Digital Forensics

Lab 02: Digital Forensic Foundations

Master Exam-Ready Study Notes — 2025/26

Contents

1 Lab 02 Overview & Objectives

This lab introduces the foundational tools and techniques every DF investigator needs before touching evidence:

1. **Install a Windows VM** — setting up an isolated examination environment.
2. **File & folder properties** — revealing hidden metadata that Windows conceals from basic users.
3. **Software write blocker** — preventing evidence contamination via a Registry modification.
4. **FTK Imager** — creating forensic images, verifying integrity with hashes, and examining evidence.

Why This Lab Matters Every step in this lab maps directly to the DF process: you **prepare** your workstation (VM + write blocker), **acquire** evidence (FTK Imager), **verify** integrity (hashing), and **examine** the image (Evidence Tree, Recycler analysis). These are the skills tested in exams and used in real investigations.

2 Lab 2.1 — Installing a Windows VM

2.1 Why Use a Virtual Machine?

Definition — Virtual Machine (VM) A **Virtual Machine** is a software-emulated computer running inside your host system. It has its own OS, storage, and memory, fully isolated from the host. In DF, VMs allow you to examine evidence and run forensic tools **without risking contamination** of your host machine or the original evidence.

The lab uses **VirtualBox** with a **Windows 10** (or 11) VM.

Detail	Value
Platform	Oracle VirtualBox (from Lab 01)
VM Image	MSEdge Win10 VirtualBox image (valid 90 days)
Credentials	User: `IEUser` Password: `Passw0rd!`
Alternative	Windows 11 ISO (25H2), 4 GB RAM, 60 GB disk, Custom Install, skip product key
Recommendation	Take a **snapshot** immediately after setup

EXAM KEY POINT — Snapshots A **VM snapshot** captures the entire state of the virtual machine at a point in time. If the VM expires (90-day limit) or you make a mistake during analysis, you can revert to the snapshot instantly. This is analogous to creating a restore point but at the virtualisation level.

UEFI Firmware (Windows 11) For optimum performance with Windows 11, UEFI firmware must be enabled. You can enable it via:

```
shutdown /r /fw /t 0
```

This restarts Windows directly into UEFI firmware settings. UEFI is the modern replacement for BIOS and is required by Windows 11.

3 Lab 2.2 — File & Folder Properties

3.1 File Attributes as Metadata

Definition — File Attribute A **file attribute** is metadata that describes or is associated with a computer file. The OS tracks: creation date, last modified date, file size, file extension (and associated application), and file permissions. This metadata is critical forensic evidence.

3.2 Revealing Hidden Information in Windows Explorer

By default, Windows **hides** important information from users. As a DF analyst, you must configure Explorer to show **everything**:

Procedure: Unhide Files in Windows Explorer

```
1. Open Windows Explorer (Win + E)
2. View > Options > Change folder and search options > View tab
3. ENABLE:  "Show hidden files, folders, and drives"
4. DISABLE: "Hide empty drives"
5. DISABLE: "Hide extensions for known file types"
6. DISABLE: "Hide protected operating system files"
7. Click "Apply to Folders" > Yes > OK
```

3.3 What Gets Revealed

After changing these settings, you can now see system files that were previously invisible on `C:\`:

File	Forensic Significance
`hiberfil.sys`	**Hibernation file** — contains a snapshot of RAM when the system hibernates. Can contain passwords, encryption keys, open documents, and running processes. Enormous forensic value for recovering volatile data.
`pagefile.sys`	**Page file (virtual memory)** — the OS swaps RAM contents to this file when memory is full. May contain fragments of any data that was ever in RAM: documents, passwords, chat messages.
`swapfile.sys`	**Swap file** — similar to pagefile.sys but used specifically for Windows Store apps (UWP apps). Contains memory swapped from modern apps.
`msdia80.dll`	A Microsoft DIA (Debug Interface Access) SDK library. Less forensically significant but shows that system files are now visible.

EXAM ALERT

Exam favourite: "As a basic user, you would not know these files were there." This is why DF analysts **must** change Explorer settings — hidden system files like `hiberfil.sys` and `pagefile.sys` can contain forensically valuable data (RAM snapshots, fragments of deleted documents, passwords, etc.) that a normal user would never see.

3.4 Column Headings & Date Types

Windows Explorer shows 4 default columns: **Name, Date modified, Type, Size**. By right-clicking any column heading and selecting "More," you can add many additional columns including:

Date Column	What It Records
Date created	When the file was first written to this location
Date modified	When the file's **content** was last changed
Date accessed	When the file was last **read** (opened/previewed)

EXAM KEY POINT — Reliability of Dates Different date types serve different forensic purposes for building a **chronology of events**. However, dates can be **unreliable**: they may be changed by copying, moving, or deliberately manipulated by a suspect. A DF investigator must understand *what* each date records, *how* it might get changed, and whether it is *reliable*.

4 Lab 2.3 — Software Write Blocker

4.1 What Is a Write Blocker?

Definition — Write Blocker A **write blocker** is a tool (hardware or software) that prevents any write operations to a storage device. It allows a forensic examiner to **read** data from a device without **modifying** it. This is essential to preserve the integrity of digital evidence.

	Hardware Write Blocker	Software Write Blocker
How it works	Physical device placed **between** the forensic workstation and the evidence device	Registry modification or driver-level block within the OS
Advantages	Tamper-proof, accepted universally in court, independent of OS	Free, quick to set up, no additional hardware needed
Disadvantages	Costs money, must carry physical device	Can be circumvented, may not be accepted in all courts
Used when	Real forensic investigations, court-admissible evidence	Lab work, preliminary examination, triage

EXAM ALERT

Critical exam point: In real forensic scenarios, a **hardware-based write blocker** should *always* be used between the forensic workstation and the evidence device. The software write blocker from this lab is a convenient alternative but is **not a substitute** for hardware write blockers in court-admissible investigations.

4.2 Why Write Protection Matters

When you plug a USB device into a Windows machine **without** write protection, the OS can:

- Update access timestamps on files
- Write Recycle Bin metadata (`$Recycle.Bin` folder)
- Create System Volume Information
- Auto-index files for Windows Search
- Install device drivers and create registry entries

All of these **modify the original evidence**, potentially destroying forensic data and breaking the chain of custody.

4.3 Creating a Software Write Blocker via the Registry

Definition — Windows Registry The **Windows Registry** is a hierarchical database that stores configuration settings for the OS, hardware, installed software, and user preferences. It is organised into "hives" (top-level keys), "keys" (folders), and "values" (data entries). Modifying specific values can change system behaviour — including enabling write protection for USB devices.

4.3.1 Step 1: Create a System Restore Point

Definition — System Restore Point A **System Restore Point** is a snapshot of the system's configuration (Registry, system files, installed programs) at a specific moment. If a Registry change causes problems, you can revert to the restore point. It does **not** back up personal files — only system settings.

Procedure: Create a Restore Point

```
1. Start menu > search "Create a restore point" > Enter
2. System Protection tab > verify Local C: is "On"
   (if Off, click Configure > Turn on system protection)
3. Click "Create..." > give it a description > Create
```

4.3.2 Step 2: Modify the Registry

Procedure: Enable USB Write Protection

```
Step 1: Start > search "regedit" > Run as Administrator > Yes

Step 2: Navigate to:
  HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control

Step 3: Right-click Control > New > Key
  Name it: StorageDevicePolicies

Step 4: Right-click StorageDevicePolicies > New > DWORD (32-bit)
  Name it: WriteProtect

Step 5: Right-click WriteProtect > Modify
  Value = 1  (write protection ON)
  Value = 0  (write protection OFF)
```

EXAM KEY POINT — The Registry Path The full path is:

`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\StorageDevicePolicies`

- `CurrentControlSet` = the configuration currently in use by the system (not a backup).
- The system keeps backups as `ControlSet001`, `ControlSet002`, etc.
- The `Select` key indicates which control set is active.
- `StorageDevicePolicies` is the key you *create*.
- `WriteProtect` DWORD: **1 = ON** (blocked), **0 = OFF** (writable).

4.3.3 Step 3: Automate with .REG Files

Instead of manually editing the registry each time, you can export `.reg` files:

File	WriteProtect Value	Effect
USB Write Protection On.reg	1	Double-click to **enable** write protection
USB Write Protection Off.reg	0	Double-click to **disable** write protection

How to create: Right-click `StorageDevicePolicies` → Export → save as `.reg` file. Toggle the value between 0 and 1, exporting each version.

5 Lab 2.4 — FTK Imager

5.1 What Is FTK Imager?

Definition — FTK Imager **FTK Imager** (by Exterro/AccessData) is a **data preview and imaging tool** used to:

- Create forensic images (bit-for-bit copies) of storage media
- Preview files and folders without modifying the original
- Recover deleted files from the Recycle Bin
- Generate and verify cryptographic hashes (MD5, SHA-1)
- Mount images as read-only drives
- Export files from forensic images

It is a **triage/acquisition** tool. Full forensic examination is done with tools like EnCase, FTK, or Autopsy.

5.2 FTK Imager Interface

Pane / Menu	Purpose
Evidence Tree	Upper-left. Hierarchical tree of evidence sources, folders, and files. Click + to expand, − to collapse.
File List	Upper-right. Displays contents of whatever is selected in the Evidence Tree. Shows Name, Size, Type, Date Modified.
Properties Pane	Lower-left. Shows metadata: object type, storage location, size.
Viewer Pane	Lower-right. Displays the data content of the selected file (hex, text, or rendered view).
View Menu	Customise which panes are visible
Mode Menu	Switch preview modes (Automatic, Text, Hex)
Help Menu	FTK Imager User Guide and version info

5.3 Forensic Imaging — Core Concepts

5.3.1 Chain of Custody

Definition — Chain of Custody The **chain of custody** is the documented, chronological history of evidence showing who collected it, who handled it, where it was stored, and every transfer between individuals. It proves that evidence has not been tampered with from seizure through to court presentation. A break in the chain can render evidence **inadmissible**.

5.3.2 Guaranteeing Evidence Integrity

EXAM KEY POINT — How Do We Guarantee Evidence Has Not Been Modified? **Cryptographic hashing.** After creating a forensic image, FTK Imager computes hash values (MD5 and SHA-1) of both the **original** source and the **image**. If the hashes **match**, the image is a perfect bit-for-bit copy. If re-verified later and they still match, the evidence has not been altered since acquisition.

5.3.3 MD5 and SHA-1 Hashing

Algorithm	Output Size	Description
MD5	128 bits (32 hex chars)	Message Digest 5. Fast but theoretically vulnerable to collisions. Still widely used in DF for speed.
SHA-1	160 bits (40 hex chars)	Secure Hash Algorithm 1. More collision-resistant than MD5. Often used alongside MD5 for dual verification.

Why are they important?

- They act as **digital fingerprints** — even a 1-bit change produces a completely different hash.
- FTK Imager generates three hash values: **Computed hash** (calculated now), **Stored verification hash** (from acquisition time), and **Report hash**.
- If all three match → "**Verify result: Match**" → evidence integrity confirmed.

5.4 Forensic Image Types

FTK Imager supports four image formats:

Format	Extension	Description
Raw (dd)	`.dd` / `.raw`	Exact bit-for-bit copy with no compression or metadata. Largest file size. Universally compatible.
SMART	`.s01`	Linux-based format. Supports compression. Less common.
E01 (EnCase)	`.E01`	**Most common DF format.** Supports compression, case metadata (case number, examiner, notes), and built-in hash verification. Used in the lab.
AFF	`.aff`	Advanced Forensic Format. Open-source, supports compression and metadata.

EXAM KEY POINT — E01 Format E01 is preferred because it stores **case metadata** (Case Number, Evidence Number, Examiner Name, Description, Notes) alongside the image data, and includes **built-in integrity verification** (MD5/SHA-1 hashes computed at acquisition time).

5.5 Logical vs. Physical Acquisition

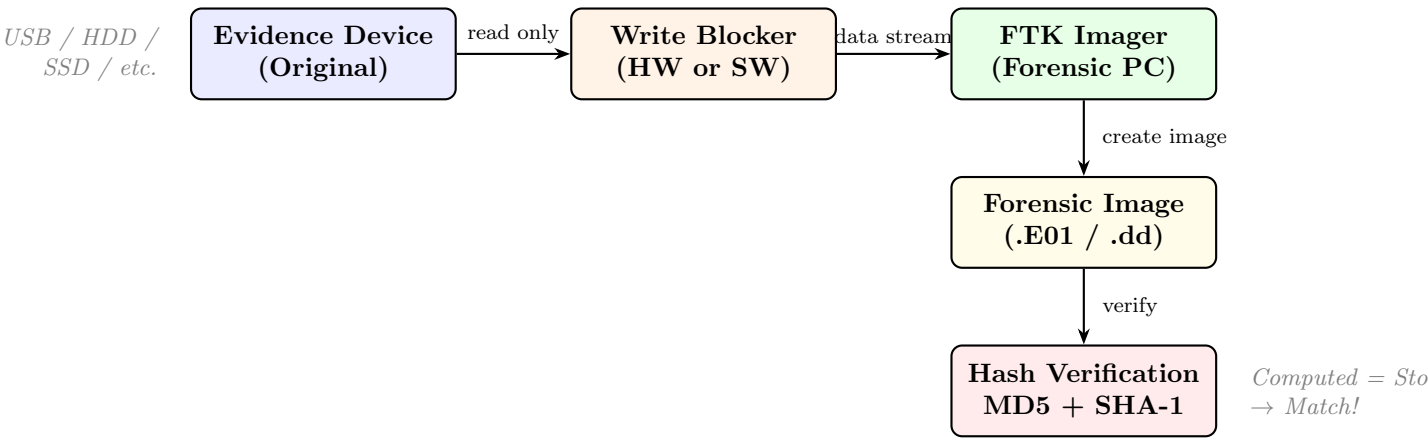
	Logical Acquisition	Physical Acquisition
What it copies	Only the **file system layer**: files, folders, and their metadata as the OS presents them	The **entire physical disk**: every sector, bit-for-bit, including areas the OS cannot see
Includes deleted files?	Only if still visible in the file system (e.g. Recycle Bin)	**Yes** — includes deleted files, file slack, unallocated space, and hidden partitions
Includes slack space?	No	**Yes**
Includes unallocated space?	No	**Yes**
Speed	Faster (less data to copy)	Slower (copies every sector)
When to use	Quick triage, when only specific files are needed	**Full forensic investigation** — the gold standard
FTK Imager option	"Logical Drive"	"Physical Drive"

EXAM ALERT

Critical distinction: A logical acquisition is like photocopying visible pages in a book. A physical acquisition is like photographing every page *including* the ones that have been torn out and glued back together. For court-admissible forensics, **physical acquisition** is almost always required because it captures **everything** on the disk.

5.6 Bit-for-Bit Imaging

Definition — Bit-for-Bit (Forensic) Image A **bit-for-bit image** is an exact, sector-by-sector duplicate of a storage device. It is "identical in every way to the original, including **file slack** and **unallocated space** or drive free space" (from the lab sheet). The original media is then stored safely while the investigation proceeds using the image.



5.7 Evidence Item Information (E01 Metadata)

When creating an E01 image, FTK Imager prompts for case metadata:

Field	Purpose
Case Number	Unique identifier for the investigation (e.g. `Case_1`)
Evidence Number	Identifier for this specific piece of evidence (e.g. `Ev_1`)
Unique Description	Brief description (e.g. `USB_Drive`)
Examiner	Name of the forensic examiner (e.g. `RS`)
Notes	Additional context (e.g. `USB drive image`)

Additional settings at image creation:

- **Image Fragment Size (MB):** Split large images into chunks (default 1500 MB). Set to 0 for no fragmentation.
- **Compression:** 0 = None, 1 = Fastest, 9 = Smallest. Lab uses 6 (balanced).
- **Verify images after creation:** Should **always** be ticked to confirm hash match.

6 Examining Evidence — The USB Key Case (`usb_a.E01`)

The lab provides a pre-made forensic image (`usb_a.E01`) of a USB key for examination.

6.1 Loading Evidence in FTK Imager

Procedure: Add Evidence Item

```
1. File > Add Evidence Item
2. Select "Image File" > Next
3. Browse to usb_a.E01 > Open > Finish
4. Expand the Evidence Tree to explore contents
```

The Evidence Tree shows the image is **FAT12** file system with folders including: `10-1 Graphics`, `D/F Search Stuff`, `Recent`, `Recycler`, `Registry File`, `Sugan Alert`, `TIF Netmail`, `Zip Files`, and `[unallocated space]`.

6.2 The Recycler Folder and SID Folders

Definition — SID (Security Identifier) A **Security Identifier (SID)** is a unique alphanumeric string that Windows assigns to every user account, group, and computer. SIDs are used internally by the OS for access control and security. They look like: `S-1-5-21-1929781967-25...`

Each folder within `Recycler` is named with a SID. Key facts:

SID Ending	Meaning
...-500	The **built-in Administrator** account. SID ending in 500 *always* means Administrator on Windows.
...-1004	A **regular user account**. User accounts created on the system are assigned RIDs (Relative Identifiers) starting from 1000. So 1004 is the 5th user account created.

EXAM KEY POINT — SID Forensic Significance SID folders in the Recycler allow a DF investigator to determine **which user account** deleted specific files. By mapping SIDs to usernames (via the SAM registry hive or `ProfileList` key), you can attribute deleted file activity to individual users.

6.3 The INFO2 File

Definition — INFO2 File The **INFO2** file (found inside each SID folder within the Recycler) is a database that records information about files sent to the Recycle Bin:

- Original file name and path
- Date/time of deletion
- Original file size
- Drive letter the file came from

Key detail: Restored files have **no drive letter** in their INFO2 entry. This allows you to distinguish between files that were deleted vs. files that were deleted and then restored.

Every SID folder has an INFO2 file, but in this evidence image only one contains actual data.

6.4 Windows Shortcuts (.lnk Files)

Under `root/Recent`, the lab identifies `.lnk` files — Windows shortcut files.

EXAM KEY POINT — .lnk Files as Forensic Evidence Windows creates `.lnk` (shortcut) files in the Recent folder whenever a user opens a file. Even if the **original file is deleted**, the `.lnk` file may persist, providing evidence that:

- A specific file existed
- A specific user opened it
- The original path and timestamps of access

`.lnk` files are rich forensic artefacts that survive long after the original file is gone.

6.5 Browser Artefacts — index.dat and TIF Netmail

Definition — index.dat The **index.dat** file is a database used by Internet Explorer to store browser usage information: cookies, browsing history, and Temporary Internet Files (TIF). It records URLs visited, timestamps, and cached content locations.

In the `TIF Netmail` folder:

- `index.dat` contains browsing history entries (view in Text mode).
- Subfolders contain cached **image files** and **.htm files** (web page fragments).
- You can match URLs in `index.dat` to the cached images — proving what websites the user visited and what content was displayed.
- View modes: **Automatic** (rendered), **Text** (raw text), **Hex** (raw bytes).

EXAM KEY POINT — Correlating Browser Evidence Matching `index.dat` entries with cached images in TIF subfolders allows a DF investigator to **reconstruct the user's browsing activity**: what sites they visited, when, and what content they viewed. This is powerful evidence in cases involving illegal downloads, communication, or web-based crimes.

7 Complete Lab 02 Exam Quick-Reference

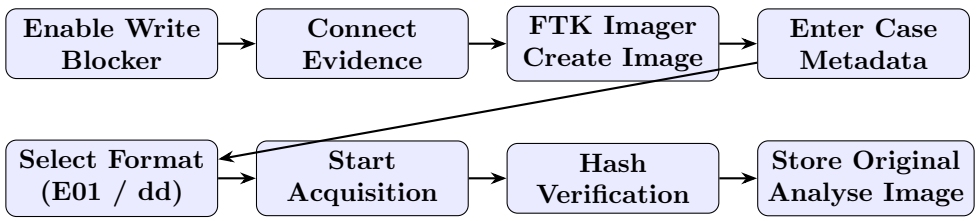
7.1 Key Definitions at a Glance

Term	Definition
File Attribute	Metadata describing a file (dates, size, extension, permissions)
Write Blocker	Tool preventing write operations to evidence media
Forensic Image	Bit-for-bit copy of a storage device including slack and unallocated space
Chain of Custody	Documented history of who handled evidence and when
MD5 / SHA-1	Cryptographic hash algorithms producing digital fingerprints of data
Logical Acquisition	Image of the file system layer only (files/folders as OS sees them)
Physical Acquisition	Image of the entire physical disk (every sector, including hidden data)
SID	Security Identifier — unique string identifying a Windows user
INFO2	Recycle Bin database listing deleted files, their paths, and deletion dates
index.dat	Internet Explorer database of browsing history, cookies, and TIF

7.2 The Registry Write Blocker Path

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\StorageDevicePolicies
                WriteProtect (DWORD) = 1 (ON)   /   0 (OFF)
```

7.3 Forensic Imaging Workflow



7.4 Tools Used in This Lab

Tool	Type	Purpose
VirtualBox	Virtualisation	Isolated Windows 10/11 examination environment
Windows Explorer	Built-in	Reveal hidden files, system files, file extensions
Registry Editor (regedit)	Built-in	Modify system configuration to enable write protection
FTK Imager	Forensic tool	Create forensic images, preview evidence, hash verification

7.5 Must-Know Summary

8 Things You Must Know from Lab 02

1. A DF analyst must **unhide** system files, hidden files, and file extensions in Explorer.
2. `hiberfil.sys` and `pagefile.sys` contain RAM snapshots — forensic gold.
3. The software write blocker uses a Registry DWORD: `WriteProtect = 1` under `StorageDevicePolicies`.
4. **Physical acquisition** captures everything (slack, unallocated); **logical** captures only the file system.
5. **E01** format is preferred: supports compression, case metadata, and built-in hash verification.
6. **MD5** (128-bit) and **SHA-1** (160-bit) hashes prove evidence integrity; all three hash values must match.
7. SID folders in the Recycler identify **which user** deleted files; SID ending in **500 = Administrator**.
8. `INFO2` files list deleted files; restored files have **no drive letter** in the entry.

End of Master Study Notes — Lab 02: Digital Forensic Foundations