Project 5: Forensics

This project is split into two parts, with the first checkpoint due on **Wednesday, November 18** at **6:00pm** and the second checkpoint due on **Friday, December 4** at **6:00pm**. The first checkpoint is worth 2% of your total grade, and the second checkpoint is worth 10%. We strongly recommend that you get started early. Each semester everyone will be given ONE late extension that allows you to turn in up to one assignment up to 24 hours after the due date. Extensions are not automatic. So, if you want to use your late extension, you MUST send an e-mail to **ece422-staff@illinois.edu**. Late work will not be accepted after 24 hours past the due date.

Start early. It may be impossible to complete this project before the deadline unless you begin several days beforehand. Please plan accordingly.

This is a group project; you SHOULD work in **teams of two** and if you are in teams of two, you MUST submit one project per team. Please find a partner as soon as possible. If have trouble forming a team, post to Piazza's partner search forum.

Strict no-leaks policy. In this project you play the role of a computer forensic analyst working to solve a murder case. Since you don't want to be fired for jeopardizing an ongoing criminal investigation, you need to follow a strict policy on collaboration. *You are bound by the Student Code not to communicate with anyone regarding any aspect of the case or your investigation (other than within your group or with course staff).* The number of pieces of evidence you find, the techniques you try, how successful said techniques are, the general process you follow, etc. are all considered part of your solution and must not be discussed with members of other groups.

Solutions MUST be submitted electronically in any one of the group member's svn directory, following the submission checklist given at the end of each checkpoint. Details on the filename and submission guideline is listed at the end of the document.

"In general, computer forensics is rather ad hoc. Traditional rules of evidence are broken all the time. But this seems like a pretty egregious example."

– Bruce Schneier

Introduction

In this project, you will play the role of a digital forensic analyst and investigate a murder mystery.

A few days after Halloween 2015, a terrible crime occurred on University of Illinois at Urbana Champaign campus. Hapless Victim was murdered in the dorm. Victim was last seen alive on November 4, 2015 early afternoon in class, and was discovered dead at approximately 11pm the same day.

Officers were not able to immediately detect the actual sign of the death and asked forensics team to investigate. While waiting for forensics team to share the result, they have performed digital forensics analysis on victim's hard disk. A suicidal note was left on the hard disk. Forensics team confirmed that the victim was dead as how the suicidal note mentioned. Hence, investigation department's initial conclusion to the case was suicidal.

However, with further investigation, they opened the possibility of murder and filtered out few suspects. They could not find all suspects, but instead, were able to obtain the hard disk of all suspects. Investigators successfully decrypted the hard disks and created image for further investigation.

Your job is to conduct a forensic examination of the disk image and document any evidence related to the murder. If you find sufficient evidence, the suspect will be extradited and face trial.

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.

Guidelines

- You SHOULD work in a group of 2.
- Your answers may or may not be the same as your classmates'.
- All the necessary files to start the project will be given under the folder called "mp5" in your SVN directory. We've also generated some empty files for you to submit your answers in. You MUST submit your answers in the provided files; we will only grade what's there!

Read this First

Collaboration: Strictly prohibited outside your group. As stated above, you are bound by the Student Code not to communicate with anyone regarding any aspect of the case or your investigation (other than within your group or with course staff). The number of pieces of evidence you find, the techniques you try, how successful said techniques are, the general process you follow, etc. are all considered part of your solution and must not be discussed with members of other groups. If anyone brings up the project, put your fingers in your ear, start yelling "LALALALA", run around and refer them to your supervisor for an official spokesperson.

Getting Started

The tools and techniques you use for your investigation are up to you, but here are some suggestions to help you get started. NOTE: This MP requires several tools to be downloaded and installed. For live analysis, you MUST use your own machine to perform the analysis as Virtual Machine is not supported on EWS Linux system. Also, decompressed disk image and virtual hard disk image file you will be generating sums up to >10GB which will take more than your allocated EWS quota. However, for dead analysis, Autopsy tool is already installed on EWS Linux and can be performed on the system. Nevertheless, **Autopsy on Windows OS** provides extra functions which pre-filters interesting evidence and will tremendously save your investigation time. Finally, we strongly recommend you to run password-cracking software on your own machine.

General Knowledge A general working knowledge of Linux is undoubtedly helpful for this project. If you don't have this yet, you may need to spend time Googling and/or experimenting to get up to speed. The TA will also answer general Linux questions as a last resort. For an excellent reference book, try *UNIX and Linux System Administration Handbook* by Nemeth, Snyder, Hein, and Whaley. Ebook can be found at `https://books.google.com/books`. See `http://en.wikipedia.org/wiki/Disk_partitioning` for some additional background.

Live Analysis Live analysis is a forensic technique in which the investigator examines a running copy of the target system. We suggest using VirtualBox for this purpose.

1. Download the compressed raw disk image (1.5 GB):
   `https://subversion.ews.illinois.edu/svn/fa15-cs461/_shared/mp5/forensics_fa15_victim.raw.gz`
2. Decompress the disk image.
   `gunzip forensics_fa15_victim.raw.gz`
3. Verify the checksum.
   `sha256sum -c SHA256SUMS.txt`
   (You can repeat this process for checkpoint 2 VM.)
4. Convert the raw disk image to a VirtualBox disk image:
   `VBoxManage convertdd forensics_fa15_victim.raw forensics.vdi -format VDI`
5. Use the VirtualBox GUI to create a new VM. Select Linux / Ubuntu (32-bit) as the machine type. Select "Use an existing virtual hard disk file" and select the VDI you just created.
6. Start the VM and explore the system.

Dead Analysis In dead analysis, the forensic investigator examines data artifacts from a target system without the system running. We suggest trying dead analysis with the Autopsy open-source forensics tool. The procedure below assumes you are working on Ubuntu Linux. (If you like, you can reuse the VM from the previous project.) Autopsy will also run on Windows and OS X. For OS X, Sleuth Kit should be installed prior to Autopsy installation. Sleuth Kit can be downloaded from `http://www.sleuthkit.org/sleuthkit/download.php`.

1. Install the Autopsy digital forensics suite:
   ```
   $ sudo apt-get install autopsy
   ```

   If you are using EWS Linux machine, load sleuth kit:
   ```
   $ module load sleuthkit
   ```

2. Launch Autopsy in the background and open the browser-based GUI:
   ```
   $ sudo autopsy &
   ```
   In a browser on the local machine, go to the URL `http://localhost:9999/autopsy`.

3. Create a new case and add the disk image:

   (a) Click New Case. Enter a case name and click New Case.

   (b) Go back to `http://localhost:9999/autopsy` and open the case you created.

   (c) Click Add Host. Enter a host name and click Add Host.

   (d) Click Add Image. Click Add Image File. Enter the path to the decompressed raw disk image. Make sure you select Type=Disk and Import Method=Symlink. Click Next.

   (e) Leave the Image File Details and File System Details as the defaults. (Note that the disk image contains 3 partitions, which Autopsy will allow you to examine separately.) Click Add. Click OK.

   (f) Select a partition to examine and click Analyze. The buttons at the top give you several analysis tools. Try File Analysis and Keyword Search to get started.

4. In addition to hints dropped elsewhere, here is an incomplete list of things to try:

   - Examine the system logs.
   - Check for deleted or encrypted files.
   - Search the drive image for strings that may indicate relevance to your investigation.

Password Cracking Password crackers may be helpful in trying to brute-force decrypt password-protected files. John the Ripper (`http://www.openwall.com/john/`) is the canonical Unix password cracker. Hydra (`http://www.thc.org/thc-hydra/`) is a tool used to brute force remote login passwords, 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 can be installed with `apt-get`.

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.

Checkpoint 1 (20 points)

The deliverables for this project are your answers to the questions below. Your answers should be *complete* but *concise*.

For each prompt, you may optionally explain the investigatory methods you used and the evidence that supports your conclusion. The files can be included in a separate directory named `explanation`.

5.1 Exploring Victim's Traces (20 points)

In this part of the project, you will be exploring the traces left on victim's hard disk.

5.1.1 Username (2 points)

What is the username of the victim used on OS?

What to submit: Submit a text file named `5.1.1.txt` that contains the username of the victim.

5.1.2 Conversation (4 points)

Whom did the victim have conversation(s) with? List each username in separate line, in the order of occurrences. If the victim had multiple conversations with the same person in different times, list the username as many as the conversation history. Note that this does not equal to the message count transferred back and forth for single conversation.

What to submit: Submit a text file named `5.1.2.txt` that contains the list of usernames the victim have conversation with.

5.1.3 Evidence file (4 points)

The police initially thought that the victim committed suicide after finding a certain file on the victim's computer.

1. What is the name of the file? Include the file extension
2. When was this file last modified? Submit in MMddhhmm format. (MM=Month, dd=day, hh=hour, mm=minute)

What to submit

1. Submit a text file named `5.1.3_name.txt` that contains the full name of the file.
2. Submit a text file named `5.1.3_time.txt` that contains the last modified time of the file.

5.1.4 Attack (10 points)

After initial conclusion, the police started to be suspicious about possibility of the murder. Do you find any trace of the attack on victim's machine?

1. If so, when did the attacker make the first contact to the victim's computer? Submit the in MMddhhmm format. (MM=Month, dd=day, hh=hour, mm=minute)
2. List all the IP addresses the attack originated from.
3. What was the victim's IP address during the attack?

What to submit

1. Submit a text file named `5.1.4_time.txt` that contains the first attack time.
2. Submit a text file named `5.1.4_attackerip.txt` that contains the list of IP addresses the attack came from.
3. Submit a text file named `5.1.4_victimip.txt` that contains the victim's IP address.

Checkpoint 1: Submission Checklist

Inside your mp5 directory svn, you will have the auto-generated files named as below. Make sure that your answers for all tasks up to this point are submitted in the following files before **Wednesday, November 18** at **6:00pm**:

SVN Directory

`https://subversion.ews.illinois.edu/svn/fa15-cs461/NETID/mp5`

Team Members

`partners.txt` : a text file containing netIDs of both members, one netID per line. Place the student's netID, whose directory contains your project submission, at the top of the file.

example content of partners.txt

```
netid1
netid2
```

Solution Format

example content of 5.1.1.txt

```
victim_username
```

example content of 5.1.2.txt

```
username1
username1
username2
username1
```

example content of 5.1.3_name.txt

```
filename.ext
```

example content of 5.1.3_time.txt and 5.1.4_time.txt

```
10311200
```

example content of 5.1.4_attackerip.txt and 5.1.4_victimip

```
1.2.3.4
```

List of solution files that must be submitted for checkpoint 1

- `partners.txt`
- `5.1.1.txt`
- `5.1.2.txt`
- `5.1.3_name.txt`
- `5.1.3_time.txt`
- `5.1.4_time.txt`
- `5.1.4_attackerip.txt`
- `5.1.4_victimip.txt`

Checkpoint 2 (100 points)

The deliverables for this project are your answers to the questions below. Your answers should be *complete* but *concise*.

Suspect VM that you will explore is given individually in your SVN directory. Follow and download the disk image provided in `suspect_vm.txt`. We have also created `code.txt` in your SVN directory. The instruction for this file will be given at some point in the project, so be patient.

If you recover files that are relevant to your responses, include them with your submission in a directory named `evidence/`. Do not change the original file name on your submission. For each prompt, you may optionally explain the investigatory methods you used and the evidence that supports your conclusion. The files can be included in a separate directory named `explanation`.

5.2 Investigating Suspect Traces (100 points)

Now, the police department has obtained multiple suspects' hard disk, distributed them across teams to be investigated in parallel. You are given with one disk image to analyze. Answer the following set of questions to fill out the report.

5.2.1 Operating Systems (10 points)

Let's first take a look at the machine environment. Be careful and specific; e.g., say "Windows 2000" instead of just "Windows."

1. Try booting the suspect's machine and using it normally. What operating system does it boot by default?
2. (manual grading) What specific behaviors of the default boot OS have on the machine?
3. What operating system did the suspect primarily use? Submit the OS's name and version number in separate line.

What to submit

1. Submit a text file named `5.2.1_default.txt` that contains the default booting OS.
2. Submit a text file named `5.2.1_behavior.txt` that contains the behavior of the default booting OS.
3. Submit a text file named `5.2.1_primary.txt` that contains the primary OS the suspect used.

5.2.2 Username (5 points)

What is the OS username of the suspect?

What to submit: Submit a text file named `5.2.2.txt` that contains the OS username of the suspect.

5.2.3 Conversation (15 points)

You have investigated conversation history of the victim in section 5.1. Now, you will do the same for the suspect.

1. Whom did the suspect have conversation(s) with? List each username in separate line, in the order of occurrences. If the suspect had multiple conversations with the same person in different times, list the username as many as the conversation history. Note that this does not equal to the message count transferred back and forth for single conversation.
2. (manual grading) What is the suspect's relationship with the victim?

What to submit

1. Submit a text file named `5.2.3_usernames.txt` that contains the list of usernames the suspect have conversation with.
2. Submit a text file named `5.2.3_relationship.txt` that contains the relationship of the suspect and the victim.

5.2.4 Search History (15 points)

There must be a reason that the user is considered as suspect.

1. Are there any indications that the suspect was trying to make an attack? How about the any indication that the suspect owned or was researching weapons of the kind involved in the murder? What are the websites suspect visited potentially related with murder? List 5 website full links, one link per line in time order. They all have to come from different domains. E.g. Two different searches on Google count as 1.
2. What did the suspect plan to use as a weapon to murder the victim?
3. How did the suspect plan to obtain the weapon?

What to submit

1. Submit a text file named `5.2.4_link.txt` that contains the list of websites visited related with the murder.
2. Submit a text file named `5.2.4_weapon.txt` that contains the weapon that suspect planned to use.
3. Submit a text file named `5.2.4_method.txt` that contains the method that suspect planned to obtain the weapon.

5.2.5 Encrypted File (10 points)

Were there any suspicious-looking encrypted files on the machine? If so, what was the password that was used to encrypt this file? Also, attach the decrypted contents as evidence. Note: Submit the actual individual files, not the compressed format.

What to submit: Submit a text file named `5.2.5.txt` that contains the password of the encrypted file and decrypted contents in `evidence/` directory.

5.2.6 Attack (20 points)

1. Which account did the suspect use to access the victim's computer? Submit the username of the account.
2. What tools did the suspect use to gain access to the victim's computer? As you investigate, be on the lookout for evidence of any other machines or network services that the suspect may have used. Be careful. The suspect might have decided not to use some tools. List the terminal command name of each tool in separate line in the order of occurrence.
3. List the suspect's IP address(es) used during the attack.
4. Did the suspect successfully connect to the victim's computer? List the filename of the private and public key used for the connection in separate line.
5. What was the password of the account obtained/used by the suspect for the victim's computer?

What to submit

1. Submit a text file named `5.2.6_account.txt` that contains the username of the suspect used to access victim's machine.
2. Submit a text file named `5.2.6_tools.txt` that contains the list of tools victim used during the attack.
3. Submit a text file named `5.2.6_ip.txt` that contains the ip address of the suspect used in the attack.
4. Submit a text file named `5.2.6_connection.txt` that contains whether the connection was successful as well as private and public key filenames.
5. Submit a text file named `5.2.6_password.txt` that contains the password of the username the suspect obtained/used during the attack.

5.2.7 File Extraction (10 points)

Did the suspect try to delete any files that may be related with the murder? List one file name that is the most suspiciously looking and extract/obtain the deleted file. *Hint:* We will be *very impressed* if you can actually recover the deleted file. Double check the file size and content once you extracted.

What to submit: Submit a text file named `5.2.7.txt` that contains the name of the deleted files with file extension and obtained files in `evidence/` directory with the original file name.

5.2.8 Escape Plan (15 points)

If the murder has happened, the suspect may have planned for after-murder scenario.

1. Are there any indications that the suspect had an accomplice who was physically present on the night of the crime, or plan to escape after the crime? If so, submit the contact information of the accomplice or company.

2. What is the location of the escape plan? Submit the GPS coordinate (latitude and longitude) in decimal in separate lines. Use Google Maps to convert the coordinates from degrees to decimal. Double check the location on the map. Submit up to 3rd decimal without rounding. E.g. 1.234567 -> 1.234

3. What was the original time of the escape? Submit in hhmm format (hh=hour, mm=minute)

4. Do you think that the suspect successfully escape? If so, submit the actual escape time in hhmm format (hh=hour, mm=minute). If not, simply say, "unknown" without the quotation.

What to submit

1. Submit a text file named `5.2.8_accomplice.txt` that contains the contact information of the accomplice.

2. Submit a text file named `5.2.8_location.txt` that contains the escape location coordinate.

3. Submit a text file named `5.2.8_originaltime.txt` that contains the original planned escape time.

4. Submit a text file named `5.2.8_actualtime.txt` that contains the actual escape time.

5.2.9 File Metadata (5 points)

Now, go back to the victim's VM. What is the creator of the evidence file you found on 5.1.3?

What to submit: Submit a text file named `5.2.9.txt` that contains the creator of the evidence file.

5.2.10 Final Decision (5 points)

Do you think the suspect actually committed the crime? If so, say yes. Otherwise, no.

What to submit: Submit a text file named `5.2.10.txt` that contains whether you think the suspect is the real criminal or not.

Checkpoint 2: Submission Checklist

Inside your mp5 directory svn, you will have the auto-generated files named as below. Make sure that your answers for all tasks up to this point are submitted in the following files before **Friday, December 4** at **6:00pm**:

SVN Directory

`https://subversion.ews.illinois.edu/svn/fa15-cs461/NETID/mp5`

Team Members

`partners.txt` : a text file containing netIDs of both members, one netid per line. Place the student's netID, whose directory contain your project submission, at the top of the file.

example content of partners.txt

```
netid1
netid2
```

Solution Format

example content of 5.2.1_default.txt and 5.2.1_primary.txt

```
OS X
10.10.4
```

example content of 5.2.3_usernames.txt

```
username1
username1
username2
username1
```

example content of 5.2.4_links.txt

```
https://www.google.com
https://www.yahoo.com
```

example content of 5.2.4_weapon.txt

```
lightsaber
```

example content of 5.2.4_method.txt

```
online shopping
```

example content of 5.2.5.txt

```
p4ssw0rd
```

example content of 5.2.6_tools.txt

```
zip
john
```

example content of 5.2.6_ip.txt

```
1.2.3.4
```

example content of 5.2.6_connection.txt

```
[yes/no]
private_key_file_name
public_key_file_name
```

example content of 5.2.6_password.txt

```
p4ssw0rd
```

example content of 5.2.7.txt

```
filename.ext
```

example content of 5.2.8_location.txt

```
1.234
5.678
```

example content of 5.2.8_originaltime.txt and 5.2.8_actualtime.txt

```
1201
```

example content of 5.2.9.txt

```
Creator Name
```

example content of 5.2.10.txt

```
[yes/no]
```

List of solution files that must be submitted for checkpoint 2

- partners.txt
- 5.2.1_default.txt
- 5.2.1_behavior.txt
- 5.2.1_primary.txt
- 5.2.2.txt
- 5.2.3_usernames.txt
- 5.2.3_relationship.txt
- 5.2.4_link.txt
- 5.2.4_weapon.txt
- 5.2.4_method.txt
- 5.2.5.txt
- evidence/"decrypted_file"
- 5.2.6_account.txt
- 5.2.6_tools.txt
- 5.2.6_ip.txt
- 5.2.6_connection.txt
- 5.2.6_password.txt
- 5.2.7.txt
- evidence/"recovered_file"
- 5.2.8_accomplice.txt
- 5.2.8_location.txt

- 5.2.8_originaltime.txt
- 5.2.8_actualtime.txt
- 5.2.9.txt
- 5.2.10.txt