Put Student Name(s) ↓	Put Student IDs ↓	Due Date	Grade Weight
Ishan Aakash Patel	146151238	As Posted	6%

Name	Lab 7: Malware Static Analysis – Xworm loader uses Steganography
Instructions	• It is an Individual assignment. Put your name + Student ID in the empty spaces above. • Submit your report name: CYT215-Lab7-Student Name & ID
Prior Knowledge	Module 6 – July 9, July 11 from Blackboard
Steps	Watch the following video - https://www.youtube.com/watch?v=11IFAIhCKrA Answer the following questions in a report format: **Part 1: Understanding Steganography and Its Uses** 1. **Define steganography. How is it different from cryptography?** 2. What are some common file types used in steganography? Why are these file types chosen?[Part 2: Detecting Steganography 3. Describe at least two techniques used to detect steganography in image files. 4. What is Exif metadata, and how can it be used in the detection of steganography? Part 3: Extracting Hidden Data 5. Explain the process of extracting hidden data from an image using a steganography tool. 6. What precautions should be taken when extracting data from suspicious files?

Part 4: Analyzing Extracted Malware

7. **What are the primary steps in static analysis of a malware file?**
8. **How does dynamic analysis differ from static analysis in malware examination?**

Part 5: Practical Application

9. **Given an image file suspected of containing hidden data, outline the steps you would take to confirm and analyze the presence of steganography.**
10. Discuss the ethical considerations and potential legal implications of using steganography.

Part 6:

11. Provide a summary of the static analysis done of the malware in the video.

OPTIONAL BUT NOT REQUIRED – FOLLOW THE STEPS IN THE VIDEO AND PERFORM STATIC ANALYSIS IN YOUR OWN SANDBOX ENVIRONMENT. YOU CAN USE THIS FOR FUTURE REFERENCE AND SHOWCASE ON YOUR CV TO FUTURE EMPLOYERS THAT YOU HAVE DONE STATIC ANALYSIS ON MALWARE. PLEASE BE CAREFUL AS THIS IS REAL MALWARE, SO ENSURE IF THAT YOUR VM IS COMPLETELY ISOLATED. AND THAT YOU CREATE SNAPSHOTS.

TOOLS NEEDED IN YOUR SANDBOX

- https://notepad-plus-plus.org/
- https://gchq.github.io/CyberChef/
- https://www.winitor.com/download

	- https://github.com/dnSpyEx/dnSpy Malware sample Here is the sanitized URL: https[://]www[.]virustotal[.]com/gui/file/1a93c7da6bb1bc0b7b4d4e34060ec15e80859886d57cea5847f18f2d7b42b2c0/behavior https[://]bazaar[.]abuse[.]ch/sample/1a93c7da6bb1bc0b7b4d4e34060ec15e80859886d57cea5847f18f2d7b42b2c0/
Grading Alerts	• If you do NOT use this template or delete any part of it or use any other template, you will be degraded. • If you do NOT follow the fie naming convention, you will be degraded. • If you do NOT submit your file in PDF; you will be degraded. • If you do NOT show your account real name (when applicable); you will be degraded. If you do NOT write (in your own words) your learning experience for the activity practices, you will be degraded. NOTE – THIS IS AN INDVIDUAL REPORT AND MUST BE IN YOUR OWN WORDS. ALL THOUGH COLLABORATION IS ENCOURAGED, YOUR REPORT SHOULD BE YOUR OWN. Please follow and abide by the Seneca Academic Integrity policy – *Academic Integrity at Seneca*

What is Academic Integrity?

The International Center for Academic Integrity defines academic integrity as a commitment, even in the face of adversity, to six Fundamental Values of Academic Integrity: honesty, trust, fairness, respect, responsibility, and courage. From these values flow principles of behavior that enable academic communities to translate ideals into action.

Why does Academic Integrity Matter?

When each member of the Seneca Community embraces and incorporates these values into our teaching, learning and working environments, then we are able to maintain the college's reputation as a leading educational institution and to graduate high quality students who are poised to succeed in their careers and contribute meaningfully to society.

Academic Integrity - Student Resources

Part 1: Understanding Steganography and Its Uses

1. **Definition of Steganography and Its Difference from Cryptography**:
    - **Steganography**: Steganography is the art and science of hiding information within another medium to avoid detection. The goal is to conceal the existence of the hidden information.
    - **Difference from Cryptography**: Cryptography involves encoding information so that only authorized parties can read it, while steganography hides the existence of the information itself. Cryptography ensures confidentiality, whereas steganography ensures secrecy.

2. **Common File Types Used in Steganography**:
    - **Image Files (e.g., JPEG, PNG)**: These are commonly used because they are ubiquitous, and the human eye is less likely to notice minor alterations in image data.
    - **Audio Files (e.g., MP3, WAV)**: These can also carry hidden data within their structure without noticeable changes in sound quality.
    - **Video Files (e.g., MP4, AVI)**: Videos provide ample space to hide data due to their large size and complex structure.
    - **Text Files and Document Files (e.g., TXT, DOCX)**: Less common but can still be used for hiding information.
    - **Reasons for Choice**: These file types are chosen because they are commonly exchanged, making hidden data less suspicious, and they have redundant or less perceptible data regions suitable for embedding information.

Part 2: Detecting Steganography

3. **Techniques to Detect Steganography in Image Files**:
    - **Statistical Analysis**: Analyzing the statistical properties of an image to detect anomalies or patterns that suggest hidden data. For instance, examining the least significant bits (LSBs) of pixel values.
    - **Steganalysis Tools**: Using specialized software designed to detect steganography, such as Stegdetect, which can identify and analyze suspected files for hidden data.
4. **Exif Metadata and Its Use in Detection**:
    - **Exif Metadata**: Exchangeable image file format (Exif) metadata contains information about the image, such as the camera settings, date, time, and location of capture.
    - **Use in Detection**: By examining Exif metadata, investigators can identify inconsistencies or unusual patterns that may indicate steganography. Additionally, metadata can sometimes contain hidden data or clues about its presence.

Part 3: Extracting Hidden Data

5. **Process of Extracting Hidden Data from an Image**:
    - **Using Steganography Tools**: Tools like StegHide, OpenStego, or StegExpose are used to extract hidden data. The process involves:
        - Loading the suspected image file into the tool.
        - Providing any required password or key if the data is encrypted.

        - Running the extraction function to retrieve the hidden data.
    - **Interpreting Results**: After extraction, the hidden data needs to be analyzed, which might involve decoding if the data is encrypted.
6. **Precautions When Extracting Data from Suspicious Files**:
    - **Use Secure Environments**: Always perform extraction in a secure, isolated environment to prevent potential malware from spreading.
    - **Scan Files for Malware**: Before and after extraction, scan the files using updated antivirus and anti-malware tools.
    - **Maintain Backups**: Keep backups of the original files to prevent data loss or corruption during the extraction process.

Part 4: Analyzing Extracted Malware

7. **Primary Steps in Static Analysis of a Malware File**:
    - **File Examination**: Analyze the file structure, size, and type.
    - **Disassembly**: Use tools like IDA Pro to disassemble the code and inspect its instructions.
    - **Signature Analysis**: Compare the malware's code against known malware signatures.
    - **Metadata Analysis**: Inspect any embedded metadata for clues about its origin or function.
8. **Difference Between Dynamic and Static Analysis in Malware Examination**:
    - **Static Analysis**: Involves examining the code without executing it. It includes analyzing file structure, disassembly, and looking for known signatures or patterns.
    - **Dynamic Analysis**: Involves executing the malware in a controlled environment to observe its behavior, interactions with the system, network activity, and any changes it makes to the system.

Part 5: Practical Application

9. **Steps to Confirm and Analyze the Presence of Steganography in a Suspected Image File**:
    - **Initial Inspection**: Perform a visual and metadata inspection of the image.
    - **Use Steganalysis Tools**: Employ tools like Stegdetect or StegExpose to scan the image for hidden data.
    - **Extract Hidden Data**: If steganography is detected, use appropriate tools to extract the hidden data.
    - **Analyze Extracted Data**: Examine the extracted data for content and potential threats.
10. **Ethical Considerations and Legal Implications of Using Steganography**:
    - **Ethical Considerations**: While steganography can be used for legitimate purposes like protecting privacy, it can also be misused for illegal activities such as hiding criminal communications or data theft. Ethical use involves ensuring it does not harm others or violate laws.
    - **Legal Implications**: The use of steganography can have legal consequences, especially if used for illicit purposes. It may lead to charges related to data concealment, fraud, or other criminal activities. Legal guidelines vary by jurisdiction, so understanding local laws is crucial.

Part 6: Summary of Static Analysis Done in the Video

11. Summary of Static Analysis of Malware in the Video

In the video, the process of static analysis of the malware file is thoroughly demonstrated. Here's a detailed breakdown:

- **File Identification**:
  - The analysis begins with identifying the malware file type. The file's extension and size are noted, and basic properties are examined to ascertain its format, such as .exe, .dll, or .doc. This helps in determining the tools and techniques suitable for further analysis.
- **Disassembly and Code Inspection**:
  - The video showcases the use of a disassembler, such as IDA Pro or Ghidra, to convert the binary code into assembly language. This step is crucial as it allows analysts to inspect the instructions and understand the program flow. By examining the assembly code, the analyst can identify function calls, loops, and data manipulations, gaining insight into what the malware does.
- **Signature and Pattern Matching**:
  - A key part of static analysis is comparing the malware's code against a database of known malware signatures. Tools like VirusTotal or YARA rules are used for this purpose. The video demonstrates how these tools can flag known malicious patterns or signatures, aiding in the rapid identification of the malware's family or type.
- **Metadata Examination**:
  - The malware's metadata is scrutinized for additional clues. This includes looking at the file's creation date, author information, version number, and any embedded resources such as icons, strings, or other files. Metadata can reveal the malware's origin, its intended target, or additional components it might carry.
- **String Analysis**:
  - Strings embedded within the executable are extracted and analyzed. These strings often include URLs, IP addresses, file paths, or API calls that the malware uses. This step helps in understanding the malware's communication protocols, command and control servers, and its overall functionality.
- **Heuristic Analysis**:
  - The video also covers heuristic analysis, where the code is examined for suspicious patterns or behaviors that are not necessarily part of known malware signatures. This might include unusual API calls, unexpected system modifications, or attempts to evade detection. Heuristics help in identifying new or previously unknown malware.
- **Comparative Analysis**:
  - The analyst compares the findings with known malware samples in databases or research papers. This comparison can confirm the malware's family and behavior, aiding in understanding its impact and the countermeasures needed.
- **Documentation and Reporting**:
  - Finally, the results of the static analysis are documented in a detailed report. This report includes the malware's characteristics, code snippets, detected signatures, and potential indicators of compromise (IoCs). The documentation is crucial for developing defense strategies and informing incident response teams.

Learning Experience

In conducting the static analysis of malware as demonstrated in the video, I gained a deep understanding of how crucial it is to examine a file without executing it. This process involved identifying the file type, disassembling the code, and looking for known signatures and patterns. The step-by-step inspection of the malware’s assembly code and embedded strings highlighted how analysts can uncover the malicious behavior and intent without risking the execution of potentially harmful software.

Furthermore, the experience underscored the importance of using various tools and techniques to gather comprehensive information about the malware. From examining metadata to comparing findings with known samples, each step provided valuable insights into the malware's characteristics and potential impact. This methodical approach to static analysis not only enhances technical skills but also emphasizes the importance of thorough documentation and reporting in cybersecurity practices.