Computer Forensics

Week 8: **DIGITAL FORENSICS ANALYSIS AND VALIDATION**

Kumi University

Lecturer: Lemi Agrey Oliver

codingissweet@gmail.com

Recap

- Last week, we delved into the intricate world of graphic file recovery, exploring various aspects crucial for understanding and executing this process effectively. Here are the key points covered:
- **Recognizing a Graphics File**: We began by discussing the fundamental concept of identifying graphic files amidst the plethora of digital data. Recognizing specific file signatures and extensions plays a vital role in this initial stage.
- **Understanding Bitmap and Raster Images**: Moving forward, we examined the difference between bitmap and raster images, emphasizing their significance in the realm of digital graphics and their implications for recovery procedures.

Recap+

- **Understanding Digital Photograph File Formats**: A deep dive into digital photograph file formats shed light on the diverse array of extensions like JPEG, PNG, TIFF, and RAW, each with its own characteristics and implications for recovery.
- **Steps in Searching for and Recovering Digital Photographs**: We outlined a systematic approach to searching for and recovering digital photographs, highlighting the importance of utilizing specialized recovery software and techniques tailored to the specific file format and storage medium.
- **Understanding Data Compression**: The lecture also touched upon data compression techniques commonly employed in graphic files, such as lossy and lossless compression, and their impact on the recovery process.

Recap++

- **Understanding Steganography in Graphics Files,** An exploration of steganography within graphic files revealed the covert techniques used to conceal data, underscoring the need for advanced tools and methodologies to detect and extract hidden information.
- **Understanding Copyright Issues with Graphics,** Lastly, we addressed the ethical and legal considerations surrounding graphic file recovery, particularly concerning copyright issues and the importance of respecting intellectual property rights.

Content

- Determining What Data to Collect and Analyze
- Approaching Digital Forensics Cases
- Basic steps for all digital forensics investigations
- **Refining and Modifying the Investigation Plan**
- Using Autopsy to Validate Data
- Validating with Hexadecimal Editors
- Addressing Data-Hiding Techniques

Determining What Data to Collect and Analyze

- Deciding on the data to collect and analyze is fundamental across various domains for making well-informed decisions.
- Begin by clearly outlining the goals and inquiries you seek to address through data analysis.
- Recognize the nature of data required, whether qualitative, quantitative, or a blend of both.
- Explore the origins of data: whether it is gathered firsthand (primary) or sourced from existing records (secondary).
- Ensure the data chosen is pertinent, dependable, and accurate to enhance the analysis' credibility.

Determining What Data to Collect and Analyze+

- Select suitable methods for data collection such as surveys, interviews, observations, or experimental approaches.
- Employ technology and software tools for effective data management and analysis.
- Uphold ethical principles throughout, including obtaining informed consent and safeguarding data privacy.
- Continuously assess and refine the data collection and analysis process to adapt to evolving needs and contexts.

Approaching Digital Forensics Cases

- Initiating a digital forensics case involves crafting an investigation plan detailing its objectives, scope, required materials, and tasks.
- While general principles are applicable across digital forensics cases, the specific approach is heavily influenced by the case type.
- For instance, gathering evidence for an e-mail harassment case might entail accessing network logs and e-mail server backups to pinpoint particular messages.
- The approach varies based on whether the case is an internal organizational inquiry or a civil or criminal investigation conducted by law enforcement.

Approaching Digital Forensics Cases+

- Internal investigations often feature relatively straightforward evidence collection due to private-sector investigators' access to requisite records and files.
- In contrast, criminal cyberstalking inquiries may involve more complexity, requiring specialized methods and potential collaboration with law enforcement agencies.
- Surveillance measures like cameras and keyloggers may be necessary in such investigations.
- Network administrators may need to monitor internet and network activities.
- Remote acquisition of an employee's drive and peripheral device monitoring may be required.

Basic steps for all digital forensics investigations

- Ensure that target drives for digital forensics investigations are recent, wiped, reformatted, and checked for malware.
- Employ standard network security practices when accessing network storage media.
- Use disk-to-disk forensic copying to reformat the target drive to the original configuration.
- Utilize tools like X-Ways Security, Digital Intelligence PDWipe, or WhiteCanyon SecureClean to wipe data from media if necessary.
- Inventory and document hardware components of the suspect's computer during seizure.
- Conduct static acquisitions by removing the original drive, if feasible, and verifying CMOS date and time values.

Basic steps for all digital forensics investigations+

- Document the data acquisition process, including the creation of a bit-stream image and MD5 or SHA-1 hash validation.
- Systematically examine the drive's contents, listing all files and folders, noting evidence locations, and their relevance to the investigation.
- Review all data files, prioritizing password-protected files for recovery using tools like OSForensics Password Recovery and Decryption.
- Identify and investigate executable files not matching known hash values, noting any suspicious system files or folders.
- Maintain strict control over evidence, documenting all findings throughout the examination process.

Refining and Modifying the Investigation Plan

- Scope in civil and criminal cases is typically outlined by search warrants or subpoenas, specifying recoverable data.
- Private-sector cases, like employee abuse investigations, may lack defined data recovery limitations, requiring a refined investigation plan.
- The investigation plan should balance breadth to encompass relevant evidence with avoiding wasteful analysis of irrelevant data.
- Flexibility is crucial; deviations from the initial plan may be necessary based on emerging evidence.
- Narrowing the search based on specific criteria, such as timeframes or types of data, optimizes efficiency.

Refining and Modifying the Investigation Plan+

- Initial identification of data types streamlines the search process, preventing excessive data collection.
- Flexibility allows for adjustments in the investigation scope based on newly discovered evidence.

Using Autopsy to Validate Data

- Autopsy for Windows is utilized for forensic analysis across various file systems including Microsoft FAT, NTFS, ExFAT, UFS1, UFS2, ISO 9660, YAFFS2, Mac HFS+, HFSX, and Linux Ext2fs, Ext3fs, Ext4fs.
- It can analyze data from diverse sources, including image files from different vendors, encompassing raw, Expert Witness, and virtual machine image files (.vdi and .vhd).
- Autopsy integrates an indexed version of the NIST National Software Reference Library (NSRL) of MD5 hashes to aid in searching and eliminating known OS and application files.
- NSRL reference hashes can be imported into Autopsy, providing further assistance in forensic analysis.

Collecting Hash Values in Autopsy

- **Access Data Tab:** Open the Autopsy software and navigate to the Data tab.
- **Select Data Source:** Choose the data source from which you want to collect hash values. This could be a disk image, a folder, or any other storage medium.
- **Configure Hash Settings:** In the data source settings or preferences, locate the option for hash calculation or hashing settings.
- **Choose Hash Algorithm:** Select the hash algorithm you want to use for generating hash values. Common options include MD5, SHA-1, and SHA-256.
- **Initiate Hashing Process:** Start the hashing process for the selected data source. Autopsy will compute the hash values for all files within the data source according to the chosen algorithm.

Collecting Hash Values in Autopsy+

- **Monitor Progress:** Monitor the progress of the hashing process within Autopsy. Depending on the size of the data source and the speed of your system, this process may take some time.
- **Review Results:** Once the hashing process is complete, review the generated hash values. Autopsy typically provides a list or report displaying the computed hashes alongside the corresponding file paths.
- **Export Hash Values:** If needed, export the hash values for further analysis or documentation. Autopsy often allows exporting hash lists in various formats such as CSV or text files.
- **Verify Integrity:** Use the generated hash values to verify the integrity of files or to compare against known hash values, such as those from the NSRL (National Software Reference Library).
- **Repeat as Necessary:** Repeat the process for additional data sources or whenever updated hash values are required for your forensic analysis.

Validating Forensic Data

- **Open Case:** Launch Autopsy and open the case containing the forensic data you want to validate.
- **Access Data Tab:** Navigate to the Data tab within Autopsy.
- **Review Data:** Review the forensic data collected during the investigation. This may include disk images, file systems, and individual files.
- **Verify Hash Values:** Check the hash values generated for the data to ensure integrity. Autopsy typically computes hash values using algorithms like MD5, SHA-1, or SHA-256.
- **Compare Hashes:** Compare the generated hash values with known or expected hash values. This comparison helps confirm that the data has not been altered or tampered with.

Validating Forensic Data+

- **Examine Metadata:** Examine metadata associated with the files and file system to ensure consistency and accuracy. This includes attributes such as file creation/modification dates, file sizes, and file permissions.
- **Analyze File Content:** Analyze the content of files to confirm their relevance to the investigation and to verify that they have not been modified inappropriately.
- **Cross-check evidence:** Cross-check the forensic data with other sources of evidence or information to validate findings and conclusions.
- **Document Findings:** Document the validation process, including any discrepancies or anomalies found, as part of the forensic report.

Validating Forensic Data++

- **Maintain Integrity:** Throughout the validation process, ensure the integrity of the forensic data by following proper procedures and maintaining a chain of custody.
- **Repeat as Needed:** Repeat the validation process as needed, especially when new evidence is collected or when discrepancies are discovered, to ensure the accuracy and reliability of the forensic data.

Validating with Hexadecimal Editors

- Validating forensic data using hexadecimal editors in Autopsy involves the following steps
- **Open Forensic Image,** Launch Autopsy, and open the forensic image or data set you want to validate using a hexadecimal editor.
- **Navigate to Hex View,** Within Autopsy, locate the option to view the data in hexadecimal format or open a hexadecimal editor tool.
- **Inspect Data and review** the hexadecimal representation of the data to identify patterns, anomalies, or signs of tampering. Pay attention to changes in file signatures, unexpected data alterations, or suspicious patterns

Validating with Hexadecimal Editors+

- **Compare to Original,** If available, compare the hexadecimal representation of the forensic data to the original source or known reference data. Look for any discrepancies that may indicate data corruption or manipulation.
- **Examine File Headers,** Focus on examining file headers and structures to ensure they match the expected format for the file type. Any inconsistencies could suggest tampering or data corruption.
- **Check for Metadata,** Inspect metadata embedded within the hexadecimal representation, such as timestamps, file attributes, and allocation information, to verify accuracy and consistency.
- **Analyze File Content,** Dive deeper into specific file contents within the hexadecimal editor to verify data integrity and identify any unauthorized modifications or deletions.

Validating with Hexadecimal Editors++

- **Use Search Functions,** Utilize search functions within the hexadecimal editor to look for specific patterns, keywords, or known signatures associated with evidence relevant to the investigation.
- **Document Findings,** Document any findings, discrepancies, or suspicious observations uncovered during the validation process. Include detailed descriptions and screenshots as necessary in your forensic report.
- **Maintain Chain of Custody,** Throughout the validation process, adhere to proper chain of custody procedures to ensure the integrity and admissibility of the forensic data as evidence.
- **Repeat as Necessary,** Repeat the validation process as needed, especially when new evidence is collected or when discrepancies are discovered, to ensure the accuracy and reliability of the forensic data.

Using Hash Values to Discriminate Data

- Many current digital forensics tools, including AccessData's FTK, offer this function.
- AccessData's Known File Filter (KFF) within FTK filters known program files and illegal files, such as child pornography, based on their hash values.
- KFF compares known file hash values with those on the evidence drive or image files to identify suspicious data.
- AccessData periodically updates the KFF with new hash values for enhanced accuracy.
- The NIST National Software Reference Library (NSRL) maintains a database of updated file hash values for various OSs and applications but does not include hash values of known illegal files.
- Other digital forensics tools like X-Ways Forensics, OSForensics, and Forensic Explorer can import the NSRL database for hash comparisons.

Validating with Digital Forensics Tools

- Validating forensic data with digital forensics tools involves the following steps:
- **Choose Suitable Tool:** Select a digital forensics tool capable of validating forensic data. Common tools include Autopsy, FTK (Forensic Toolkit), EnCase, X-Ways Forensics, and OSForensics.
- **Open Data Source:** Launch the chosen digital forensics tool and open the forensic data source or image you want to validate.
- **Review Data:** Review the forensic data within the tool's interface, examining file metadata, content, and structures.
- **Verify Hash Values:** Utilize built-in features to verify hash values of files within the forensic image or data source. Compare these hash values with known reference values to ensure data integrity.

Validating with Digital Forensics Tools+

- **Examine Metadata:** Analyze metadata associated with files, such as timestamps, file attributes, and allocation information, to confirm accuracy and consistency.
- **Check File Signatures:** Use the tool's capabilities to check file signatures and headers to ensure they match the expected format for each file type.
- **Perform Keyword Searches:** Conduct keyword searches within the digital forensics tool to locate specific evidence or indicators relevant to the investigation.
- **Use File Carving:** Employ file carving techniques offered by the tool to recover deleted or fragmented files and verify their integrity.

Validating with Digital Forensics Tools++

- **Analyze File Content:** Dive deeper into file content using the tool's analysis features to identify any unauthorized modifications or discrepancies.
- **Document Findings:** Document all findings, observations, and discrepancies uncovered during the validation process using the tool's reporting functionalities.
- **Maintain Chain of Custody:** Throughout the validation process, ensure adherence to proper chain of custody procedures to preserve the integrity and admissibility of the forensic data as evidence.
- **Repeat as Necessary:** Repeat the validation process as needed, especially when new evidence is collected or when discrepancies are discovered, to ensure the accuracy and reliability of the forensic data.

Addressing Data-Hiding Techniques

- Data hiding is the practice of concealing information within other data, structures, or media to prevent unauthorized users from detecting or accessing the information. Users can apply data hiding across various domains and contexts, including information hiding in programming, invisible ink, digital art and design, and cybersecurity[1]
- Data hiding techniques include hiding entire partitions, changing file extensions, setting file attributes to hidden, bit-shifting, using encryption, and setting up password protection

Data hiding benefits

- Ensures confidentiality of sensitive data in cyberspace.
- Prevents unauthorized access to private information.
- Mitigates risk of data breaches and cyberattacks.
- Facilitates secure communication channels.

Addressing Data-Hiding Techniques+

Data hiding benefits+

- Protects digital assets and intellectual property.
- Bolsters authentication and verification processes.
- Enhances the organization’s overall cybersecurity posture.

Addressing Data-Hiding Techniques++

- **Encryption,** It utilizes cryptographic algorithms to convert data into ciphertext, making it unreadable to parties without the correct decryption key.
- **Steganography,** refers to hiding information within other data or media, such as images, audio files, or videos. Since hackers cannot assume the specific embedding method, they cannot detect the hidden data.
- **Obfuscation,** this method alters the data structure, format, or logic to make data more difficult to understand, therefore protecting sensitive information from unauthorized access or tampering.
- **Data masking,** refers to replacing sensitive information with fictional or scrambled data that maintains the same structure and format.

Other Data-Hiding Techniques

Hiding Files by Using the OS

- One early method of concealing data involved changing file extensions.
- For instance, a suspect might change the extension of an Excel file (.xlsx) to that of a JPEG (.jpg) to hide incriminating evidence.
- When an investigator attempts to open the file in Excel, an error message will indicate it can't be opened.
- Advanced digital forensics tools, however, analyze file headers and compare extensions to detect discrepancies.

Other Data-Hiding Techniques++

- If a mismatch is found, the tool flags the file for further investigation as potentially altered.
- Another method involves setting the Hidden attribute in a file's Properties dialog box.
- Investigators can reveal hidden files in File Explorer by selecting the appropriate option.
- Nevertheless, digital forensics tools have the capability to uncover hidden files.

Hiding Partitions

- Hiding Partitions with diskpart:
  - ✓ Utilize the Windows disk partition utility, diskpart.
  - ✓ Execute the "remove letter" command in PowerShell to unassign the partition's letter, thus concealing it from File Explorer.
  - ✓ To reveal the partition, employ the "assign letter" command in disk part.

Other Data-Hiding Techniques+

- **Gap Detection and Analysis,** Gaps between partitions, typically 128 bytes in Windows Vista and later, can be accessed by most digital forensics tools or hexadecimal editors.
- **Tools for Detecting Hidden Partitions:**Digital forensics tools like WinHex and Autopsy can detect and display both regular and hidden partitions on a disk volume.
- WinHex additionally enables viewing of all connected disks.

Other Data-Hiding Techniques+

Marking Bad Clusters

- Concealing data in free or slack space within disk partition clusters is a data-hiding technique utilized in FAT file systems, albeit no longer prevalent.
- This method involves using older tools like Norton DiskEdit from the Norton Utilities suite by Symantec.
- In Norton DiskEdit, one can designate good clusters as bad clusters in the File Allocation Table (FAT), rendering them inaccessible by the operating system unless reverted to good clusters using a disk editor.
- To flag a good cluster as bad in Norton DiskEdit, you input the letter 'B' in the corresponding FAT entry during FAT table inspection.
- Once marked as bad, these clusters can be read from and written to using any DOS disk editor, remaining hidden as they appear as unusable to the OS.

Other Data-Hiding Techniques+++

- The last version of DiskEdit available was included in Norton Utilities version 8.0, designed to operate exclusively in MS-DOS and compatible with only FAT-formatted disk media.
- Unlike running it via Windows command prompts, DiskEdit can't be run through those means.
- Symantec Norton Utilities, including DiskEdit, can be obtained from Win World PC in a 7z compressed file format. Instruction details are available in the accompanying READ.ME file.
- Additional Norton DOS utilities can be explored on vetusware.com under Norton Utilities 8.0.

Other Data-Hiding Techniques++++

Bit-Shifting

- Home computer users with programming skills in manufacturers' assembly languages have developed low-level encryption programs.
- These programs change the order of binary data, rendering the modified data unreadable when opened with software like text editors or word processors.
- Users apply these programs to rearrange bits for each byte in a file to securely encrypt sensitive or incriminating information.
- An assembler program, referred to as a "macro," is executed on the file to mix up the bits, ensuring data security.
- Another program is then utilized to restore the original order of the scrambled bits for file access.
- Such encryption techniques, still in use today, can complicate data analysis for investigators examining a suspect drive.

Other Data-Hiding Techniques+++++

- Initial steps involve identifying unfamiliar files that could potentially reveal new evidence.
- Training in assembly language and higher-level programming languages like Visual Basic, Visual C++, or Perl can be advantageous in understanding and employing these encryption methods.
- Another known data-hiding technique involves shifting bit patterns to modify byte values within the data.
- Bit-shifting transforms readable code into data resembling binary executable code, adding a layer of obfuscation.
- Tools like WinHex and Hex Workshop offer functionalities to shift bits and adjust byte patterns for entire files or specified data segments.

Other Data-Hiding Techniques+++++

Steps to follow in shifting bits[2]

- 1. Start Notepad, and in a text document, type **TEST FILE. Test file is to see how shifting bits will alter the data in a file**.
- 2. Save the file as **Bit_shift.txt** in your work folder, and exit Notepad.
- 3. Start WinHex, using the **Run as administrator** option. (If necessary, when the UAC message box opens, click **Yes**.) Click **File**, **Open** from the menu. Navigate to your work folder, and then double-click **Bit_shift.txt**.
- 4. To set up WinHex for bit-shifting, click **Options**, **Edit Mode** from the menu. Click **Default Edit Mode (=editable)**, if necessary, and then click **OK**

Other Data-Hiding Techniques+++++

❖ 5. Highlight all the data in the file by clicking **Edit**, **Select All** from the menu.

❖ 6. Click **Edit**, **Modify Data** from the menu. In the Modify Block Data dialog box, click the **Left shift by 1** bit option button, and then click **OK**.

❖ 7. Click **File**, **Save As** from the menu, and save the file as **Bit_shift_left. txt** in your work folder. Exit and then restart WinHex.

❖ 8. To return the file to its original configuration, you need to bit-shift it back to the right. Make sure the data is highlighted, and then click **Edit**, **Modify Data** from the menu. In the Modify Block Data dialog box, click **Right shift by 1 bit**, and then click **OK**.

❖ 9. Save the file as **Bit_shift_right.txt** in your work folder, and leave this file open in WinHex for the next activity

Understanding Steganalysis Methods

- Steganalysis is the study and practice of detecting hidden information, often embedded in digital media like images, audio files, or other types of data.
- It is a counterpart to steganography, which is the art of hiding information.
- Understanding steganalysis involves familiarizing oneself with various techniques and methods used to uncover hidden data.

Key methods used in steganalysis

- **Statistical Methods,** Statistical steganalysis involves analyzing the statistical properties of the media to detect anomalies caused by the hidden information.

Understanding Steganalysis Methods+

- These methods often rely on the assumption that embedding information alters the statistical characteristics of the cover medium.
  - ✓ **Histogram Analysis:** Examines the frequency distribution of pixel values or other elements to detect irregularities.
  - ✓ **Chi-Square Test:** Used to detect deviations in expected vs. observed frequencies of pixel values.
  - ✓ **Sample Pair Analysis:** A technique that analyzes pairs of samples to detect changes in their distribution.

Understanding Steganalysis Methods++

- **Transform Domain Methods, t**ransform domain methods involve transforming the media (e.g., using Fourier or Wavelet transforms) and analyzing the coefficients for anomalies.
  - ✓ **Discrete Cosine Transform (DCT),** Commonly used in JPEG images, this method analyzes the DCT coefficients for signs of hidden data.
  - ✓ **Wavelet Transform,** Useful for detecting hidden data in images and audio files by examining wavelet coefficients.
- **Machine Learning-Based Methods,** Machine learning approaches use algorithms to classify data as either containing hidden information or not. These methods involve training a model on a labeled dataset.

Understanding Steganalysis Methods+++

- ✓ **Support Vector Machines (SVM):** A classification method that finds the optimal hyperplane to separate steganographic and non-steganographic data.
- ✓ **Neural Networks:** Deep learning models can learn complex patterns in data to detect steganographic content.
- ✓ **Random Forests:** An ensemble learning method that uses multiple decision trees to improve detection accuracy.

➢ **Signature-Based Methods,** These methods look for known patterns or signatures left by specific steganographic tools or techniques.

- ✓ **Tool-Specific Signatures,** any steganographic tools leave unique patterns that can be detected.
- ✓ **Payload Signatures, c**ertain types of hidden data may introduce specific patterns that can be recognized.

Understanding Steganalysis Methods++++

- **Structural Analysis,** Structural methods involve analyzing the structure and metadata of the media file for inconsistencies.
  - ✓ **File Structure Analysis,** Examines headers, footers, and other structural components of files for signs of manipulation.
  - ✓ **Metadata Analysis,** Checks for unusual metadata entries or patterns.

Challenges in Steganalysis

- **Evolving Steganographic Techniques,** As steganography methods evolve, steganalysis must continuously adapt.
- **False Positives/Negatives,** Balancing the rate of false positives and false negatives is crucial for effective detection.
- **Complexity and Computation,** Some methods require significant computational resources, which can be a limitation.

Examining Encrypted Files

- Advanced encryption programs like PGP or BestCrypt make data unreadable to unauthorized users.
- Encrypted files require a password or passphrase for access, making recovery difficult without it.
- Key escrow technology in commercial encryption programs helps recover data if passphrases are forgotten or keys are corrupted.
- Forensic examiners can use key escrow to try to retrieve encrypted data.
- Cracking encryption schemes usually requires resources beyond those available to small or medium organizations.
- When encountering encrypted data in an investigation, try to get the suspect to disclose the passphrase.

Examining Encrypted Files+

- Some encryption schemes are so complex that breaking them can take from days to decades.
- Key sizes ranging from 128 bits to 4096 bits make brute-force attacks nearly impossible with current technology.
- Quantum computing may eventually render many current encryption methods obsolete.
- Currently, some encryption schemes remain unbreakable with commercially available tools.

Summary

This lecture covers key aspects of digital forensics, focusing on the following topics

Determining Data to Collect and Analyze, Emphasizes the importance of selecting relevant data sources to ensure the integrity and relevance of evidence in an investigation.

Approaching Digital Forensics Cases, Outlines the foundational steps of any investigation: acquisition, preservation, examination, analysis, and reporting.

Highlights the need for a systematic approach to handle digital evidence effectively.

Refining and Modifying the Investigation Plan, Stresses the importance of adapting strategies based on new findings and evolving circumstances to stay on track.

Summary+

Using Autopsy to Validate Data, Introduces Autopsy, an open-source digital forensics tool, for authenticating and validating the integrity of acquired data.

Validating with Hexadecimal Editors, Explains the use of hexadecimal editors for detailed examination and validation of digital artifacts, file structures, and metadata.

Addressing Data-Hiding Techniques, Discusses methods like steganography, encryption, and file obfuscation, teaching how to recognize and counteract these techniques to uncover hidden evidence.

Reference

[1] Data hiding - glossary. (2023). Retrieved from https://nordvpn.com/cybersecurity/glossary/data-hiding/

[3] Nelson, B., Phillips, A., & Steuart, C. (2019). *Guide to computer forensics and investigations* Course Technology Cengage Learning.