

Technical Forensic Report & Handover Notes

Disk Image Investigation (Class Project)

Report date: 11 March 2026

Project scope: Class-wide investigation (14 students)

Class lead: Alexis

Primary custodian: Marius

Team structure: 4 teams (Team 1–4), each with a team lead

Team 1: *Mathis, Mathys, Rajesh*

Team 2: *Marius, Kevin, Reza, Mehdi*

Team 3: *Daniel, Alexis, Imram*

Team 4: *Amélie Servant, Ornedá Kamberi, Sadik, Sayem*

Purpose of this document: explain, step-by-step, what was done from the beginning, what was validated, what was discarded, where files are stored, and exactly how to continue safely.

1) Executive Summary (What we proved)

We analyzed a segmented RAW/dd forensic image (original.001 ... original.159, ~238 GB). After partition mapping and reproducible extraction using The Sleuth Kit, we identified that partition P5 did not contain a normal NTFS structure nor BitLocker signatures. We treated it as a potential VeraCrypt/TrueCrypt raw encrypted volume, successfully mounted it read-only using VeraCrypt with the password nitheesh123hseehtin, and found a directory containing multiple files consistent with stolen/exfiltrated data (financial data, customer database dump, bank document image, spreadsheet, email conversation screenshot, meeting video). Evidence files were exported to an exhibits directory and hashed individually with SHA-256 for integrity.

Sensitive content is not reproduced in this report. Evidence is referenced by Exhibit ID + SHA-256.

2) Investigation Context & Team Workflow

- The teacher provided a hard drive to the class.
- The class (14 people) is divided into 4 teams for parallel work.
- Alexis coordinates investigation tasks at the class level (class lead).
- Marius is the primary custodian (custody/handling oversight).
- Amélie (Team 4 lead), my work is relevant to the whole class because it identifies and preserves key evidence on P5.

3) Timeline (What happened when)

- **Sun 22/02/2026:** The original disk was write-protected; forensic acquisition performed using FTK Imager (image creation).
- **Tue 24/02/2026:** We downloaded the ZIP, extracted it, and began analysis on the image (not on the original disk).
- **Mon 02/03/2026 – Wed 04/03/2026:** Partition mapping with TSK, extraction of key partitions, identification of encrypted volume P5, VeraCrypt mount (read-only), evidence export and hashing.
- **Thu 05/03/2026 – Sun 08/03/2026:** Redaction of the report

4) Forensic Methodology (Why these steps)

We follow standard digital forensics flow:

1. **Preserve:** work on the image only; avoid writing; record hashes.
2. **Identify:** map partitions; determine file systems; detect encryption.
3. **Acquire/Extract:** reproducibly extract partitions/volumes for focused work.
4. **Analyze:** mount/decrypt volumes in read-only mode; locate relevant artifacts.
5. **Preserve Evidence:** export evidence, hash per file, log metadata.
6. **Document:** record commands, paths, dates, and results for the final report.

5) Inputs, Storage Layout, and “Do Not Touch” Rules

5.1 Inputs (read-only conceptually)

- Segmented RAW image:
evidences/original.001 → evidences/original.159
- Acquisition logs/ hashes:
evidences/original.001.txt

5.2 Where outputs are stored

- Extracted volumes/containers:
evidences/containers/
- Hash logs for extracted volumes:
evidences/hashes.txt
- Evidence exports (“exhibits”):
evidences/exhibits/p5_vc/
- Hashes for evidence files:
evidences/hashes_exhibits.txt
- Metadata for evidence files:
evidences/exhibits/p5_vc/stat.txt

5.3 Important handling rule

- Any mounts of encrypted volumes must be read-only.
- Evidence files must be copied using `cp -a` (preserve timestamps) and hashed immediately.

6) Step-by-Step: What we did from the beginning

Step 1: Verify partition structure (TSK)

Goal: understand where data lives and locate suspicious areas (unallocated space, encrypted partitions).

Command:

Title : Partition Mapping (TSK)

```
mmls /evidences/original.001
```

Displays the disk image partition layout (start sector, length, type). Used to precisely identify which TSK entry corresponds to each partition/volume before extraction.

Key output interpretation:

- **P1:** NTFS (UnionSine), visible accessible partition (contains hint scripts).
- Large unallocated region (massive gap), suspicious (possible hidden volumes/data remnants).
- **P3/P4:** FAT32 partitions (one was initially suspected BitLocker-related).
- **P5:** NTFS/exFAT partition entry, but later extraction appeared structureless (“data”) → suggests encryption.

Relevant table:

Slot	Identifier	Start	Length	Type/FS
002	000:000	2048	47331328	NTFS (P1)
003	—	47333376	335646720	Unallocated
006	001:000	382980096	2045952	FAT32 (P3)
007	000:002	385026048	24573952	FAT32 (P4)
009	000:003	409602048	78790656	NTFS/exFAT (P5)

Naming convention used in this report:

Some tools (FTK/Windows) display only user-visible volumes, while TSK (*mmls*) lists metadata entries, logical partitions, and unallocated gaps.

To keep this report readable, we use internal labels (P1, P3, P4, P5).

P5 (internal label) refers to the partition entry TSK Slot 009 (identifier **000:003**, start sector **409602048**, length **78790656** sectors).

This is the encrypted volume later mounted with VeraCrypt at */mnt/vc_p5*.

Step 2: Extract partitions reproducibly (mmcat)

Goal: work on clean partition-level files, avoid tool confusion, and make hashing easier.

Critical point: some earlier extraction attempts produced questionable files.

We corrected this by extracting using the exact TSK identifier shown by mmls (e.g., 000:003).

Extract P5 (Slot 009 / 000:003)

Command:

Title: Partition Extraction (TSK → Standalone Volume File)

```
mmcat -i raw /evidences/original.001 000:003 > evidences/containers/BDE_H_FIXED.vol
```

Extracts the partition/TSK entry 000:003 from the RAW image and writes it to a standalone file. This creates a reproducible “partition image” for focused analysis without touching the full disk image.

Title: Integrity Hashing of Extracted Volume (SHA-256)

```
sha256sum evidences/containers/BDE_H_FIXED.vol | tee -a /evidences/hashes.txt
```

Computes the SHA-256 hash of a file to preserve integrity and appends it to a hash log. tee -a prints the hash to the terminal while also saving it to hashes.txt.

Result:

- A ~38 GB file was created and hashed.
- This extraction is now the reference for P5 analysis: BDE_H_FIXED.vol.

Step 3: Validate whether P5 is BitLocker or a normal filesystem

Goal: confirm or reject BitLocker/NTFS hypotheses using quick, objective tests.

Tests performed:

- Search for BitLocker signature FVE-FS across the extracted volume → no occurrences.
- file -s on the extracted volume → returns data, not NTFS/FAT/BitLocker.
- dislocker-metadata was attempted and stalled (consistent with “not BitLocker”).

Command:

Title: Volume Type Identification (Signature-Based Check)

```
file -s /evidences/containers/BDE_H_FIXED.vol
```

Attempts to identify the real type of the extracted volume by inspecting file signatures (filesystem/encrypted data). The -s option forces inspection even if the file appears as raw data.

Title: BitLocker Metadata Verification

```
dislocker-metadata -V /evidences/containers/BDE_H_FIXED.vol
```

Tries to parse BitLocker metadata from the volume. If metadata cannot be read (no output or tool stalls), it supports the conclusion that the volume is not BitLocker-encrypted.

Conclusion:

- P5 is not BitLocker.
- P5 behaves like a raw encrypted container/partition (VeraCrypt/TrueCrypt plausible because the header is encrypted and doesn't show clear signatures).

Step 4: Treat P5 as VeraCrypt and attempt read-only mount

Goal: confirm if P5 is a VeraCrypt volume using known credentials from the investigation.

Mount details:

- VeraCrypt password: nitheesh123hseehtin
- Mount point: /mnt/vc_p5
- Read-only mode: enabled

Command:

Title: VeraCrypt Read-Only Mount (Decryption Without Alteration)

```
sudo veracrypt --text --non-interactive \  
--mount /evidences/containers/BDE_H_FIXED.vol /mnt/vc_p5 \  
--password "nitheesh123hseehtin" --pim 0 --protect-hidden=no --readonly
```

Mounts the encrypted volume using VeraCrypt via CLI in non-interactive mode. --readonly ensures no writes occur, preserving forensic integrity; --pim 0 uses the default PIM; --protect-hidden=no disables hidden-volume protection prompts.

Result:

- Mount succeeded.
- The presence of a normal directory structure under /mnt/vc_p5/ confirms the correct decryption/mount.

Step 5: Identify relevant evidence inside the mounted volume

Goal: locate files matching the “data theft” scenario.

Location: /mnt/vc_p5/New folder/

Evidence found (high-level):

- Text file containing highly sensitive financial information
- SQL dump containing customer data
- Bank statement image
- Spreadsheet (requires review)
- Screenshot of email conversation
- Meeting video

There were also many empty “New folder (X)” directories, likely intended to dilute/hide the real evidence files.

Step 6: Preserve evidence: export + per-file hashing + metadata

Goal: make exhibits usable in court-style reporting: stable copies + hashes + timestamps.

Export destination: /home/alexis/evidences/exhibits/p5_vc/

Export command:

Title: Evidence Export (Preserve Timestamps and Attributes)

```
cp -a "/mnt/vc_p5/New folder/Nitheesh_Corporation.txt" \  
  "/mnt/vc_p5/New folder/customer data.sql" \  
  "/mnt/vc_p5/New folder/Barclays_Statement.jpg" \  
  "/mnt/vc_p5/New folder/20231113151155_5869.xls" \  
  "/mnt/vc_p5/New folder/mail-conversation.png" \  
  "/mnt/vc_p5/New folder/meeting2.mp4" \  
  /evidences/exhibits/p5_vc/
```

Copies the evidence files from the mounted volume into a dedicated exhibits directory.

The -a (archive) option preserves timestamps and file attributes, which is important for timeline reconstruction and reporting.

Hashes + metadata:

Title: Switch to Exhibits Working Directory

```
cd/evidences/exhibits/p5_vc
```

Changes the current working directory to the exhibits folder so that hashing and metadata collection are performed consistently on the exported evidence copies.

Title: Per-Exhibit Integrity Hashing (SHA-256)

```
sha256sum * | tee -a /home/alexis/evidences/hashes_exhibits.txt
```

Computes SHA-256 hashes for each exported evidence file and appends them to hashes_exhibits.txt. These hashes are referenced in the Evidence/Exhibits table.

Title: Evidence Metadata Capture (Filesystem Timestamps and Attributes)

```
stat * > stat.txt
```

Records file metadata (size, permissions, and MAC times: Modify/Access/Change) for each exhibit into stat.txt, supporting timeline analysis and documentation.

7) Evidence Table (Exhibits)

Exhibit ID	File	Size	Original path	Export path	SHA-256	Relevance
Ex01	Nitheesh_Corporation.txt	706 B	/mnt/vc_p5/New folder/Nitheesh_Corporation.txt	~/evidences/exhibits/p5_vc/Nitheesh_Corporation.txt	0f9663a1c727f18ca87893c62f205c4c391ca03a8b15f0cd2c5610eb22ae8e0f	Contains highly sensitive financial data → direct evidence of theft/exfiltration.
Ex02	customer data.sql	363 K	/mnt/vc_p5/New folder/customer data.sql	~/evidences/exhibits/p5_vc/customer data.sql	3f130cb3b2a1ea3ca371766c9bc9047e9cb614f9452d274d3d7ff97ceb338896	Customer database dump → structured customer info likely exfiltrated.
Ex03	Barclays_Statement.jpg	433 K	/mnt/vc_p5/New folder/Barclays_Statement.jpg	~/evidences/exhibits/p5_vc/Barclays_Statement.jpg	abd8af4dbc1f055c0e9aaa409a105f2a5caf70acbf420b82358d2bafbdad1ade4	Bank statement image → confidential banking document.
Ex04	20231113151155_5869.xls	14 K	/mnt/vc_p5/New folder/20231113151155_5869.xls	~/evidences/exhibits/p5_vc/20231113151155_5869.xls	700e1ea19a397e6b7c3ee48c37ccb6209fd1bf4c3ef2456ee301a570f7f3c11b	Spreadsheet to analyze (sheets/values/formulas).
Ex05	mail-conversation.png	36 K	/mnt/vc_p5/New folder/mail-conversation.png	~/evidences/exhibits/p5_vc/mail-conversation.png	3cce569837edd0539de4510fc656a07ac3b33a1a6ae4cc9b600960ebc86a1f23	Email conversation screenshot → contextual communication evidence.
Ex06	meeting2.mp4	3.8 M	/mnt/vc_p5/New folder/meeting2.mp4	~/evidences/exhibits/p5_vc/meeting2.mp4	d72911e9584f9dae184deb92204dc8aab355deeb7f7149d5290cf714f6da96cb	Meeting recording → multimedia context evidence.

8) Evidence analysis

The collected evidence indicates that the encrypted container stored multiple types of sensitive data, including customer financial records, banking documents, communication artifacts, and multimedia evidence. The presence of both raw database dumps and structured spreadsheets suggests that the stolen data was processed and organized after extraction.

Exhibit ID	File concerned	Person involved	Stolen data (detailed)	Description / Relevance
Ex01	Nitheesh_Corporation.txt	Nitheesh	Corporate financial records, transaction references, internal financial information	Text document containing sensitive corporate financial information that may indicate unauthorized extraction of financial records.
Ex02	customer data.sql	Damon Haley; Bayley Messner; Conley Tijerina; Aela Rayford; Cypress Kain; Vinh Casiano	Customer database including: full names, payment card identifiers, expiration dates, internal customer records	SQL database dump containing structured customer records that appear to include financial and personal data.
Ex03	Barclays_Statement.jpg	John Doe Lee	Banking information including account statement details, transaction history, account holder identification	Image of a Barclays bank statement that may reveal account activity and confidential banking information.
Ex04	20231113151155_5869.xls	Damon Haley; Bayley Messner; Conley Tijerina; Aela Rayford; Cypress Kain; Vinh Casiano	Structured financial dataset including: customer names, payment card references, expiration dates, possible card issuer fields, internal financial summaries	Spreadsheet organizing multiple customer financial records, likely derived from the SQL database dump for easier analysis or distribution.
Ex05	mail-conversation.png	Josh; Nitheesh	Email communication data including	Screenshot of an email exchange suggesting

			conversation context related to bank detail verification	potential phishing activity or coordination regarding financial information.
Ex06	meeting2.mp4	Ken Morris, Cr Lou Brown, Cr Andrew Brown, Cr Susan O'Regan, Mayor Jim Mylchre..., Cr Claire St Pierre, Cr Grahame Webbe., Roger Gordon, Cr Bruce Thomas, Marcuq Gower, Cr Mike Pettit, Cr Elwyn Andree-..., Liz Stolwyk, Philip Coles	Audio-visual discussion potentially referencing financial data handling or data transfer	Video recording of a meeting.

SQL database:

File Size	363 KB (3,193 lines)
Format	SQL Server dump, UTF-8 with BOM
Date Range	July 2012 - May 2014

Tables:

- Customer: 91 records (names, cities, countries, phones)
- Supplier: 29 records (companies, contacts)
- Product: 78 items (\$2.50 - \$263.50)
- Order: 830 transactions (\$1.35M total revenue)
- OrderItem: 2,155 line items

Geographic Spread:

- 21 countries (USA: 13, Germany: 11, France: 11)

Findings:

- Customer ID 44: Renate Messner (Frankfurt a.M., Germany)
- Only linkage to credit card file (shared surname with Bayley Messner)

Classification: Business database (Northwind sample data), contains Personally Identifiable Information

9) Key Conclusions So Far

- P5 is not BitLocker and not a normal NTFS volume in raw form.
- P5 was successfully decrypted/mounted with VeraCrypt (read-only), confirming it is an encrypted container/partition.
- Evidence strongly indicates storage of exfiltrated sensitive data (financial + customer DB + supporting artifacts).
- Evidence was preserved with exports and SHA-256 hashes, ready for deeper analysis and correlation.

10) Perspectives

9.1 Validate mount and evidence (quick reproducibility check)

1. Mount P5 read-only (same VeraCrypt command).
2. Confirm files exist under /mnt/vc_p5/New folder/.
3. Verify exported evidence hashes match current exports (SHA-256).

9.2 Deep Analysis Tasks (Recommended Division - Detailed)

Team 1: customer data.sql (Database dump analysis)

Objective: determine what data was exfiltrated (scope, structure, number of records) and extract high-level indicators without reproducing sensitive values.

Tasks

1. **Identify SQL type and schema**
 - Confirm if the file is a full dump, partial dump, or a single table export.
 - Extract table names and column names (no content values).
2. **Estimate scale**
 - Count the number of INSERT statements per table (approx. record count).
 - Identify whether the dump contains PII fields (names, phones, addresses, payment references, etc.).
3. **Timeline indicators**
 - Look for timestamps inside the dump (e.g., created_at, last_update) if present.
4. **Integrity + traceability**
 - Work only on the exported exhibit copy: ~/evidences/exhibits/p5_vc/customer data.sql
 - Reference **Ex02 + SHA-256** in the report.

Deliverables

- Table list + column list (high level)
- Estimated record counts
- “Data category mapping” (PII types found)
- 3-5 key screenshots/logs of commands output

Team 2: 20231113151155_5869.xls (Spreadsheet content + structure)

Objective: understand whether the spreadsheet aggregates stolen data, references the SQL/text file, or contains additional sensitive content.

Tasks

1. **File type validation**
 - Confirm it is truly XLS and not a disguised file (magic bytes / file identification).
2. **Extract spreadsheet structure**
 - List sheet names.
 - Determine which sheets contain data (non-empty ranges).
3. **Look for relationships**
 - Search for keywords that may link it to the SQL dump or financial data (e.g., “customer”, “card”, “statement”, “export”, “SQL”).
4. **Check formulas and external links**
 - Identify formulas referencing other files or paths.
 - Identify any embedded objects or hidden sheets.
5. **Preserve reporting rules**
 - Do not paste raw sensitive cells in the report; summarize categories and counts.

Suggested tools

- LibreOffice (offline VM) or ssconvert / xls2csv if available.

Deliverables

- Sheet list + brief description per sheet
- Summary of what data categories appear
- Presence/absence of links to other files
- Screenshots of structure (not raw sensitive data)

Team 3: Multimedia evidence (JPG / PNG / MP4 metadata + context)

Objective: extract metadata (timestamps, origin) and interpret what the media shows, without altering files.

Targets

- Ex03: Barclays_Statement.jpg
- Ex05: mail-conversation.png
- Ex06: meeting2.mp4

Tasks

1. **Metadata extraction**
 - For images: EXIF (if present), creation/modification timestamps, software used.
 - For MP4: duration, codec, creation time (container metadata).
2. **Context summary**
 - Briefly describe what is depicted (e.g., “bank statement screenshot”, “email conversation screenshot”) without quoting sensitive text.
3. **Timeline correlation**
 - Compare metadata timestamps to file system timestamps in stat.txt and to the “Nov 2023” cluster.
4. **Integrity**
 - Use only exhibit copies and keep them read-only.

Deliverables

- Metadata table (file → key metadata fields)
- A short narrative summary (what the media supports in the case)
- A timeline snippet (timestamps + interpretation)

Team 4: Correlation with P1 (“UnionSine”) + timeline building

Objective: connect the encrypted volume and stolen files to activity on P1 (hints, scripts, user activity), and build a coherent incident timeline around November 2023.

Tasks

1. Identify hint sources

- Locate and summarize HintsForThePassword.bat and any other hint scripts on P1.
- Document where the VeraCrypt password came from (high level).

2. Look for “exfil preparation” artifacts

- Search for archive tools (zip/7z/rar), staging folders, suspicious naming (“New folder” patterns), cleanup traces (\$RECYCLE.BIN usage).

3. Build a timeline

- Use file timestamps from stat.txt (exhibits) + any relevant timestamps from P1 artifacts.
- Highlight key moments: encryption container creation → file placement → potential cleanup.

4. Handover readiness

- Provide exact paths, commands, and checks so any teammate can reproduce results.

Deliverables

- Timeline table (date/time → event → supporting artifact)
- Correlation notes linking P1 hints to VeraCrypt mount success
- A “reproducibility checklist” (commands + expected outputs)