DF Lecture 4 Notes

Here's your chapter with **clean formatting**, improved structure, **no change in wording**, and better readability—perfect for exam prep:

Lecture 4 - Disk Structure, Boot Process, File Systems

Dr. Zunera Jalil
Email: zunera.jalil@au.edu.pk

What We Are Going to Talk About Today

- Types of Forensic Requests & Crime Scenes
- Disk Structure, ZBR and Wear Leveling
- Understanding Boot Sequence
- CMOS, BIOS, EFI and UEFI
- File Systems
- Key Files in Windows OS
- Recovering the Evidence Files

Types of Forensic Requests & Crime Scenes

Digital Forensic Investigation Process

- Sources of Evidence
- Device Identification

Types of Forensics Requests

Intrusion Analysis

- Who gained entry?
- What did they do?
- When did this happen?
- How did they do this?

Damage Assessment

- What was available for the intruder to see?
- What did he take?

Tool Analysis

- What tools were used?
- How were they executed?

Log File Analysis

- What events were monitored?
- Firewall/Router/Server log files?

Evidence Search

- Deleted Files
- Hidden Files, Encrypted Files
- Known Remote Access Tools
- Hidden Partitions

A Typical Forensic Investigation Case

1. An incident occurs in XYZ company and company’s server has been compromised.

2. A user contacts company's legal/security department for advice.
3. Legal advisor suggests to get services of a forensic investigator (internal or external?).
4. Forensic investigator seizes the evidence at the crime scene and transports it back to forensic lab.
5. The forensic investigator acquires evidence and creates two bit-stream copies of the disk image.
6. Then creates an MD5 hash of the files.
7. Examines the evidence for proof of crime.
8. Prepares a report concluding investigation finding.
9. Keeps report securely.
10. Legal advisor/advocate studies report and sees the charges/penalties that may apply (may present in court of law).
11. Investigator destroys all evidences.

Packaging Electronic Evidence

Watch This Video:
https://mosequipment.com/blogs/blog/digital-forensics-field-to-lab-evidencehandling-solution

---

Disk Structure

Hard Disk

- Hard disk drives are organized as a concentric stack of disks or 'platters'.
- Each platter has 2 surfaces.

Hard Disk Internal Geometry

- Know the difference: Geometry, Head, Tracks, Cylinders, Sectors

Hard Disk Capacity

- How?

Understanding Hard Disks

- Properties handled at the drive's hardware or firmware level
  - Zone Bit Recording (ZBR)
  - Track density
  - Areal density
  - Head and cylinder skew

Zoned Bit Recording (ZBR)

- Optimizes data storage by dividing the disk surface into concentric zones, each with a varying number of sectors per track.
- Zones and Tracks:
  - ZBR divides the disk into concentric bands called zones. Each zone consists of multiple tracks.
- Varying Sector Density:
  - Outer tracks: more sectors (longer track circumference)
  - Inner tracks: fewer sectors (shorter track circumference)
- Improves media transfer rate (faster on outer cylinders)

Solid State Storage Devices (SSSD)

Wear Leveling

- Technique to prolong service life of flash memory (SSDs, USB flash drives)
- Memory cells usually have 10,000–100,000 read/write limit
- Firmware ensures even wear by shifting data to less-used cells

SSD Wear Leveling

- Old memory cell addresses marked by “Garbage Collector”
- Firmware erases unallocated cells over time
- TRIM command improves garbage collection
- Flash cells use "delete before write" – must erase cell before writing

---

Understanding Boot Sequence

Bootstrap Process

- **CMOS**: Battery-powered chip storing BIOS settings
- **BIOS**: Program embedded on microprocessor; initializes hardware during bootup

---

BIOS, CMOS, EFI and UEFI

- BIOS and UEFI are firmware
- Interpret and initialize hardware
- BIOS reads first sector of hard drive (Sector 0)

EFI and UEFI

- **EFI**: Extensible Firmware Interface
- **UEFI**: Unified Extensible Firmware Interface
    - Newer, includes Secure Boot

The Boot Sequence

- CMOS stores config and date/time
- BIOS/EFI manages input/output at hardware level

- BIOS: Firmware initializes hardware, finds boot device

Steps in Boot Sequence

1. Power Good
2. CPU wakes up
3. POST
4. Loading Drivers
5. Video
6. Memory
7. The Hands OFF
8. Bootstrap Loader

File Systems

File System

- Gives OS a road map to data on disk
- Determines data storage method
- Must know OS and file system

Understanding File System

File: A collection of data with a name
FAT: File Allocation Table – OS uses it to locate files

- Two copies kept for recovery
- Fixed storage location
- FAT stores:
    - Unused (0x0000)

    - Used cluster
    - Bad cluster (0xFFF7)
    - Last cluster (0xFFF8–0xFFFF)
- Starting cluster = file's first cluster

More Concepts

- Lost Cluster
- Fragmentation

File System and Operating System

- Depends on OS
- Newer OS = More supported files
- DOS/Win95 = FAT16
- FAT12: For floppy disks (up to 16MB)
- FAT32: Supports up to 2TB

---

exFAT

- Introduced in 2006 by Microsoft
- Intended for portable media devices

NTFS

- Windows NT File System
- Supports spanning volumes (files over multiple disks)

Key System Files in Windows OS

- **Hiberfile.sys**: Stores memory state in Hibernate Mode
- Used in memory dump, pagefile, hiberfile investigations

Key Windows Files:

- Ntoskrnl.exe
- Ntkrnlpa.exe
- Hal.dll
- Win32k.sys
- Ntdll.dll
- Kernel32.dll
- Advapi32.dll
- User32.dll
- Gdi32.dll

HPA and DCO

- **HPA (Host Protected Area)**
  - Reserved, read-only HDD area not accessible to users
- **DCO (Device Configuration Overlay)**
  - Used by vendors for recovery tools or hidden information
  - Set before shipping HDD

HEX Editors

WinHEX

- File type search
- File restoration
- Binary ↔ Hex conversion
- Deleted data in hidden partitions
- Volume snapshotting
- Directory browsing
- OFFSET
- File Carving

File Carving

- Forensic technique to recover files based on content (signatures)
- Works without file system metadata
- Looks for file fragments in damaged/unallocated space

File Types and HEX Values

File Type	HEX Signature
PDF	25 50 44 46
PNG	89 50 4E 47
BMP	42 4D
JPEG	FF D8 FF
MPEG/MP3	FF Ex or FF Fx
MP3	49 44 33 03
WAV	57 41 56 45

File Type	HEX Signature
DOCX	50 4B 03 04 14 00 06 00
JAR	50 4B 03 04 14 00 08 00 08 00
ZIP	50 4B 03 04
RAR	52 61 72 21 1A 07
WMV	30 26 B2 75 8E 66 CF 11
AVI	41 56 49 20
GIF	47 49 46
DOC/PPT/XLS	D0 CF 11 E0 A1 B1 1A E1

Reference:
https://www.garykessler.net/library/file_sigs.html

Home Task

- Perform exercises under FORENSICS tab:
  https://www.computersecuritystudent.com/FORENSICS/Windows/Miscellaneous/lesson3

References

- Chapter 4 – Text Book
- Chapter 1, 2, and 3 – *File System Forensic Analysis* by Brian Carrier

---

Let me know if you'd like this as a **PDF or Word doc** for printing or annotating!

A Typical Forensic Investigation Case (Shortened)

1. Incident occurs, compromising the company's server.
2. User contacts legal/security department.
3. Legal advisor recommends a forensic investigator.
4. Investigator collects and transports evidence to lab.
5. Creates two bit-stream copies of the disk image.
6. Generates MD5 hash of the files.
7. Analyzes evidence for criminal activity.
8. Prepares and secures the investigation report.
9. Legal advisor reviews the report for legal action.
10. Investigator destroys all evidence post-case.

To solve the given problem, we'll calculate:

(a) Block size B (in bytes)

We are given:

- **Total storage capacity** = 9.09 GB = 9.09×2^{30} bytes ≈ **9,760,751,616 bytes**
- **Number of sectors on all recording surfaces** = 17,755,614

Now, block size B =

$$\frac{\text{Total storage in bytes}}{\text{Number of sectors}} = \frac{9,760,751,616}{17,755,614} \approx \boxed{550 bytes}$$

(b) Average block access time t_B (in ms)

Block access time typically includes:

1. **Average seek time** = 11.5 ms
2. **Average rotational latency** = $\frac{1}{2} \times \text{rotation time}$
   - Rotation speed = 5400 RPM = 90 RPS
   - One rotation = $\frac{1}{90}$ sec ≈ 11.11 ms
   - So, average latency ≈ $\frac{11.11}{2}$ = **5.56 ms**
3. **Transfer time**

- Block size = 550 bytes
- Use the **average** of data transfer rate: $\frac{44+65}{2} = 54.5$ MB/s = 54.5×10^6 bytes/s
- Transfer time = $\frac{550}{54.5 \times 10^6} \times 1000$ ms $\approx$ 0.01 ms

Now, summing them up:

$$t_B = 11.5 + 5.56 + 0.01 = \boxed{17.07 \text{ ms}}$$

✅ Final Answers:

- **(a)** Block size $B \approx$ **550 bytes**
- **(b)** Average block access time $t_B \approx$ **17.07 ms**

Do you like this personality