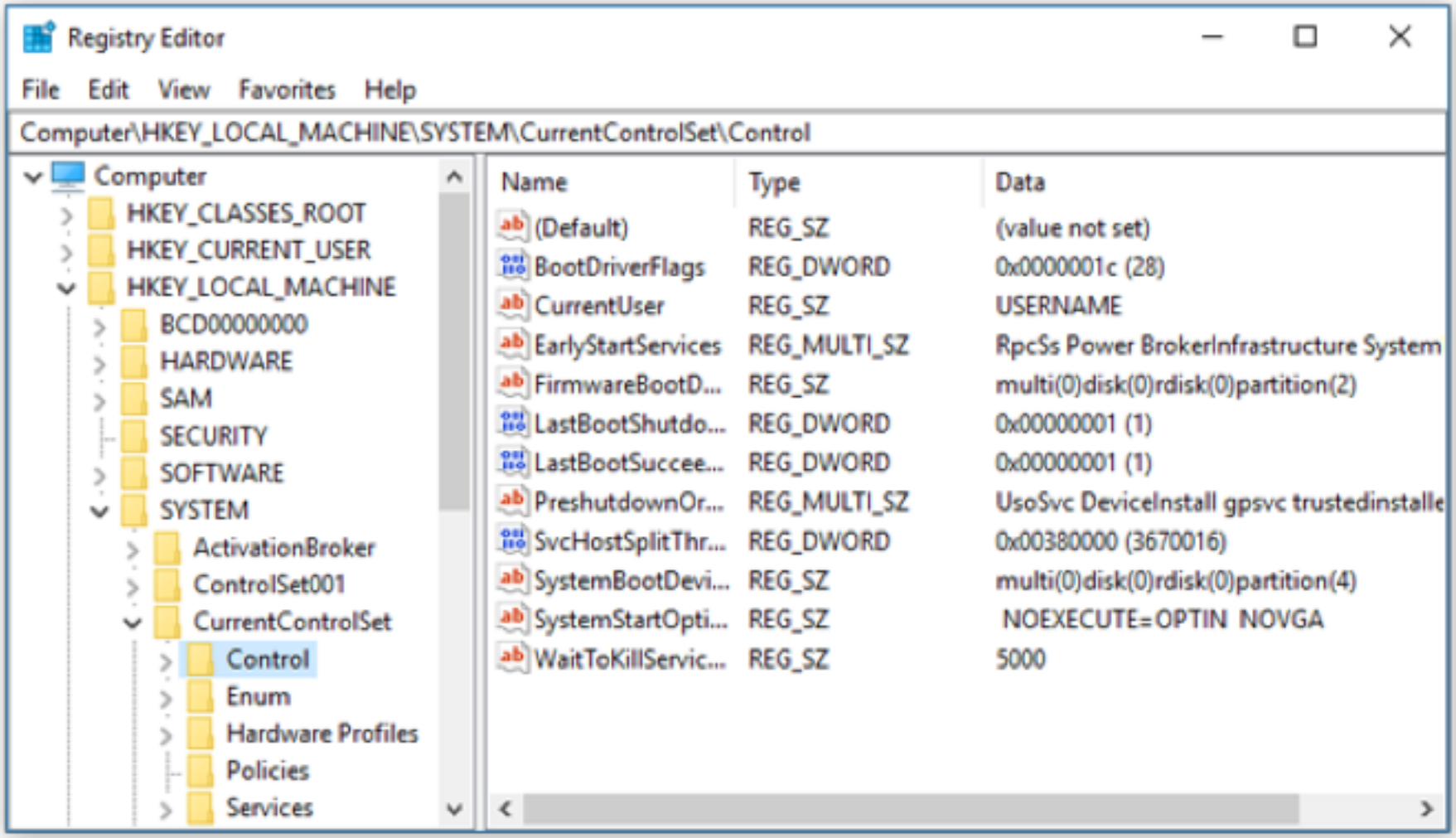
Registry Keys and Values

- Windows Registry is a binary key-value store that also supports subkeys
- A hierarchical database that maintains configuration settings for applications
- Typically there are five top-level entries, or "hives," in the registry

Hives

- **HKEY_USERS:** all actively loaded user profiles for the system
- **HKEY_CURRENT_USER:** actively loaded profile for logged-on user
- **HKEY_LOCAL_MACHINE:** "vast array" of configuration information for the system (hardware and software)
- **HKEY_CURRENT_CONFIG:** hardware profile used at startup
- **HKEY_CLASSES_ROOT:** configuration information related to which applications open which files

Registry Editor



Traces of User Log On/Off

- Last user to log in:
  - HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Authentication\LogonUI\LastLoggedOnUser
- Last time the computer was shut down:
  - HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Windows\ShutdownTime
- Last write time on the ntuser.dat file indicates last logout time.

Connection of USB Devices

- Recently connected USB devices are shown in HKLM\SYSTEM\CurrentControlSet\Enum\USBSTOR
- Contains two levels of subkeys:
  - The device type (e.g., Cruzer micro 1 GB flash storage)
  - The instance ID (usually a serial number, but will be pseudorandom if no serial number is found)

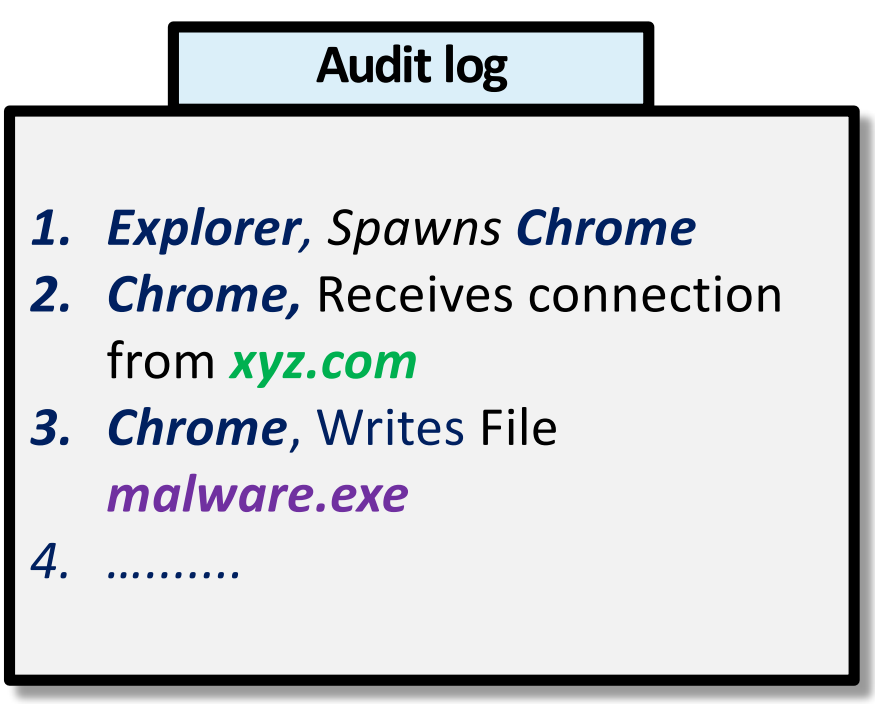
WINDOWS EVENT LOGS

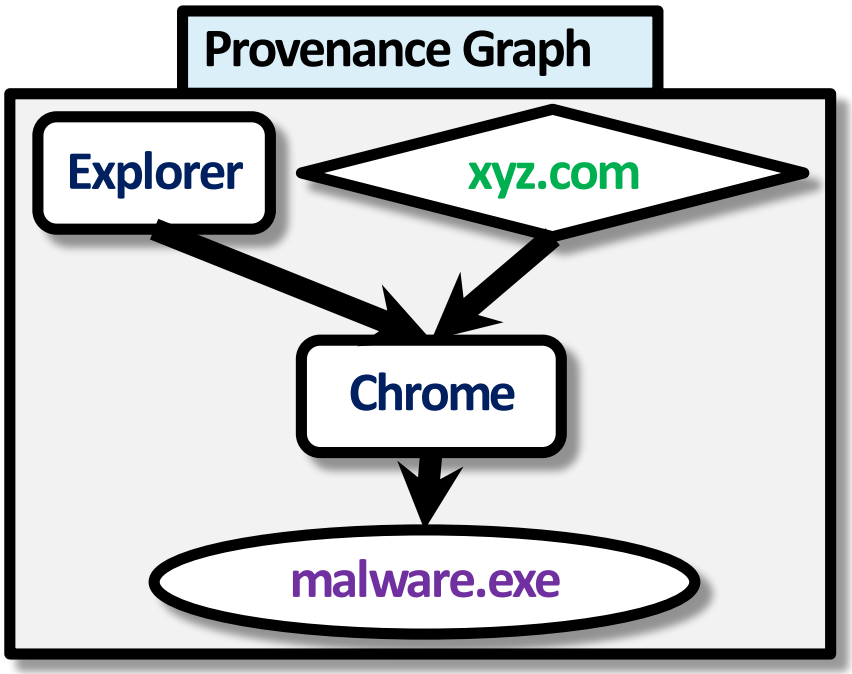
Event Logs

- Event Tracing for Windows (ETW)
  - Designed for performance debugging
  - But can be used for forensics
- Tracks events logged by applications and the system in separate log files in *C:\Windows\system32\winevt\logs*
- Contains information about some of the same events as the Registry

Attack Reconstruction using Logs

- Data Provenance
- Investigate suspicious file = malware.exe





Anti-Forensics

Approaches to deal with encryption

Time stamp modification

Secure file systems

Hardware-based security solutions

Thank You

Any Questions?