Project 4: Forensics

This project is due on **Monday, December 13th** at **5 p.m.** and counts for 9% of your course grade. Late submissions will be penalized by 10% plus an additional 10% every 12 hours until they're received. The professor may grant individual extensions, but only under truly extraordinary circumstances. As always, *we recommend that you begin early.*

This is a group project; you will work in **groups of two** and submit one project per group. If you wish, you may work with a different partner than you worked with for the previous project.

You are bound by the Honor Code not to communicate with anyone outside your group and the courses staff regarding your solutions to this project. The number of pieces of evidence you find, the techniques you try, how successful said techniques are, the general progress you follow, etc. are considered part of your solution and must not be shared between groups.

Start early. It may be impossible to complete this project before the deadline if you do not begin several days beforehand. Please plan accordingly.

1 Introduction

In this project, you will investigate the murder of Hapless Victim, a leading figure in the university community who was killed while working in the EECS building sometime between midnight and 6 a.m. Photos of the crime scene are bogged down with paperwork, but officers recovered the projectile shown below (Exhibit A), which appears, inexplicably, to have been the cause of death.

Officers have arrested the leading suspect, Nefarious Criminal, and seized his computer. An image of his hard drive is available at http://www.eecs.umich.edu/courses/eecs398/forensics_release.vdi. Your job is to conduct a forensic examination of his computer and document any evidence relating to the murder.

Objectives:

- Understand how computer use can leave persistent traces and why such evidence is often difficult to remove or conceal.
- Gain experience applying the security mindset to investigate computer misuse and intrusion.
- Learn how to retrieve information from a disk image without booting the operating system, and understand why this is necessary to preserve forensic integrity.

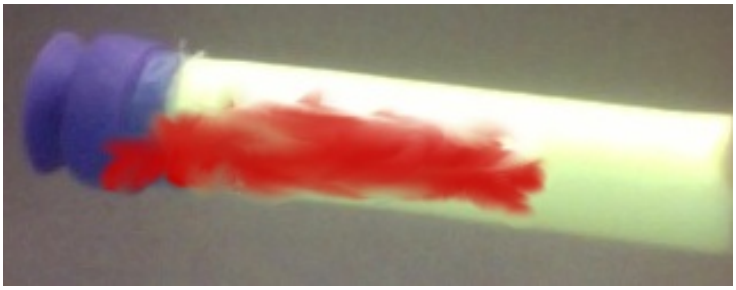

Exhibit A — Projectile recovered at the crime scene. Ballistics experts have identified it as a "Nerf blaster dart."

2 Tasks and Deliverables

The deliverables for this project are your answers to the prompts. Unless otherwise indicated, your answers should be *complete* but *concise*. None of the questions should require more than 1–2 paragraphs to answer.

For each prompt, explain the investigatory methods you used and the evidence that supports your conclusion. Place your responses in a plain text file (not DOC, RTF, PDF, or any other format) called `report.txt`. You may include recovered files with your submission, but your report must clearly indicate which of these files are relevant to each response.

As you investigate, be on the lookout for evidence of any other machines or network services that the suspect may have used. These may contain important evidence and raise further questions you'll need to investigate. Per Section 4.1, be sure to contact your supervisor *before* attempting to access any such machines or accounts. Again, start early; management has been known to take up to 24 hours to respond on weekdays and longer on weekends, although we try to respond promptly.

1. Try booting the suspect's machine and using it normally. What *specific* behaviors of this machine make this a bad idea? We strongly recommend that you mount the suspect's drive from a safe system before continuing (see the hints on creating a raw image later in this section).

2. What operating system does the suspect use? Be careful and specific; e.g., say "Windows 2000" instead of just "Windows." (No attachment necessary.)

3. What is the username of the account typically used by the suspect? (No attachment necessary.)

4. Do you have any evidence that the suspect had an accomplice that was physically present on the night of the crime?

5. Were there any suspicious-looking encrypted files on the machine? If so, please attach their contents and a brief description of how you obtained the contents.

6. What evidence do you have that the suspect owned or was researching weapons of the kind involved in the murder? Please attach the specific evidence and a brief explanation.

7. Did the suspect try to delete any files before his arrest? Please attach the name(s) of the file(s) and any indications of their contents you can find. (Hint: We will be impressed enough to

grant extra credit if you manage to recover the *original* contents of a particular incriminating file, but we do not expect you to do so.)

8. Is there anything else suspicious about the machine?

3 Hints and Resources

In addition to the hints we've dropped elsewhere, here is an incomplete list of some things you may want to try:

- Examine the system logs.
- Check for deleted or encrypted files.
- Search the drive image itself (e.g., using `grep -a` or `strings`) for strings that may indicate relevance to your investigation. You'll need to convert the disk image to a "raw" binary image; see the help for the `VBoxManage clonevdi` command.

Some additional resources that *may* help you:

- http://darkdust.net/writings/diskimagesminihowto explains how to find the partitions in a disk image and mount one of them, albeit in a different context than forensics. http://en.wikipedia.org/wiki/Disk_partitioning provides some background that you may be missing.
- John the Ripper (http://www.openwall.com/john/) is the canonical Unix password cracker. Cain and Abel (http://www.oxid.it/cain.html) and Hydra (http://www.thc.org/thc-hydra/) are two other well-known general-purpose password crackers. fcrackzip (http://home.schmorp.de/marc/fcrackzip.html) is a ZIP password cracker, and pdfcrack (http://sourceforge.net/projects/pdfcrack/) is a PDF password cracker. John, fcrackzip, and pdfcrack are conveniently available in the Debian package repositories and may be available for other Linux distributions as well.

  When using a password cracker, it is wise to make sure that the password is not susceptible to a dictionary attack and does not use a restricted character set (e.g., lowercase letters, letters only, letters and numbers only) before spending time on a full brute-force crack. It is also a good idea to crack a very vulnerable password first to make sure you are using the tool correctly.
- "Deleted files recovery howto" (http://e2undel.sourceforge.net/recovery-howto.html) explains how to recover deleted files on the ext2 filesystem using `e2undel` and `debugfs`. http://www.xs4all.nl/~carlo17/howto/undelete_ext3.html explains how to attempt to recover deleted files on the ext3 filesystem.
- A general working knowledge of Linux is probably helpful for this project as well. If you don't have this yet, you may need to spend a little time Googling and/or experimenting to get up to speed. The TA will also answer general Linux questions as a last resort.

4 Policies and Mechanics

4.1 Ethics

In this project you will be investigating the use of computers as part of an entirely fictitious crime. You may access Web pages and other obviously-public computer services in a read-only fashion, but *you must not attempt to log into any computers or accounts over which you do not already have sole control*, even if you recover authentication credentials for those accounts from your analysis. If you think that doing so is necessary to recover additional evidence (hint, hint), you must first email the course TA at `deman@umich.edu`. (Failure to do so is certain to cause wasted time on your part.)

4.2 Collaboration

As stated above, you are bound by the Honor Code not to communicate with anyone outside your group and the courses staff regarding your solutions to this project. The number of pieces of evidence you find, the techniques you try, how successful said techniques are, the general progress you follow, etc. are considered part of your solution and must not be shared between groups.

4.3 If You Get Stuck

Given the nature of this assignment and its strict collaboration policy, HQ recognizes the need for some hints. We have developed standard hints for each question we have asked; if your group gets stuck, you may email `deman@umich.edu` with the names of your group members, the question for which you would like a hint, and the progress you have made thus far on that question. Be sure to put "EECS 398" in the subject line of your email. Each group may receive up to three hints in total, and we will enforce a one-hour delay between hints for each group.

Note: Requesting access to a remote machine does not count as a hint request, nor does asking for help with the first three questions, which are intended to help you get started.

We will respond to hint requests in a best-effort, first-come first-served fashion. In particular, you will not necessarily receive a hint before the project deadline if you request one within 24 hours of the deadline. Start early!

4.4 Turning It In

This project is due **Monday, December 13th** at **5 p.m.** For submission, place your files in a directory named "`proj4`" and archive it as "`proj4.`*`member1uniqname`*`.`*`member2uniqname`*`.tar.gz`". (To help us process submissions easily, please put your uniqnames in alphabetical order). Attach the archive file to an email to `eecs398@umich.edu` with "`Proj4 Submission`" as the subject. You should receive a confirmation email within 15 minutes.