

Lab 3: Establishing Evidence Integrity and Chain of Custody

MSCS630: Cybercrime Investigation and Digital Evidence

Dr. Mouatez Karbab



September 17, 2026

1 Lab Overview and Scenario

You are a Junior Digital Forensics Investigator assigned to a potential corporate espionage case, designated “Case M57-Patents”. An employee of the company M57 is suspected of exfiltrating sensitive intellectual property, specifically new patent filings. The IT department has imaged the suspect’s work USB drive in EnCase format and has made the image available to you, together with the SHA-256 hash it recorded at the moment of imaging.

Your first and most critical task is not to analyse the contents of the image. It is to establish a forensically sound foundation for the entire investigation: acquire the evidence file, prove that it has not changed since IT imaged it, create a verified working copy for your analysis, and document every action in a Chain of Custody log. A mistake at this stage can render every later finding inadmissible. This lab simulates that foundational process.

2 Learning Objectives

Upon successful completion of this lab, students will be able to:

- Explain why the Chain of Custody decides whether digital evidence is admissible.
- Create and maintain a Chain of Custody log that records every action taken on an item.
- Use command-line tools (`wget`, `sha256sum`, `cp`, `date`) to acquire and handle digital evidence.
- Distinguish original evidence from a forensically sound working copy.
- Apply SHA-256 hashing to verify the integrity of evidence at each stage.
- Relate these procedural steps to the authenticity and admissibility requirements of Chapter 3.

3 Prerequisites: Tools & Data Setup

This lab requires a BASH-compatible command line: Linux, macOS, or the Windows Subsystem for Linux (WSL) on Windows 10/11.

3.1 Tool Installation: Command-Line Utilities

The tools, `wget` and `coreutils` (which provides `sha256sum`, `cp` and `date`), are standard on Linux. macOS ships `shasum` instead of `sha256sum`; use `shasum -a 256` wherever this handout says `sha256sum`, and `curl -O` wherever it says `wget`.

- **GUI Instructions:** There is no GUI tool for the core of this lab. The Chain of Custody log can be kept in a spreadsheet (LibreOffice Calc, Excel, Google Sheets) or a plain text editor.
- **CLI Instructions (BASH):**

1. Open your terminal and check that the tools are present.

```
1 wget --version | head -1
2 sha256sum --version | head -1
3
```

2. If either command is missing, install it.

```
1 # Debian / Ubuntu
2 sudo apt-get update && sudo apt-get install -y wget coreutils
3
4 # Fedora / Red Hat
5 sudo dnf install -y wget coreutils
6
```

3.2 Data Source: The M57-Patents Work USB Image

The evidence is a disk image from the **M57-Patents scenario** of the Digital Corpora project, a public dataset built for forensic education. The scenario contains daily images of four employees' computers, each several gigabytes. This lab uses the image of Charlie's work USB drive taken on 11 December 2009, which is small (about 9 MB) and is a genuine EnCase (.E01) image.

Digital Corpora publishes the image but not a hash beside it. The course therefore issues the hash file itself, standing in for the IT department's record: it was computed when the course downloaded the image, and it is what your copy must match.

1. **Create a dedicated case directory.** Keep the evidence apart from everything else on your machine.

```
1 mkdir M57-Patents-Case
2 cd M57-Patents-Case
3
```

2. **Download the image and the course-issued hash file.**

- **Disk image (Digital Corpora):** <https://digitalcorpora.s3.amazonaws.com/corpora/scenarios/2009-m57-patents/usb/charlie-work-usb-2009-12-11.E01>
- **Official hash (course portal):** <https://courses.karbab.net/mscs630/charlie-work-usb-2009-12-11.E01.sha256>

```
1 # The disk image (about 9 MB)
2 wget https://digitalcorpora.s3.amazonaws.com/corpora/scenarios/2009-m57-
   patents/usb/charlie-work-usb-2009-12-11.E01
3
4 # The file holding the official SHA-256 hash
5 wget https://courses.karbab.net/mscs630/charlie-work-usb-2009-12-11.E01.
   sha256
6
```

3. **Do not verify yet.** Verification is a logged action, and the log does not exist until Step 1.

4 Lab Procedure: Step-by-Step Analysis

Follow these steps in order, documenting each action in your Chain of Custody log *as you perform it*. Take the timestamp with `date -u` at the moment of the action, not later when you tidy the log: a log reconstructed from memory does not meet the reproducibility standard of Chapter 4.

4.1 Step 1: Create a Chain of Custody Log

Before touching the evidence file, create the log that will track its entire lifecycle under your control. This document is as important as the evidence itself.

1. Create the Log File

- **GUI Instructions:** Open a spreadsheet application (LibreOffice Calc, Excel) or a text editor (Gedit, Notepad++, VS Code).
- **CLI Instructions:** Use a command-line editor such as `nano`.

```
1 nano chain_of_custody_M57.txt
2
```

- *Observation/Expected Outcome:* An empty file, ready for logging.

2. Set up the Log Structure

- **GUI Instructions:** Create columns for Case ID, Item ID, Item Description, Custodian, Timestamp (UTC) and Notes/Action Taken.
- **CLI Instructions:** Write a header line, separating fields with pipes (`|`).

```
Case | Item | Description | Custodian | Time (UTC) | Action
-----|-----|-----|-----|-----|-----
```

- *Observation/Expected Outcome:* The log is formatted and ready for its first entry.

4.2 Step 2: Log the Acquisition of Original Evidence

With the log ready, document the evidence that is already in your case directory.

1. Confirm the Evidence Is Present

- **GUI Instructions:** Open the M57-Patents-Case folder in your file manager and confirm both files are there.
- **CLI Instructions:** List the directory with sizes. If you skipped the setup download, run the `wget` commands from section 3.2 now.

```
1 ls -l
2
```

- *Observation/Expected Outcome:* `charlie-work-usb-2009-12-11.E01` (9,265,553 bytes) and `charlie-work-usb-2009-12-11.E01.sha256` are present.

2. Make the First Log Entry

- **GUI/CLI Instructions:** Take the UTC time, then add the first entry. The Item ID is an identifier you assign and will reuse for every action on this item.

```
1 date -u
2
```

- **Case ID:** M57-Patents
 - **Item ID:** M57-USB-Orig
 - **Item Description:** Original evidence image, Charlie’s work USB drive (charlie-work-usb-2009-12-11.E01, 9,265,553 bytes)
 - **Custodian:** [Your Name]
 - **Timestamp (UTC):** [Output of `date -u`]
 - **Notes/Action Taken:** Acquired from Digital Corpora via wget; official hash file acquired from the course portal.
- *Observation/Expected Outcome:* The log’s first entry establishes when the evidence came into your possession and from where.

4.3 Step 3: Verify the Integrity of Original Evidence

This step proves that the file on your disk is a bit-for-bit copy of the file IT imaged.

1. Calculate the SHA-256 Hash of the Downloaded Image

- **GUI Instructions:** This is a command-line task; there is no reliable GUI equivalent.
- **CLI Instructions:** Run `sha256sum` on the image. It finishes in under a second for a file this size.

```
1 sha256sum charlie-work-usb-2009-12-11.E01
2
```

- *Observation/Expected Outcome:* A 64-character hexadecimal string followed by the filename.

2. Compare the Calculated Hash with the Official Hash

- **GUI Instructions:** Open `charlie-work-usb-2009-12-11.E01.sha256` in a text editor and compare it, character by character, with the value you calculated.
- **CLI Instructions:** Let `sha256sum` do the comparison. The `-c` flag reads the expected hash from the file, recomputes the image’s hash, and prints one word.

```
1 sha256sum -c charlie-work-usb-2009-12-11.E01.sha256
2
3 # To see both values side by side:
4 cat charlie-work-usb-2009-12-11.E01.sha256
5
```

- *Observation/Expected Outcome:* `charlie-work-usb-2009-12-11.E01: OK`. If it prints `FAILED`, stop: the file on your disk is not the evidence. Delete it, download it again, and repeat this step. Do not carry on with a copy that does not match.

3. Log the Verification Step

- **GUI/CLI Instructions:** Add a new entry for the same **Item ID**, with a fresh **date -u**.
 - **Notes/Action Taken:** Calculated SHA-256 hash. Value: [the calculated hash]. Verified against official hash file `charlie-work-usb-2009-12-11.E01.sha256` with `sha256sum -c`. Result: OK.
- *Observation/Expected Outcome:* The log now carries cryptographic proof that the evidence was unaltered at the time of acquisition.

4.4 Step 4: Create and Log a Working Copy

Never work on the original evidence. This step creates the duplicate you will analyse.

1. Create the Working Copy

- **GUI Instructions:** In your file manager, copy the original image, paste it into the same folder, and rename the copy `charlie-work-usb-2009-12-11-WORK.E01`.
- **CLI Instructions:** Use `cp`.

```
1 cp charlie-work-usb-2009-12-11.E01 charlie-work-usb-2009-12-11-WORK.  
   E01  
2
```

- *Observation/Expected Outcome:* A third file in the directory, `charlie-work-usb-2009-12-11-WORK.E01`.

2. Log the Creation of the Working Copy

- **GUI/CLI Instructions:** Add a *new item* to the log. The working copy is a distinct evidence item derived from the original.
 - **Case ID:** M57-Patents
 - **Item ID:** M57-USB-Work
 - **Item Description:** Working copy of the USB image (`charlie-work-usb-2009-12-11-WORK.E01`)
 - **Custodian:** [Your Name]
 - **Timestamp (UTC):** [Output of `date -u`]
 - **Notes/Action Taken:** Created working copy from original evidence item M57-USB-Orig with `cp`.
- *Observation/Expected Outcome:* The log shows a new item and names the item it was derived from.

4.5 Step 5: Verify the Integrity of the Working Copy

The final step proves that the working copy is a perfect duplicate of the original.

1. Calculate the SHA-256 Hash of the Working Copy

- **GUI Instructions:** Again a command-line task.
- **CLI Instructions:** Run `sha256sum` on the copy.

```
1 sha256sum charlie-work-usb-2009-12-11-WORK.E01  
2
```

- *Observation/Expected Outcome:* The working copy's hash.

2. Compare the Working Copy Hash to the Original Hash

- **GUI/CLI Instructions:** Compare it with the hash of the *original* from Step 3. The values must be identical. One command shows both:

```
1 sha256sum charlie-work-usb-2009-12-11.E01 charlie-work-usb-2009-12-11-  
   WORK.E01  
2
```

- *Observation/Expected Outcome:* Two lines, the same 64 characters on each. That match is what makes the copy a forensically sound duplicate.

3. Log the Final Verification

- **GUI/CLI Instructions:** Add a final entry for the working copy item.
 - **Notes/Action Taken:** Calculated SHA-256 hash. Value: [the working copy's hash]. Verified identical to original evidence item M57-USB-Orig. Match confirmed. Ready for analysis.
- *Observation/Expected Outcome:* The log is complete for the acquisition and preparation phase. It tells a verifiable story of how the evidence was handled, with a signature, a time and a reason for every action.

5 Lab Report

Complete the following report from the actions you performed and the log you kept. It simulates the initial report you would file for the case.

Cybercrime Investigation - Preliminary Analysis Report

Case ID: LAB-03-Integrity-[Your-Name]

Investigator: [Your Name]

Date of Analysis: [Today's Date]

1. Evidence Details

- **Filename:** charlie-work-usb-2009-12-11.E01
- **Size in bytes:** [From `ls -l`]
- **SHA-256 Hash:** [The hash you calculated for the original evidence file]
- **Source and time of acquisition (UTC):** [From your log]

2. Executive Summary

[Two or three sentences describing the procedure you followed: acquisition of the M57-Patents USB image, SHA-256 verification against the issued hash, creation of a working copy, and the Chain of Custody log kept throughout.]

3. Key Findings: Procedural Integrity Summary

- **Original Evidence Integrity:** [State whether the calculated SHA-256 hash of charlie-work-usb-2009-12-11.E01 matched the official value, and what that establishes about the file's condition since IT imaged it.]
- **Working Copy Creation:** [State that a working copy was created from the verified original, and why the original is then left untouched.]
- **Working Copy Integrity:** [State how you showed the working copy is a true duplicate, and what a mismatch would have meant.]

4. Importance of Procedural Integrity

[Explain why these steps are fundamental to a digital investigation. Discuss the consequences of a mismatched hash or an undocumented action: which admissibility requirement from Chapter 3 each failure would breach, and how a defence lawyer would use the gap in cross-examination.]