Project 1 Documentation - Andrew Chan

1. Objective

The objective of this project is to perform a forensic workflow starting with a VirtualBox disk image, .VDI, o produce a verified forensic image, .E01, and to document the integrity verification at each step. The goal is to presere evidence integrity while creating working and forensic copoies for analysis

2. Environment and Tools

OS: Windows (PowerShell) **Hypervisor Image Format:** VirtualBox (.vdi) **Tools Used:**

- PowerShell `get-filehash`
- QEMU `qemu-img`
- EWT Tools `ewfacquire`, `ewfverify`

All of these commands were executed locally within the virtual machine through HackTheBox

3. Evidence Intake

The original evidence file provided was a WirtualBox disk image.

- **Filename:** `FreeBSD_Forensics_PartI.vdi`

A SHA-256 hash was calculated to create a baseline integrity value before any changes were made

Command Used: `get-filehash ./FreeBSD_Forensics_PartI.vdi -algorithm sha256`

Result:

- SHA-256 hash created and stored within a new file named `FreeBSD_Forensics_PartI.sha256sum`

This hash will then serve as a reference value for the original evidence

4. Conversion to RAW

To create forensic tooling compatibility, the VDI image will be converted into a RAW disk image using QEMU.

Command Used: `C:\"Program Files"\qemu\qemu-img.exe convert -p -f vdi -O raw ./FreeBSD_Forensics_PartI.vdi ./FreeBSD_Forensics_PartI.raw`

The RAW image represents a sector by sector disk layot which is suitable for forensic usage.

5. RAW Disk Validation

The RAW image was inspected and hashed to confirm successful conversion

Disk Information: `C:\"Program Files"\qemu\qemu-img.exe info ./FreeBSD_Forensics_PartI.raw`

Checksum Calculation: `get-filehash .\FreeBSD_Forensics_PartI.raw -algorithm sha256 > .\FreeBSD_Forensics_PartI-RAW.sha256sum` **Result:**

- SHA-256 hash recorded and stored in a new file named `FreeBSD_Forensics_PartI-RAW.sha256sum`

The SHA-256 hash of the RAW image is different from the VDI hash. This makes sense because there is differences in file format and metadata layout.

6. Forensic Image Creation (E01)

The RAW image was acquired into the EWF (E01) using `ewfacquire`, which produced a forensic image with embedded metadata and integrity checks.

Command Used: `ewfacquire ./FreeBSD_Forensics_PartI.raw`

Acquisition Metadata: Examiner: Andrew Case Number: 1 Evidence Number: 1 Media Type: Fixed Disk Image Format: EnCase 6 (.E01) Compression: Deflate Sector Size: 512 bytes

The acquisition completed successfully without any errors.

7. Forensic Image Verification

The integrity of the E01 image was verified using `ewfverify`

Command Used: `ewfverify FreeBSD_Forensics_PartI.E01`

Result: MD5 hash stored in E01: `a37fd5124513584ef27e421f37e522f1` MD5 hash calculated during verification: `a37fd5124513584ef27e421f37e522f1` Verfication Stats: SUCCESS

This confirms that the forensic image has not been altered since acquisition.

8. Explanation of Hash Differences

The SHA-256 hashes of the `.vdi` and `.raw` files are difference because the conversion process changes the file structure, metadata, and block layout. However, the forensic integrity is preserved because the E01 image embeds its own hash values, which we saw were successfully verified using `ewfverify`.

9. What I learned

This project helped me understand the concepts of general forensics and how we can use hasing algorithms like SHA-256 as a checksum. I also learned things like Evidence intake, working copies, and forensic images. One big takeway is that forensic validity is created through repeatable verification rather than just matching hashes across different formats. Tools like `ewfverify` are able to provide stronger

guarantees rather than simple file hashing by validating embedded integrity data.

Some things that I learned that aren't related directly to the project is how the Windows filesystem paths differe from Linux. Windows uses volume specific root paths like `C:\` rather than a single unified root. Also, I learned that PowerShell doesn't rely on the ~ home directory convention in the same way that Linux does.

10. Conclusion

All required steps for forensic imaging and preparation were completed successfully. The evidence pipeline preserved integrity at every stage which resulted in a verified forensic image that is suitable for further analysis in tools like Autopsy.