

MAURICE LAMBERT

# ANTIVIRUS BYPASS

**1. IS IT REALLY AN ANTIVIRUS ?**

**2. WHAT ABOUT UNKNOWN MALWARE ?**

**3. HOW « ANTIVIRUS » WORKS ?**

**4. ARE HASHES GOOD SIGNATURES ?**

**5. GENERIC PATTERNS**

**6. PACKER**

**7. OBFUSCATION**

**8. ENTROPY BYPASS**

**9. EXECUTABLE FORMAT EXPLOITATION**

**10. ARE ANTIVIRUSES USELESS ?**

**ANTIVIRUS BYPASS**

**DO YOU THINK THAT  
AN "ANTIVIRUS" IS  
REALLY A  
PROTECTION  
AGAINST VIRUSES ?**



# IS IT REALLY AN ANTIVIRUS ?

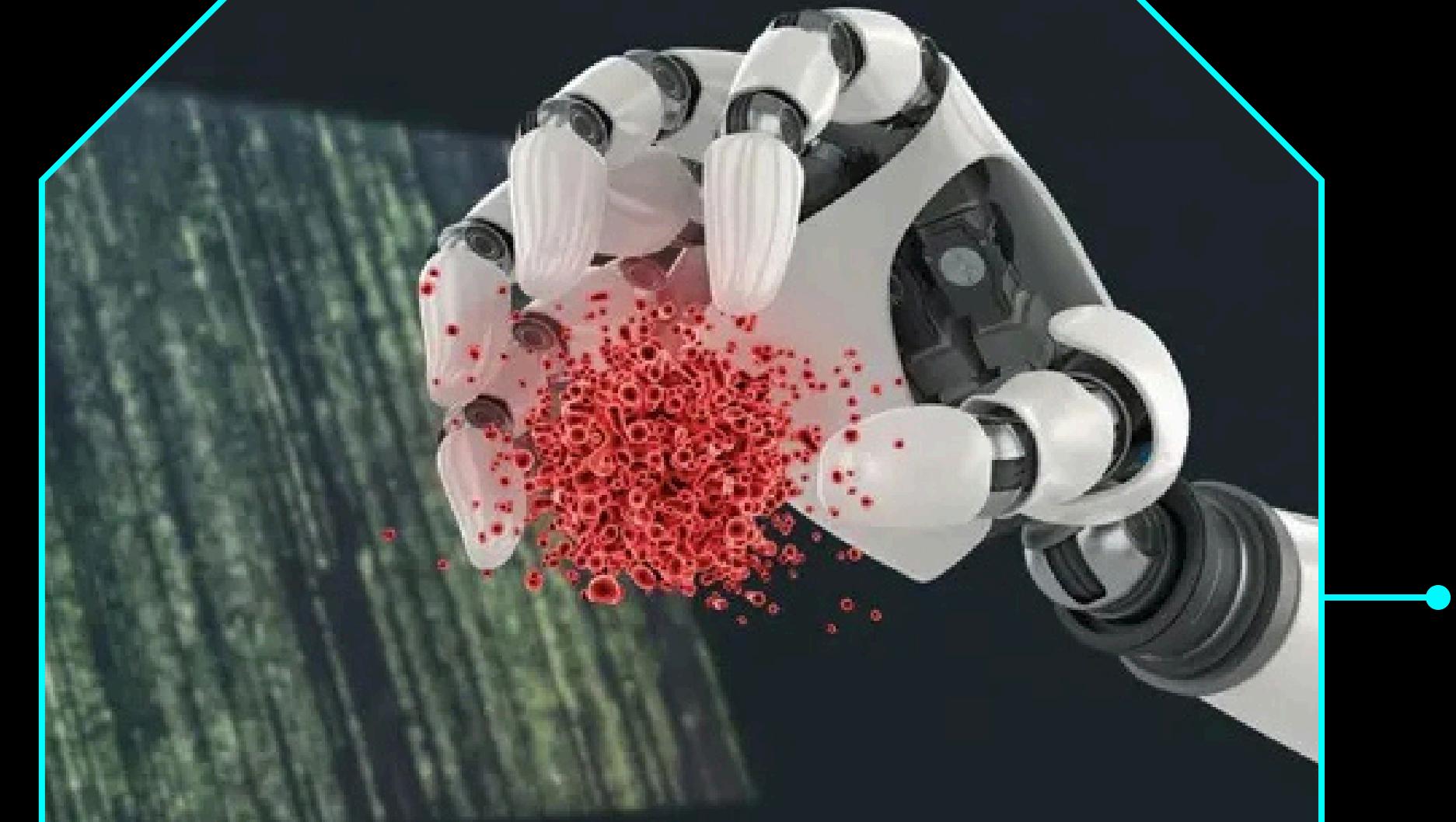
**IT'S NOT AN ANTIVIRUS BUT AN ANTIMALWARE**

The term "antivirus" is often used instead of "antimalware" due to historical reasons and public familiarity, despite antimalware being a more accurate term. Antivirus software was first developed to combat computer viruses, which were the primary digital threats in the early days of computing. As the landscape of digital threats evolved, the term "antivirus" remained popular even as the software expanded its capabilities.

# ANTIVIRUS USAGES

## CHECK FILES ON DEMAND

- Check a specific selection of files
- Check the full filesystem
- Check each X time (scheduled tasks)
- Don't check a file when is written
- What about deleted file ?
  - Easy way to bypass basic antivirus: delete malware after execution
    - No persistence
    - Download the malware for each execution
- Role: identify threats (antivirus don't protect, there is no active protection in antivirus, this is the role for EPP)
- Goal: identify and delete malwares (malicious softwares)



**ANTIVIRUS BYPASS**

**DO YOU THINK  
THAT AN  
ANTIVIRUS CAN  
DETECT UNKNOWN  
MALWARE ?**

# UNKNOWN MALWARE DETECTION

## SHOULD AN ANTIVIRUS DETECT UNKNOWN MALWARE ?

A basic antivirus may struggle to detect unknown malware consistently. While traditional antivirus software primarily relies on signature-based detection, which is effective for known threats, it has limitations when dealing with new, unknown malwares.



# HOW «ANTIVIRUS» WORKS?

## SIGNATURES AND REPUTATION

Antivirus software works by scanning files, programs, and network traffic to detect and remove malicious code.

- Signature-based detection
  - Hashes
  - Generic patterns
- Reputation
- Quarantine and removal



**ANTIVIRUS BYPASS**

**DO YOU THINK  
THAT A HASH  
SIGNATURE IS A  
GOOD SIGNATURE ?**

# ARE HASHES GOOD SIGNATURES ?



2597322a49a6252445ca4c8d713320b238113b3b8fd8a2d6fc1088a5934cee0e

54/71 security vendors flagged this file as malicious

WndResizerApp.exe

Community Score 54 / 71

Detection Details Relations Behavior Community 7

Join our Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

Basic properties

	MD5	SHA-1	SHA-256	Vhash	Authentihash	ImpHash	SSDeep	TLSH
e758e07113016aca55d9eda2b0ffeebe	8c1e63a01148e20085d418c0b23021bc5eca0709	2597322a49a6252445ca4c8d713320b238113b3b8fd8a2d6fc1088a5934cee0e	11705677f5f57560b012z61009d6z150c5za0600dc3z17z13	50f87a543288017a9228574417a9c18d57f57d1bae25a028079d34e1ef26175f	607d0c9fdb370b1ce70573304bcd084	196608>JWx2zpdra2YbT8yN+8Mne6nd7g25FjZC8OHTRbFd/Or+GvJbU9Rdf/kuFLOyomFl:JYCrdiNTF5nZ9C8Ud29JuFT172C601A03CDA0026F0AF11716AA9FF79E12F6F722F3525535150BA19FD322436E14F6A		

## HASHES TYPES

- Cryptographic Hashes
  - md5
  - sha1
  - sha256
  - sha512
  - sha3
  - blake2
  - ...
- Imphash (Import Hash)
- SPHF (Similarity Preserving Hash Functions)
  - SSDeep Hash (Context Triggered Piecewise Hashing)
  - TLSH (Trend Micro Locality Sensitive Hash)
  - VHash
- Authentihash
- ...

# GENERIC PATTERNS

## FUNCTIONS, ENTROPY, STRINGS...

- Imported / exported functions
  - Memory access/permissions
  - Debug functions
  - Network functions
  - Command line functions
  - COM "Interface" functions (DIIRegisterServer)
  - ...
- Suspicious shannon entropy
  - Very high (greater than 7.2)
  - Very low (smaller than 2 or 3)
- File size
  - CIA maldev rule: executable smaller than 150KB
  - Lot of malwares use more than 1 GB overlay
- Strings
  - Bitcoin wallet
  - IOC: IP, Domain, URL
  - Number of strings
- Section names and characteristics





# SUMMARY OF THE FIRST PART

## ANALOGY TO THE ANTIVIRUS BEHAVIOUR

An antivirus behaviour is similar to the following discussion, with a file in the role of the girl and the antivirus in the role of the friend

- You: Do you think she's a good girl?
- Friend: She is not on the list of girls I know as malicious
- You: Okay but you don't know if it's a good girl.
- Friend: Yeah, yes, she's pretty.
- You: Yes but that doesn't answer my question...
- Friend: She is fine because her appearance does not look suspicious.

**ANTIVIRUS BYPASS**

**IT'S TIME TO  
BYPASS !**



# PACKER

## KNOWN AND TRUSTED PACKERS CUSTOM PACKER

- Used in production
- Modify sections name
- Compress
  - Reduce size
  - Greater entropy score
- Best antivirus decompress it and analyse data

- Bypass antivirus
- Require access to suspicious calls
  - Can bypass suspicious imports
- Most of the time the entropy increases (encryption)
- Can modify sections name

The maldev CIA posture: don't use packer.

# OBFUSCATION

## SCRIPTS OBFUSCATION

- Random variables name
- Usage of eval or exec functions
- Hide the code structure
- Encoding or/and encryption
- Add useless code
- Hide constants value

## EXECUTABLE OBFUSCATION

- Add useless instructions

The maldev CIA posture: don't use obfuscation in executable.



```
<?php //>
.... 4d2c 6ae2 9282
.... /*$a/*A.
.... a90a 0a2d 603d 364d 4164
.... zVO=MNEIM,j
.... a0a0 e292 82e2 8aa5 cea6 e29c
.... 3457 6947 3c6d 6961 5d4b 4134 2a2f
.... 98b8 e29c aee2 9799 e29e bbe2 878f
.... a4 29e a90a 0a2a 2f28 227e 222c 2f2a
.... e29e 9ee2 9694 e297 b0e2 98a8 e289 a3e2
.... 80bf e286 94e2 88b5 e291 a7e2 9ea2 e28b
.... c2 aee2 9382 e295 a7e3 88a4 e288 8be2 93ad
.... 45ce a6e2 9ca4 e29e a90a 0a2d 603d 364d 4164
.... 722a 2f2f 2fe2 92a5 e29e a2e2 9eb6 c2ae e293
.... 0 e287 8f0a 2f2a 0a0a e292 82e2 8aa5 cea6 e29c
.... 2a 4a3c 3421 7a52 7661 647a 322e 2d7b 7b62 3f2a
.... 8a 81ef b981 e298 b8e2 9cae e297 99e2 9ebb e287
.... aa5 cea6 e29c a4e2 9ea9 0a0a 2a2f 3d2f 2a3d 7037
.... 078e e28a 92e3 8a81 efb9 81e2 98b8 e29c aee2 9799
.... 6ae2 9282 e28a a5ce a6e2 9ca4 e29e a90a 0a2a 2f24
.... 92a2 e28a 870a 0a2a 2f2f 2fe2 92a5 e29e a2e2 9eb6
.... e297 99e2 9ebb e287 8f0a 2f2a 0a0a e292 82e2 8aa5
.... a 2a2f 7b2f 2a2d 7643 3035 6c63 2a2f 2f2f e292 a5e2
.... 98b8 e29c aee2 9799 e29e bbe2 878f 0a2f 2a0a 0ae2
.... 4 e29e a90a 0a2a 2f24 612f 2a63 6d76 6242 372a 2f2f
.... a 81ef b981 e298 b8e2 9cae e297 99e2 9ebb e287 8f0a
.... 5 cea6 e29c a4e2 9ea9 0a0a 2a2f 5b31 2b33 305d 2f2a
.... d e296 b5e2 80ba e29c 8fe2 978e e28a 92e3 8a81 efb9
.... 4 7a56 4f3d 4d4e 4549 4d2c 6ae2 9282 e28a a5ce a6e2
.... 1 e285 91e2 96a2 e295 85ef bd80 e295 a9e2 988f e29d
.... F e29e 99e2 8ab7 e29e 9fe2 8aa1 e285 91e2 96a2 e295
.... 9eb6 c2ae e293 82e2 95a7 e388 a4e2 888b e293 ade2
.... 8aa5 cea6 e29c a4e2 9ea9 0a0a 2d60 3d36 4d41 647a
.... 6441 6c70 792a 2f2f 2fe2 92a5 e29e a2e2 9eb6 c2ae
.... 9e2 9ebb e287 8f0a 2f2a 0a0a e292 82e2 8aa5 cea6
.... 032 362b 3333 5d2f 2a3a 474a 3378 4521 785f 6228
.... 82 8a92 e38a 81ef b981 e298 b8e2 9cae e297 99e2
.... 2 82e2 8aa5 cea6 e29c a4e2 9ea9 0a0a 2a2f 2e24
.... bbe2 8a9f 0a0a 7e42 4d62 583b e291 aee2 95a4
.... 92a5 e29e a2e2 9eb6 c2ae e293 82e2 95a7 e388
.... 0a e292 82e2 8aa5 cea6 e29c a4e2 9ea9 0a0a
.... 2 8992 e297 bde2 979b e28f a5e2 9994 e285
.... b9a3 e297 abe2 86a9 e380 85e2 8090 e289
.... 295 9623 cb9c c2a1 e28b bae2 8b97 e29e
.... a 0a0a 2a2f 2f2f e292 a5e2 9ea2 e29e
.... bbe2 878f 0a2f 2a0a 0ae2 9282 e28a
.... 3a 5738 7776 3d3d 7049 6c43 4758
.... 38a 81ef b981 e298 b8e2 9cae
.... cea6 e29c a4e2 9ea9 0a0a
.... 4c 2471 2b28 3224 5624
.... 2 95a7 e388 a4e2
.... 0a0a 2d60
```

**ANTIVIRUS BYPASS**

# **NOT DOCUMENTED TIPS AND TRICKS**



# ENTROPY BYPASS

## PADDING

- Documented
- Problem: significant increase in file size
- Resolve: global file entropy
- Problem: don't modify the high entropy sector

## SPECIFIC ENCODING/ENCRYPTION

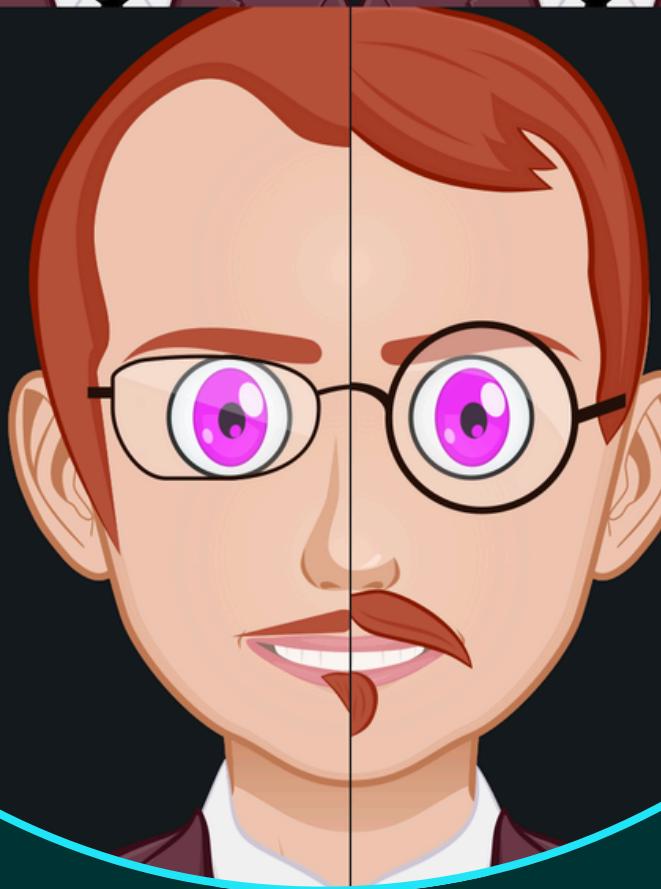
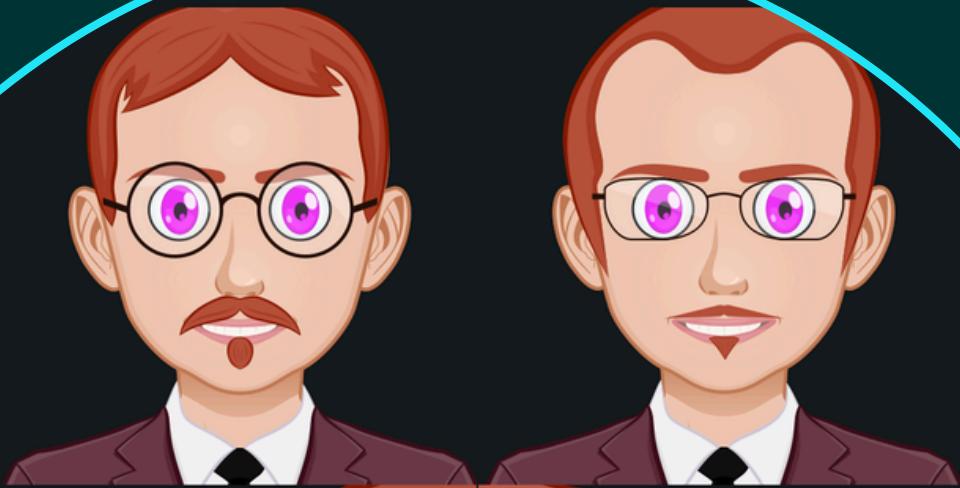
- I have never found any documentation online
- Problem: small increase in file size
- Resolve: high entropy sector

Implementation and POC: [EntropyEncoding](#).

# EXECUTABLE FORMAT EXPLOITATION

## MODIFY UNUSED FIELDS, "METADATA" FIELDS AND STRINGS

- PE
  - DOS STUB
  - Rich headers
  - Timestamps
  - Fields reserved for future uses
  - Filename
  - Copyright
  - Description
  - ...
- ELF
  - Fields reserved for future uses
  - "Usage" in help message
  - Copyright message
  - ...



**ANTIVIRUS BYPASS**

**TESTS TIME !**

# VIRUS TEST

## BASIC EXECUTABLE INFECTION

- Payload replication in other files
- Usages
  - Persistence
  - Defense evasion
  - "Analyst evasion"
- Virus type
  - Executable/DLL
  - Script
    - Admin scripts
    - Server scripts
  - Office documents
    - doc/docm and other microsoft office documents
    - PDF
    - RTF
  - System file
    - LNK
  - Archive files (add malicious files in trusted archive)
  - ...
- Infect the file using Pelinjector



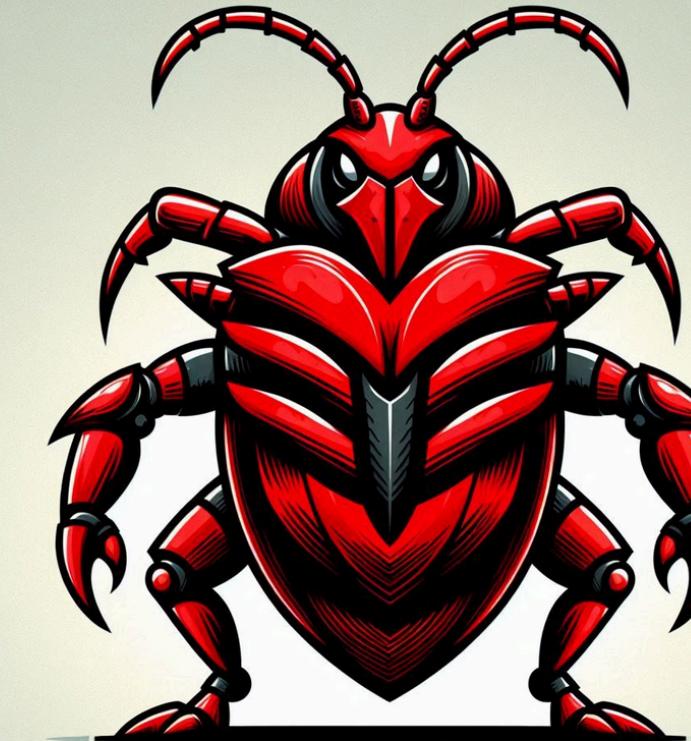
# UNKNOWN MALWARE DETECTION

## DOWNLOAD CUSTOM MALWARES

- Download non-obfuscated/non-packed malwares files
  - [Spyware](#)
  - [Keylogger](#)
  - MbrWiper
  - [Ransomware](#)
  - ...



# FILELESS MALWARE EXECUTION



## PE LOADER

- Python PE Loader (PyPeUrlLoader)
  - Download the malware over HTTP(S)
  - Optional file decryption (useful to bypass firewall)
  - Stock the file in memory (don't write it on the disk)
  - Load it using PyPeLoader as the Windows linker
  - Execute the malware from entry point
- Don't write any malware on the disk
- No problem with entropy detection
- Require an internet/network access
- No persistence

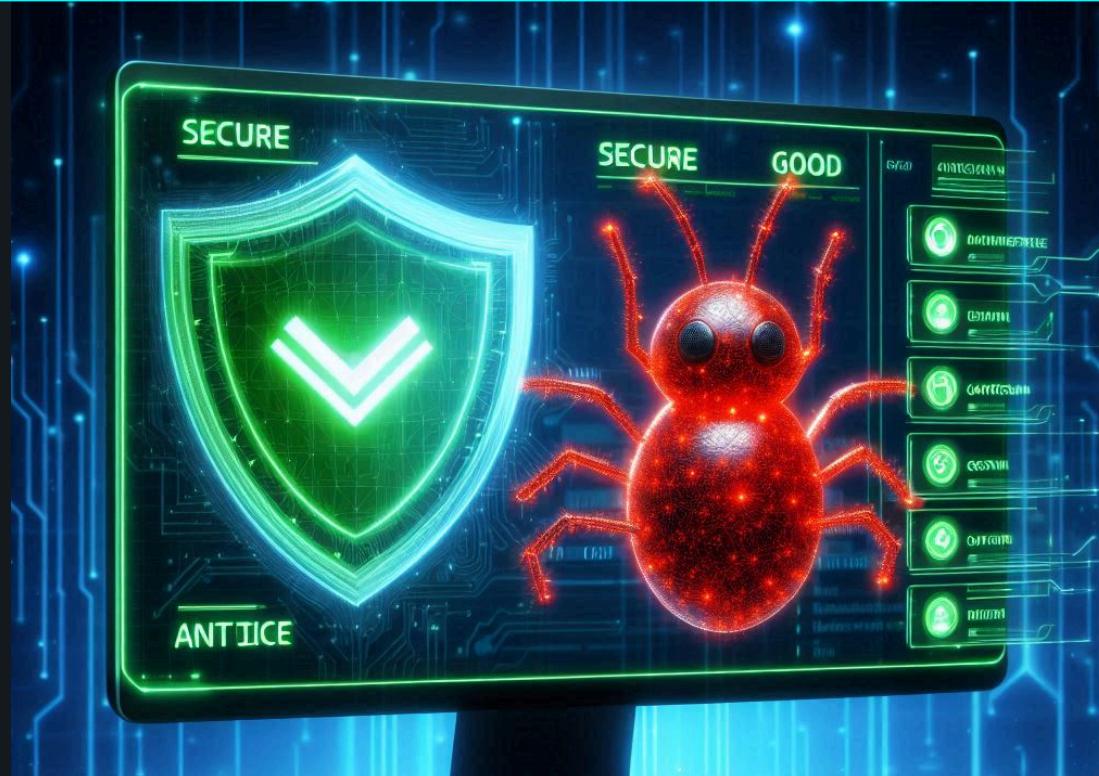
# PACKER DETECTION

## USING A PYTHON PACKER

- Using [PyPePacker](#)
  - Compress
  - Encryption
  - Reduce entropy (using [EntropyEncoding](#))
  - Possible conversion to PE (with high entropy in the overlay)
  - Possible obfuscation using [PyObfuscator](#)



# KNOWN HACKTOOL FILE DETECTION



## BYPASS USING EXE FILE STRUCTURE

- Download [ChromePasswordsStealer](#)
  - [Antivirus detection](#)
- Download it using [BypassHash](#) executable
  - No antivirus detection
  - Run the executable to verify that it works well
  - Compare hashes
    - Cryptographic hashes
    - ssdeep
  - Compare size

**ANTIVIRUS BYPASS**

# **THE END: ARE ANTIVIRUSES USELESS?**



# ANTIVIRUS TODAY

## USE CASES FOR ANTIVIRUS TODAY

- EPP use antivirus signatures in real time
- EPP use antivirus signatures in memory
- Common antivirus bypass techniques use suspicious items monitored by EDR
- A good signature system generates fewer false positives than newer technologies (like machine learning, correlations, ...)
- A good signature system generates fewer bugs
- A good signature system requires less maintenance
- A good, up-to-date signature system protects you from known attack campaigns

## RECOMMENDATION FOR PERSONAL WINDOWS

- Use microsoft defender because there is an EPP integrated (you pay the EPP with the Windows license).

**ANTIVIRUS BYPASS**

# **APPENDICES**

# ImpHash

- Resolving ordinals to function names when they appear
- Converting both DLL names and function names to all lowercase
- Removing the file extensions from imported module names
- Building and storing the lowercased string . in an ordered list
- Generating the MD5 hash of the ordered list

# PE format

- IMAGE\_DOS\_HEADER
  - DOS Stub
  - Rich headers
  - IMAGE\_NT\_HEADERS
    - IMAGE\_FILE\_HEADER
    - IMAGE\_OPTIONAL\_HEADER32 | IMAGE\_OPTIONAL\_HEADER64
      - IMAGE\_DATA\_DIRECTORY
  - IMAGE\_SECTION\_HEADER
  - <section 1>
  - <section 2>
  - ...
  - <section N>
  - Overlay
- 
- Instructions
  - Imports (ILT, IAT)
  - Exports (EAT)
  - Relocations
  - Resources
  - Initialized data
  - Uninitialized data
  - ...

# ELF headers

- ELF Header
- ELF Section Header
- <section 1>
- <section 2>
- ...
- <section N>
- Overlay
- 
- Instructions
- Imports
- Exports
- Relocations
- Metadata
- Initialized data
- Uninitialized data
- ...

# Windows loader

- Parse the PE file headers
- Load the PE file sections into memory
- Process the import table
  - Load required DLLs
  - Resolve external function addresses
  - Overwrite function pointers
- Apply base relocations
- Set permissions for each section
- Start execution at entry point address