

DF6: Fergus File Carving Report

Course: CS573 / Introduction to Digital Forensics

Assignment: DF Assignment 6 — Fergus File Carving

Prepared: July 25, 2026, 20:46 PDT

DRAFT FOR REVIEW — not submitted to Canvas

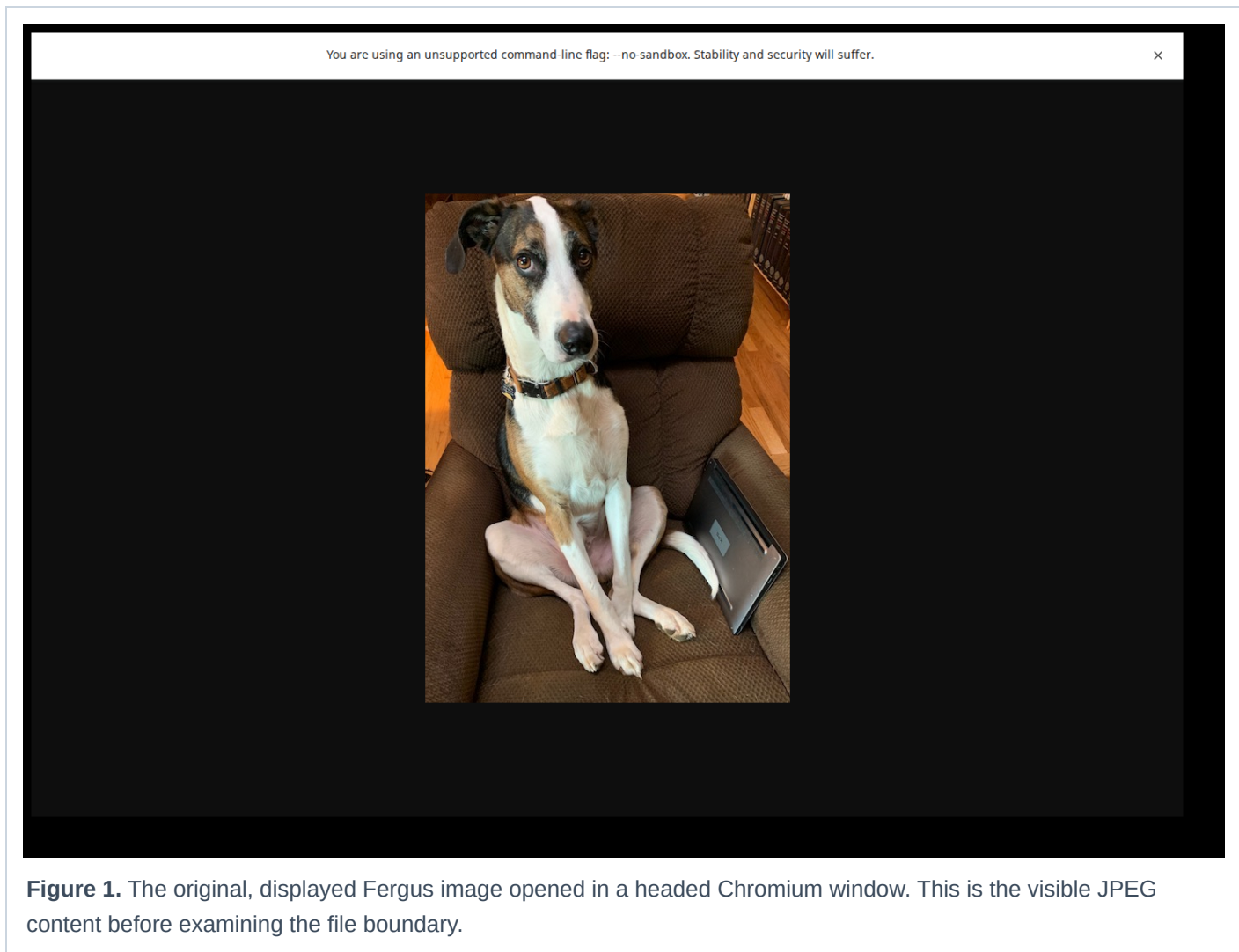
1. Purpose

The purpose of this exercise was to inspect the provided Fergus image for embedded files, identify file boundaries using hexadecimal file signatures, carve any hidden data, and validate the resulting output.

2. Tools and Process

1. I downloaded the course-provided source file, `image003.jpg`, and preserved it as the original evidence file.
2. I viewed bytes near the end of the file using `xxd` in a headed GNOME Terminal window. I used the JPEG end-of-image signature `FF D9` and the PDF header `%PDF-` to identify the boundary.
3. I carved the byte range beginning at offset 310,299 through the end of the source file into a separate PDF.
4. I validated the carved file with `file`, `pdfinfo`, and text extraction. I also opened both the source image and the recovered PDF in headed Chromium windows.

3. Evidence Screenshots



```

File Edit View Search Terminal Help
DF6 Fergus | Actual xxd output from the source file
JPEG EOI is FF D9 at offsets 310297-310298; PDF begins at 310299.

0004bbe0: 35 6d 47 c2 be 22 d1 ac fc 39 77 2e 8f 6b 75 e0 5mG..."9w..ku.
0004bbf0: 8f 0b 5e dc 5b d9 ff d9 15 9c a7 ..^.[.m.)....e..
0004bc00: dd 12 48 11 4c 85 70 1d f2 e4 17 67 66 b5 ee e8 ..H.L.p....gf...
0004bc10: ae be 60 a0 ac b4 e8 ba 9f ff d9 25 50 44 46 2d ..'.....%PDF-
0004bc20: 31 2e 33 0a 25 c4 e5 f2 e5 eb a7 f3 a0 d0 c4 c6 1.3.%.....
0004bc30: 0a 34 20 30 20 6f 62 6a 0a 20 20 20 20 20 20  obj.<< /Len
0004bc40: 67 74 68 20 35 20 30 20 52 PDF header: %PDF- 5 0 R /Filte

Red annotation will identify: FF D9 | %PDF-
ubuntu@zero:~$

```

Real headed GNOME Terminal capture of xxd output; red boxes added afterwards only to annotate the verified carving bytes.

Figure 2. Real headed GNOME Terminal capture of xxd output from the source file. The red boxes were added after capture only to identify the actual FF D9 JPEG end marker and the following %PDF- header used for carving.

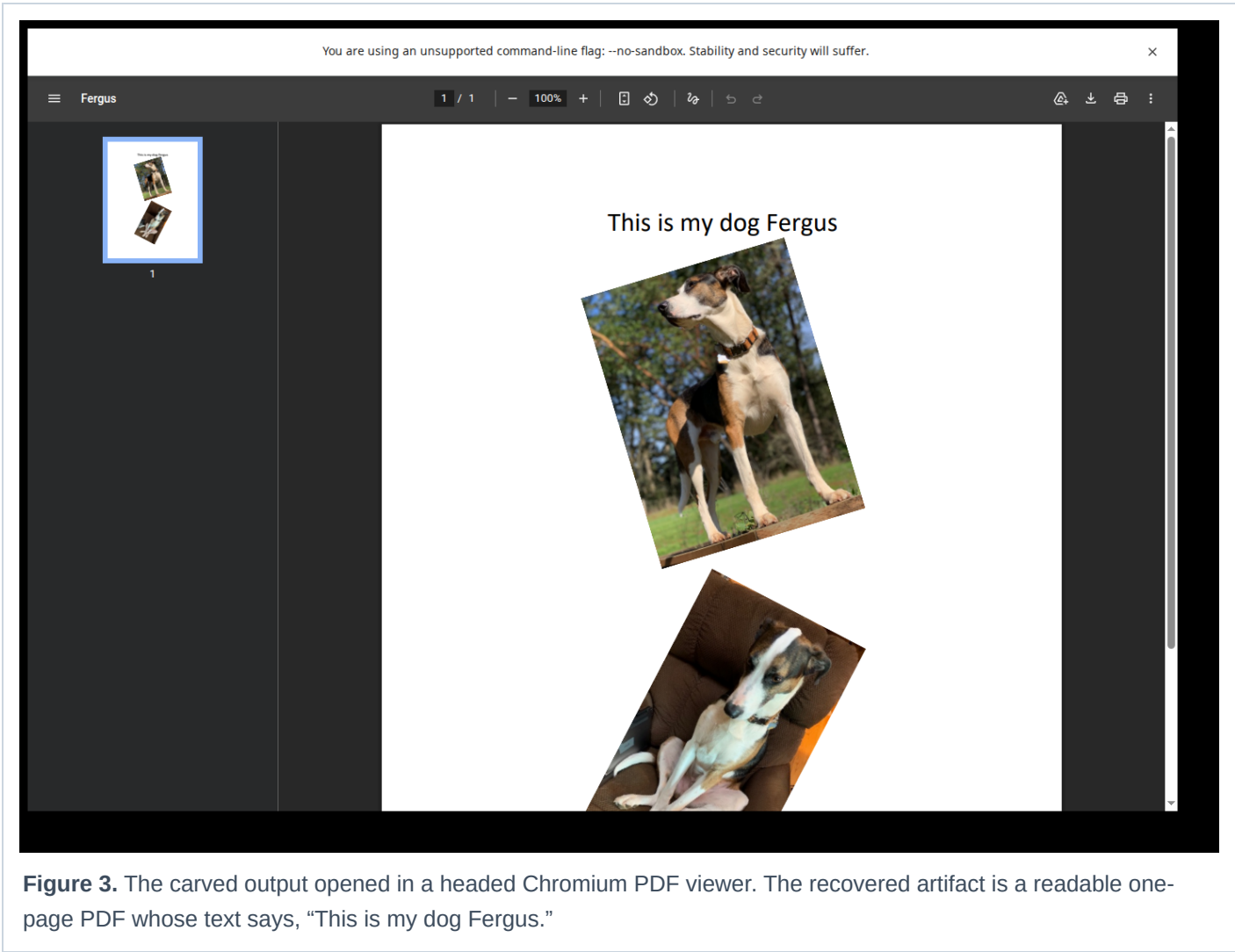


Figure 3. The carved output opened in a headed Chromium PDF viewer. The recovered artifact is a readable one-page PDF whose text says, “This is my dog Fergus.”

4. File Boundary and Carving Result

The original file is a valid JPEG. The JPEG end-of-image marker `FF D9` occurs at zero-based offsets 310,297–310,298 (hex `0x0004BC19–0x0004BC1A`). Immediately afterward, at offset 310,299 (hex `0x0004BC1B`), the bytes begin with the PDF header `%PDF-`. This shows that a PDF was appended after the normal end of the JPEG.

Artifact	Size	Detected type / observed data
Original source: image003.jpg	523,648 bytes	JPEG/JFIF image, 437 × 611 pixels. The source contains the visible image followed by appended PDF data.
Visible JPEG portion	310,299 bytes	Valid JPEG/JFIF image. It opens normally and contains the displayed Fergus photograph.

Carved hidden file	213,349 bytes	Valid PDF 1.3, one page, not encrypted. Its PDF text is “This is my dog Fergus.” PDF metadata identifies the title as <i>Fergus</i> and subject as <i>Fergus demonstrating steganography</i> .
--------------------	---------------	--

The original source SHA-256 was

e18a17b9b5caf28ae3b81fb53aab218f29f90a04dc64934344b7c5d390b943d1 . The carved PDF SHA-256 was 379677da19d9c2b463bd4f357f7252a3613d8e5c08e07033f859df5a9066ce6b .

5. Conclusion

The file signatures establish that the original JPEG was extended with a complete PDF file. Carving from the PDF header at offset 310,299 through the end of the source produced a valid, readable PDF. The result is consistent with file concatenation used to conceal data after a normal JPEG end marker. I do not infer intent from the appended data alone.

BTW: Fergus is really cute!

This report is the primary submission candidate because it contains the screenshots, carving evidence, process, tool description, and file comparison requested by the rubric. The recovered PDF may also be attached as a supporting artifact if Canvas permits multiple file uploads.