

# Malware Analysis Report

## Understanding Malware and Its Analysis

The term *malware* originates from the words malicious software, referring to any software designed with harmful intent. While malware can be categorized based on its behavior, this discussion will not delve into those classifications. Instead, we will focus on the essential steps to take when encountering potential malware on a system.

## Why Malware Analysis Matters

Malware analysis is a critical skill in cybersecurity, playing a vital role across various security domains. Professionals who perform malware analysis include:

- **Security Operations (SecOps) Teams** – They examine malware to develop detection mechanisms for identifying malicious activities within their networks.
- **Incident Response Teams** – Their goal is to assess the damage caused by malware and implement strategies to mitigate and reverse its impact.
- **Threat Hunting Teams** – These experts analyze malware to extract Indicators of Compromise (IOCs), which they use to proactively search for threats within a network.
- **Malware Researchers at Security Vendors** – They study malware to enhance security products by improving threat detection and prevention capabilities.
- **Threat Research Teams at OS Vendors (e.g., Microsoft, Google)** – These professionals investigate malware to uncover exploited vulnerabilities and strengthen the security of operating systems and applications.

By understanding malware and its analysis, security professionals can better defend against cyber threats and enhance overall system protection.

## Static Analysis

Static analysis refers to examining malware without executing it. This approach involves analyzing various properties of a **Portable Executable (PE) file** or a **malicious document** without running them. For instance, reviewing a document's metadata or structure without opening it falls under static analysis.

Common static analysis techniques include:

- Extracting **strings** from the malware to identify potential commands, URLs, or embedded messages.
- Inspecting the **PE header** to gather information about different sections of the executable.
- Using a **disassembler** to examine the code without executing it.

## Dynamic Analysis

Malware must execute to carry out its malicious intent. Regardless of how well it is obfuscated, once it runs, it becomes easier to detect.

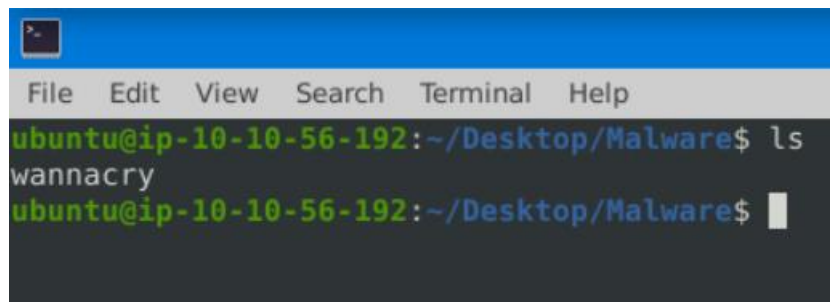
While static analysis can reveal important details about malware, it may not always be sufficient. In cases where malware conceals its properties to evade detection, dynamic analysis becomes essential. This approach involves running the malware in a controlled environment to observe its behavior, such as:

- File system modifications
- Network connections
- Process creation and registry changes

## Identifying the File Type

While a file's extension often indicates its type, malware authors frequently use deceptive extensions to mislead users. For example, a malicious executable might appear as a harmless document or image.

To determine a file's actual type without relying on its extension, various methods can be used. On Linux, the `file` command is a simple yet effective way to inspect a file's true nature. This command analyzes the file's magic number and other properties to accurately identify its format, regardless of its extension.



```
File Edit View Search Terminal Help
ubuntu@ip-10-10-56-192:~/Desktop/Malware$ ls
wannacry
ubuntu@ip-10-10-56-192:~/Desktop/Malware$
```



```
ubuntu@ip-10-10-56-192:~/Desktop/Malware$ file wannacry
wannacry: PE32 executable (GUI) Intel 80386, for MS Windows
ubuntu@ip-10-10-56-192:~/Desktop/Malware$
```

## Extracting Strings

One valuable method for analyzing a file is extracting readable text using the `strings` command. This command scans a file and lists human-readable sequences of characters, which can provide useful insights into its functionality.

Using strings is straightforward.

```
ubuntu@ip-10-10-56-192:~/Desktop/Malware$ strings wannacry |head -15
!This program cannot be run in DOS mode.
Rich
.text
.rdata
@.data
.rsrc
49t$
TVWj
PVVh
```

```
RegCloseKey
RegQueryValueExA
RegSetValueExA
RegCreateKeyW
CryptReleaseContext
CreateServiceA
CloseServiceHandle
StartServiceA
OpenServiceA
OpenSCManagerA
ADVAPI32.dll
SHELL32.dll
OLEAUT32.dll
WS2_32.dll
fclose
fwrite
fread
```

## Calculating File Hashes

File hashing generates a fixed-size unique identifier for a file, much like a **fingerprint**. This identifier can be used to verify file integrity and detect known malware samples. In malware analysis, hashes help analysts track malicious files and compare them against threat intelligence databases.

Commonly used hashing algorithms include:

- **MD5** (md5sum) – Fast but prone to collisions (not recommended for security-sensitive applications).
- **SHA-1** (sha1sum) – More secure than MD5 but still vulnerable to certain attacks.

- **SHA-256** (sha256sum) – Stronger and widely used for security purposes.

### AV Scans and VirusTotal

Using antivirus (AV) scans or searching for a file's hash on VirusTotal can help determine if a file is malicious. Security researchers classify malware based on its behavior, and these tools provide insights into known threats.

Best Practices for Online Scanning:

Prefer hash searches: Instead of uploading a file, search for its hash (MD5, SHA-1, or SHA-256) to check if it has already been analyzed.

Be cautious when uploading: Uploading malware samples to online scanners may expose sensitive information. Only do so if you fully understand the risks.

```
ubuntu@ip-10-10-56-192:~/Desktop/Malware$ md5sum wannacry
84c82835a5d21bbcf75a61706d8ab549  wannacry
ubuntu@ip-10-10-56-192:~/Desktop/Malware$
```

## Analyzing the PE Header with pecheck

The **pecheck** utility, available in the **Remnux VM**, is a powerful tool for inspecting the **Portable Executable (PE) header** of Windows binaries. This analysis helps in understanding the structure and characteristics of a suspicious executable.

### Key Insights from pecheck Output:

- **Section Details:** Displays sections like `.text`, `.rdata`, `.data`, and `.rsrc`, along with their entropy values, which can indicate potential obfuscation or packing.
- **File Hashes:** Automatically extracts various hashes (MD5, SHA-1, SHA-256) for easy comparison with threat databases.

- **Imported Functions:** Lists the **IMAGE\_IMPORT\_DESCRIPTOR**, showing functions the file imports from dynamic link libraries (DLLs). In the case of **WannaCry**, pecheck reveals imports from ADVAPI32.dll, a system library used for advanced Windows API functions.
- **Other Metadata:** Can display compile timestamps, entry points, and other useful indicators of potential malicious behavior.

```
ubuntu@ip-10-10-56-192:~/Desktop/Malware$ pecheck wannacry
PE check for 'wannacry':
Entropy: 7.995471 (Min=0.0, Max=8.0)
MD5      hash: 84c82835a5d21bbcf75a61706d8ab549
SHA-1    hash: 5ff465afaabcbf0150d1a3ab2c2e74f3a4426467
SHA-256  hash: ed01ebfbc9eb5bbea545af4d01bf5f1071661840480439c6e5babe8e080e41a
SHA-512  hash: 90723a50c20ba3643d625595fd6be8dcf88d70ff7f4b4719a88f055d5b3149a
.text entropy: 6.404235 (Min=0.0, Max=8.0)
.rdata entropy: 6.663571 (Min=0.0, Max=8.0)
.data entropy: 4.455750 (Min=0.0, Max=8.0)
.rsrc entropy: 7.999868 (Min=0.0, Max=8.0)
```

## Analyzing Malware with Hybrid Analysis

**Hybrid Analysis** is an online sandbox that runs suspicious files in a controlled environment and generates reports. Instead of uploading a file, it's safer to **search for its hash** to check if it has already been analyzed.

For known malware like **WannaCry**, existing reports provide:

- **Threat Score & Verdict** – Clearly marked as *malicious* with a **high AV detection rate**.
- **Behavior Overview** – Summarizes actions and links them to **MITRE ATT&CK techniques**.
- **Process Execution Details** – Shows **cmd.exe running scripts** to delete backups, a common ransomware tactic.
- **Network Activity** – Displays **suspicious connections** made by the malware.
- **Extracted Strings & Files** – Helps identify scripts and commands used by the malware.
- **Community Comments** – Insights from researchers who analyzed the same file.

HYBRID ANALYSIS

SandboxQuick ScansFile CollectionsResourcesRequest Info

IP, Domain, Hash...

More

Search results for 84c82835a5d21bbcf75a61706d8ab549

Login to Download all DNS Requests (CSV)Login to Download all Contacted Hosts (CSV)

Multi-ProcessExtracted FilesSample not sharedNetwork TrafficTOR analysisDecrypted SSL traffic

Copy hashesSelect all

Timestamp	Input	Threat level	Analysis Summary	Countries	Environment	Action
February 27th 2025 14:22:20 (UTC)	ed01ebfbc9eb5bbea545af4d01bf5f071661840480439c6e5babe8e080e41aa.exe PE32 executable (GUI) Intel 80386, for MS Windows ed01ebfbc9eb5bbea545af4d01bf5f071661840480439c6e5babe8e080e41aa	malicious	AV Detection: 98% Trojan.Ransom.WannaCryptor #tag #wannacry #worm #ransomware #wannacrypt0r #wcry #goat #isfb #papras #ursnf #banker #emotet #trookit #backdoor #coinminer #exploit #hacktool #maldoc #metasploit #meterpreter #plugx #windows-server-utility #adwind #agenttesta #alenspy #chantor #chtholic #crider #crimson #darkcomet #dofal #dridex #dyro #dyroza #farsit #gootit #hancitor #hawkeye #infostealer #keylogger #tikibot #msal #nanocore #netwire #neutrino #neverquest #poisonivy #pony #predator #qakbot #smokeloader #stealer	-	quickscan	

Falcon Sandbox Reports (49)

Characteristics LegendShow All As ListSubmit

Not all reports are visible. 8 malicious and 35 error reports are hidden.

Windows 11 64 bit

WannaCry.exe.sample  
December 22nd 2024 05:15:54 (UTC)

Malicious

Threat Score: 100/100

Labeled As: Trojan.Ransom.Wa...

Indicators: 6 51 187

Characteristics: 0 2 0

Windows 11 64 bit

WannaCry.exe.sample  
December 15th 2024 20:06:07 (UTC)

Malicious

Threat Score: 100/100

Labeled As: Trojan.Ransom.Wa...

Indicators: 6 48 184

Characteristics: 0 2 0

Windows 11 64 bit

WannaCry.exe.sample  
October 31st 2024 17:08:56 (UTC)

Malicious

Threat Score: 100/100

Labeled As: Trojan.Ransom.Wa...

Indicators: 7 47 174

Characteristics: 0 2 0

Windows 10 64 bit

ed01ebfbc9eb5bbea545af4d01bf5...  
October 5th 2024 13:46:11 (UTC)

Windows 10 64 bit

ed01ebfbc9eb5bbea545af4d01bf5...  
May 1st 2024 05:45:35 (UTC)

Windows 10 64 bit

owo\_im\_not\_ransomware\_xd.exe  
May 15th 2023 06:49:15 (UTC)

Anal  
Anti-  
Falc  
Rela  
Inci  
Com  
Back

**Tip:** Click an analysed process below to view more details.

- WannaCry.exe.sampleCry.exe (PID: 9448) 21/24
  - icacds.exe icacds . /grant Everyone:F /T /C /Q (PID: 10084) Hash Seen Before
  - attrib.exe attrib +h . (PID: 10348) Hash Seen Before
  - taskcd.exe (PID: 9444) Hash Seen Before
  - cmd.exe %WINDIR%\system32\cmd.exe /c 16781734874223.bat (PID: 5660) Hash Seen Before
    - cscript.exe //nologo m.vbs (PID: 8596) Hash Seen Before
  - attrib.exe attrib +h +s %SAMPLEDIR%\\$RECYCLE (PID: 7476) Hash Seen Before
  - taskcd.exe (PID: 9048) Hash Seen Before
  - taskcd.exe (PID: 8516) Hash Seen Before
  - taskcd.exe (PID: 9736)
  - taskcd.exe (PID: 1424)
  - taskcd.exe (PID: 7652)
  - taskcd.exe (PID: 9864)
  - taskcd.exe (PID: 4656)
  - taskcd.exe (PID: 8944)
  - taskcd.exe (PID: 7392)
  - taskcd.exe (PID: 4368)
  - taskcd.exe (PID: 5440)
  - taskcd.exe (PID: 8768)

## All Details: Off

All Strings (1807)	Interesting (1256)	WannaCry.exe.sample (53...)	u.wnry (1)	taskse.exe (1)	taskdl.exe (1)	cscript.exe:8596 (163)	WannaCry.exe.sample.exe...
taskdl.exe:9444 (46)	cmd.exe:5660 (17)	@WanaDecryptor@exe.l...	screen_1.png (1)	16781734874223.bat (1)	fwnry (1)	@Please_Read_Me@txt (1)	rwnry (1)
00000000.pkx (1)	m.vbs (1)						

[illegible]

While these approaches are effective in most cases, **advanced malware may use obfuscation, anti-analysis, or sandbox evasion techniques**, making detection more



challenging. Continuous learning, improved security tools, and collaborative research are essential to staying ahead of evolving threats.

By leveraging these techniques, security teams can enhance threat detection, improve incident response, and strengthen overall cybersecurity defenses.