NCSA Threat Hunting

Who am I

Whoami

Instructor Information

Sumedt Jitpukdebodin

Cybersecurity Specialist, **Secure-D Center Company**
Content Creator, **SEC Playground Company**

Professional Certifications
GIAC Penetration Tester (GPEN)
Offensive Security Certified Professional (OSCP)
Certified Penetration Testing Engineer(CPTE)
eLearnSecurity Certified Professional Penetration Tester (eCPPT)
eLearnSecurity Web Application Penetration Tester (eWPT)
CompTIA CySA+
And more….

Education
Bachelor of Computer Engineering, Kasetsart University
Master of Information System Security, Mahanakorn University

Experience
11 years experience in
- Cyber Security Incident response
- Penetration Testing and Vulnerability Assessment

Community
OWASP Thailand Chapter Committee, Admin Group of 2600 Thailand

Threat Hunting

Threat Hunting

What is Hunting

Hunting is the process of looking for interesting events that are not defined as malicious by existing automated tools

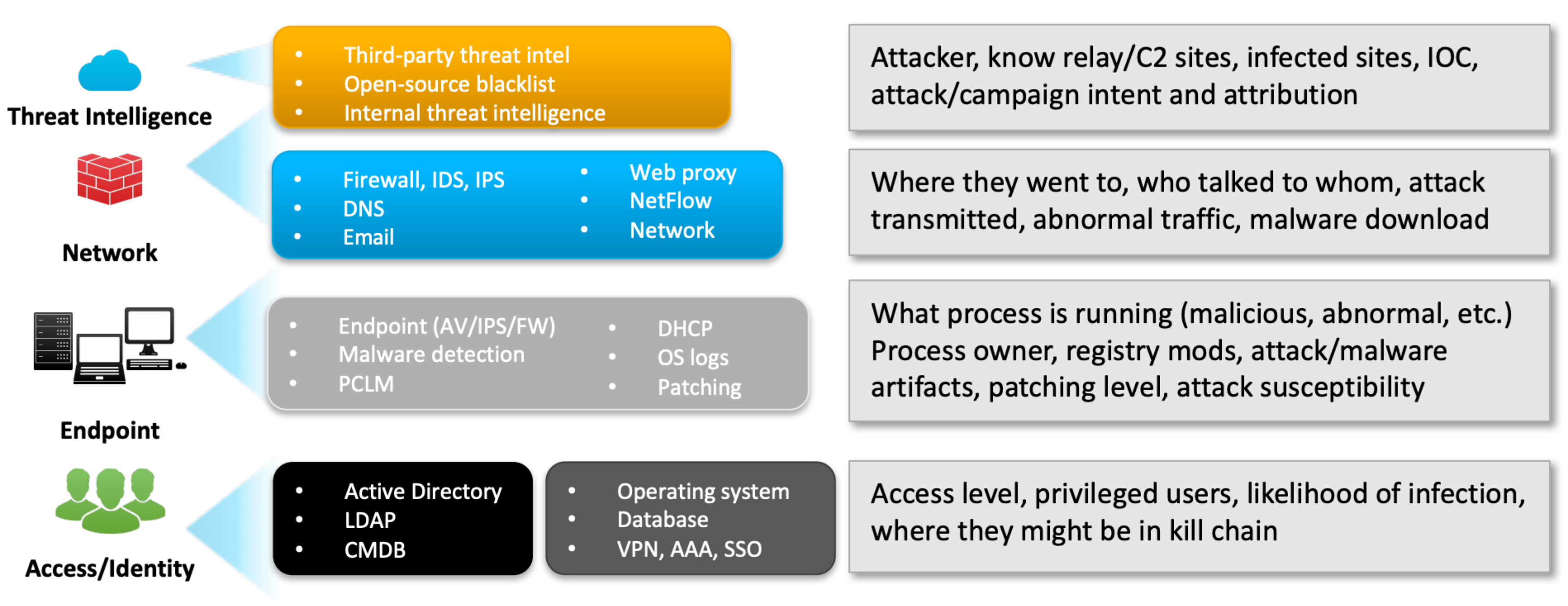
It uses the knowledge, tools, data, and experience that exists within an organization to determine if events are associated with an attacker or innocuous

Mature Security Programs have functional

- Threat Intelligence
- Signature Tuning / Sensor Management
- Alert Monitoring
- Incident Response

Threat Hunting

Input of Threat Hunting



Threat Intelligence

- Third-party threat intel
- Open-source blacklist
- Internal threat intelligence

Attacker, know relay/C2 sites, infected sites, IOC, attack/campaign intent and attribution

Network

- Firewall, IDS, IPS
- DNS
- Email
- Web proxy
- NetFlow
- Network

Where they went to, who talked to whom, attack transmitted, abnormal traffic, malware download

Endpoint

- Endpoint (AV/IPS/FW)
- Malware detection
- PCLM
- DHCP
- OS logs
- Patching

What process is running (malicious, abnormal, etc.) Process owner, registry mods, attack/malware artifacts, patching level, attack susceptibility

Access/Identity

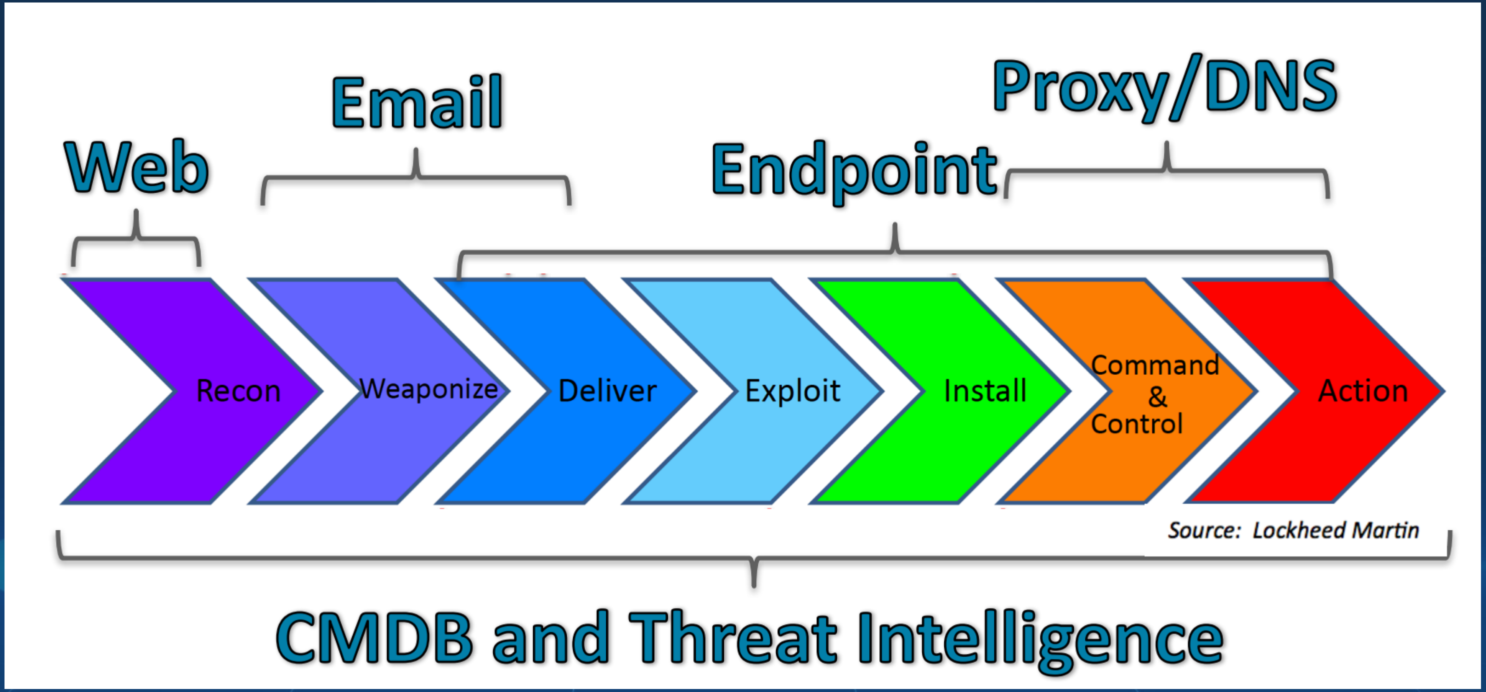
- Active Directory
- LDAP
- CMDB
- Operating system
- Database
- VPN, AAA, SSO

Access level, privileged users, likelihood of infection, where they might be in kill chain

Threat Hunting Workshop by Splunk

Threat Hunting

Input of Threat Hunting

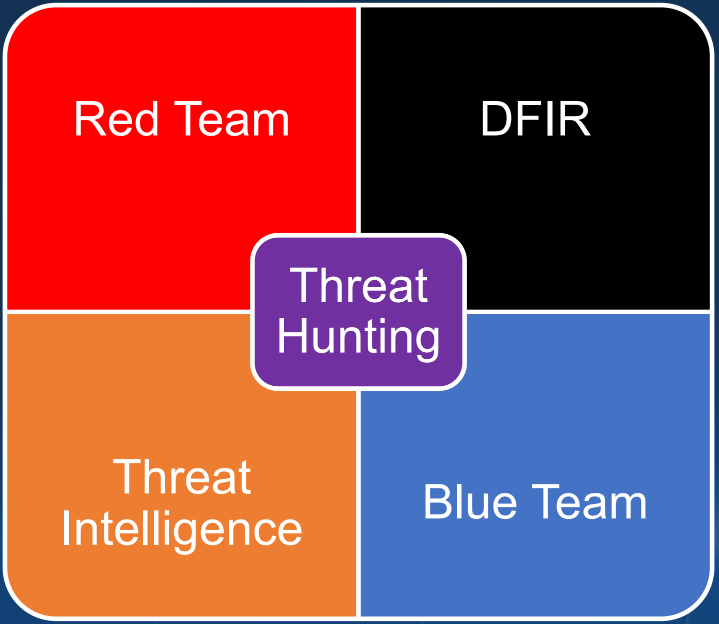


Source: Lockheed Martin

Threat Hunting Workshop by Splunk

Threat Hunting

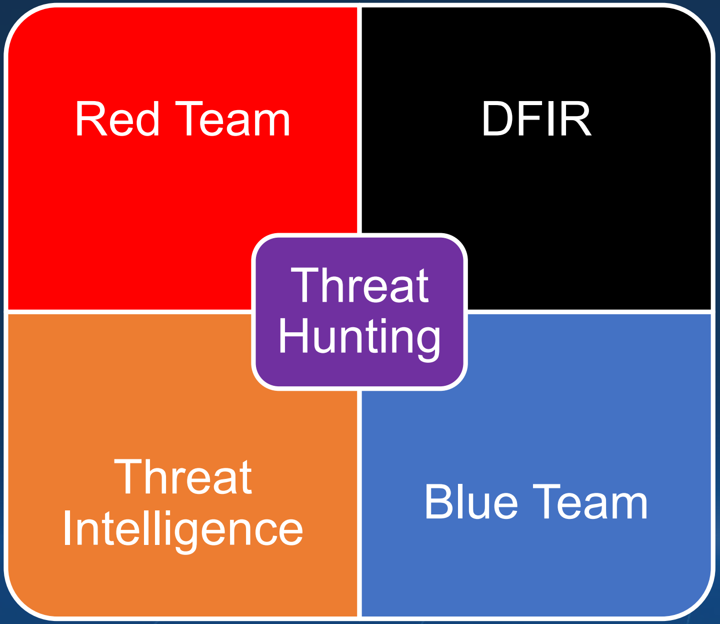
Hunting Skillsets /Abilities



- **Operating System**
  - Knowledge of Operating System internals, OS security mechanisms, knowledge of typical security issues of different operating systems
- **Network Architecture**
  - Understanding how computer networks work, OSI Layer, knowledge of TCP/IP, knowledge of basic protocols (DNS, DHCP, HTTP, SMTP, FTP, SMB);
- **Attack Methods/TTPs / Cyber Kill Chain**
  - Knowledge of specific attack vectors, understanding how an attacker attempts to penetrate your network, which attack vectors and tools he/she can use on different attack stages;
- **Analytical Mindset**
  - Having a mindset of curiosity, Ability to generate and investigate hypotheses. As an analyst, it's increasingly important to be specific in what questions you're looking to answer during threat hunting.

Threat Hunting

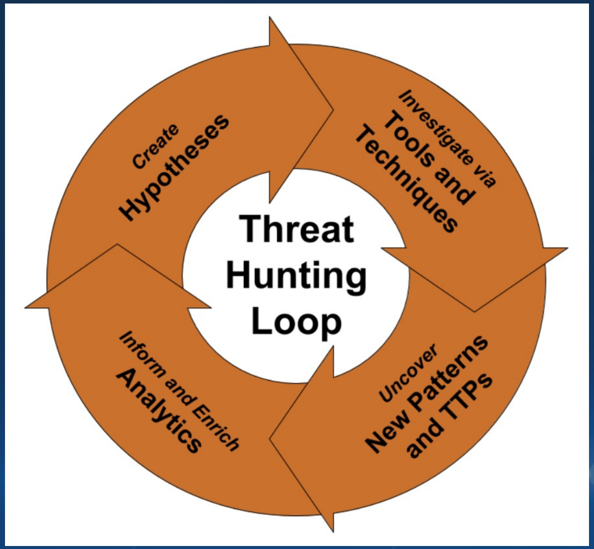
Hunting Skillsets /Abilities#2



- **Log Analysis**
  - Knowledge of different log sources and event types generated by different sources, the ability to analyze logs for anomalies and pivot between data sources to see the big picture;
- **Network Analysis**
  - The ability to read and understand packet capture data and determine the malicious nature of network traffic;
- **Cyber Threat Intelligence**
  - Having a skill and knowledge to leverage threat intelligence for threat hunting purposes, always seek for new information from threat intelligence report
- **Malware Analysis**
  - Malware analysis a highly specialized skill that aims to determine the origin and purpose of an identified instance of malware.

Threat Hunting

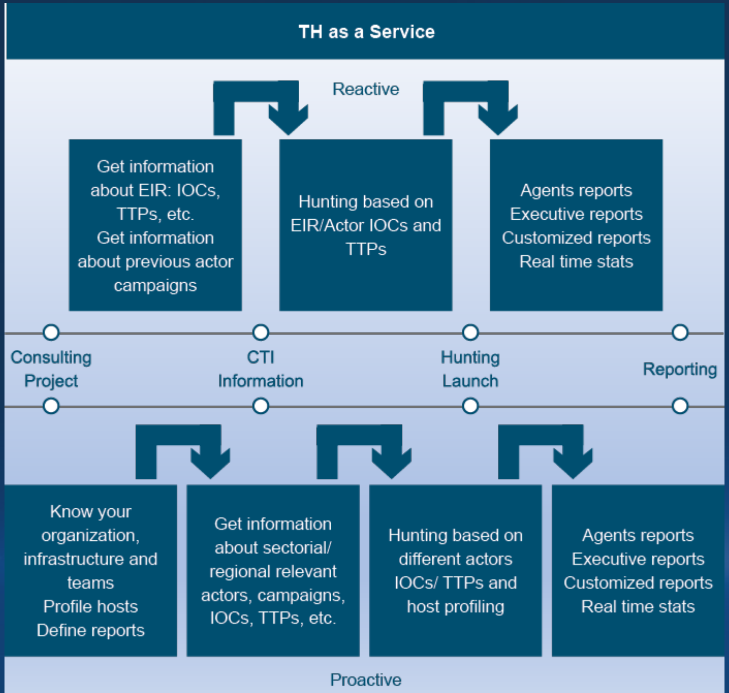
Threat Hunting Methodology


Threat Hunting Loop
Create Hypotheses
Investigate via Tools and Techniques
Uncover New Patterns and TTPs
Inform and Enrich Analytics

- **Create Hypotheses**
  - Threat Hunting begins with questions, such as "How would a threat actor infiltrate our infrastructure?"
- **Investigation via Tools and Techniques**
  - Need to be tested using all the relevant tools and techniques. The importance of Data sources and detection engineering capability from the organization, determine the result of this process.
- **Uncover new patterns and TTPs**
  - Even if the hypotheses result is not proven, It does not necessarily mean that no malicious activity is present or the hunters create a wrong hypotheses. It can be the current visibility in the organization is not enough or the tools that used by threat hunters is not good enough to help them to investigate the case.
- **Inform and Enrich Analytics**
  - Successful hunting process and then should be automated to make the efficient process for the threat hunters to reduce Threat Hunting team's time and to limit them from continuously repeating the same process

Threat Hunting

Types of Hunting


TH as a Service
Reactive
Get information about EIR: IOCs, TTPs, etc. Get information about previous actor campaigns
Hunting based on EIR/Actor IOCs and TTPs
Agents reports Executive reports Customized reports Real time stats
Consulting Project
CTI Information
Hunting Launch
Reporting
Know your organization, infrastructure and teams Profile hosts Define reports
Get information about sectorial/ regional relevant actors, campaigns, IOCs, TTPs, etc.
Hunting based on different actors IOCs/ TTPs and host profiling
Agents reports Executive reports Customized reports Real time stats
Proactive

Reactive Hunting

- Hunting activity generated while investigating an event
- Example:
  - Discover a malicious totallylegit.exe while investigating an alert for totallynotavirus.exe. Create a Hunt for totallylegit.exe.

Proactive Hunting

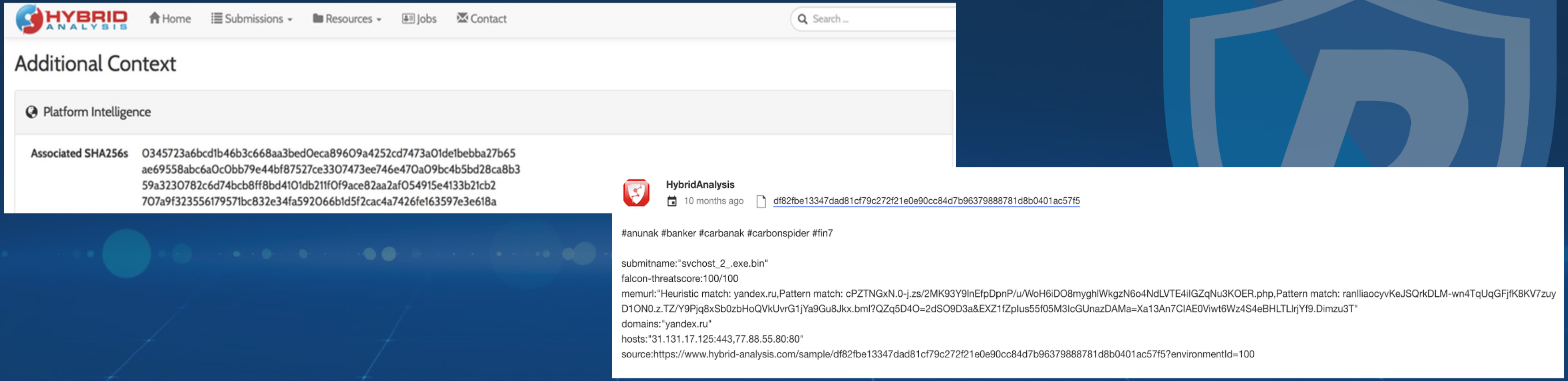
- Hunting activity based on Hunting Goal
- Example:
  - Search the network for large files traversing the ingress/egress points of the network

Pyramid of Pain

Pyramid of Pain

Hash Value

- **Description**: These are unique identifiers for files, often generated using algorithms like MD5, SHA-1, or SHA-256.
- **Ease for Adversaries**: Easy to change by modifying the file slightly.
- **Pain for Adversaries**: Low. Adversaries can easily generate new hash values by altering files.
- **Use Case**: Good for identifying known malware samples but not effective for advanced threats.



Additional Context

Platform Intelligence

Associated SHA256s 0345723a6bcd1b46b3c668aa3bed0eca89609a4252cd7473a01de1bebba27b65
ae69558abc6a0c0bb79e44bf87527ce3307473ee746e470a09bc4b5bd28ca8b3
59a3230782c6d74bcb8ff8bd4101db211f0f9ace82aa2af054915e4133b21cb2
707a9f323556179571bc832e34fa592066b1d5f2cac4a7426fe163597e3e618a

HybridAnalysis
10 months ago   df82fbe13347dad81cf79c272f21e0e90cc84d7b96379888781d8b0401ac57f5

```
#anunak #banker #carbanak #carbonspider #fin7

submitname:"svchost_2_.exe.bin"
falcon-threatscore:100/100
memurl:"Heuristic match: yandex.ru,Pattern match: cPZTNGxN.0-j.zs/2MK93Y9lnEfpDpnP/u/WoH6iDO8myghlWkgzN6o4NdLVTE4iIGZqNu3KOER.php,Pattern match: ranlliaocyvKeJSQrkDLM-wn4TqUqGFjfK8KV7zuy
D1ON0.z.TZ/Y9Pjq8xSb0zbHoQVkUvrG1jYa9Gu8Jkx.bml?QZq5D4O=2dSO9D3a&EXZ1fZplus55f05M3IcGUnazDAMa=Xa13An7CIAE0Viwt6Wz4S4eBHLTLlrjYf9.Dimzu3T"
domains:"yandex.ru"
hosts:"31.131.17.125:443,77.88.55.80:80"
source:https://www.hybrid-analysis.com/sample/df82fbe13347dad81cf79c272f21e0e90cc84d7b96379888781d8b0401ac57f5?environmentId=100
```

Pyramid of Pain

Hash Value

Property of Hashing:

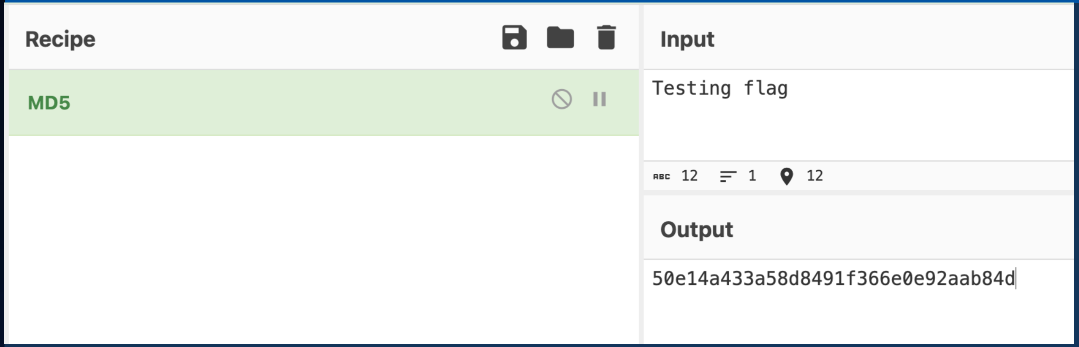
- **Deterministic:** The same input will always produce the same hash value.
- **Fast Computation:** Efficient to compute the hash value for any given data.
- **Irreversible:** It should be infeasible to generate the original input from the hash value.
- **Collision-Resistant:** It should be difficult to find two different inputs that produce the same hash value.

Common Hashing

- **MD5:** Widely used but considered broken and unsuitable for further use.
- **SHA-1:** Previously used for SSL certificates but now considered weak.
- **SHA-256:** Part of the SHA-2 family, widely used and considered secure.

Pyramid of Pain

Hash Value



Recipe

MD5

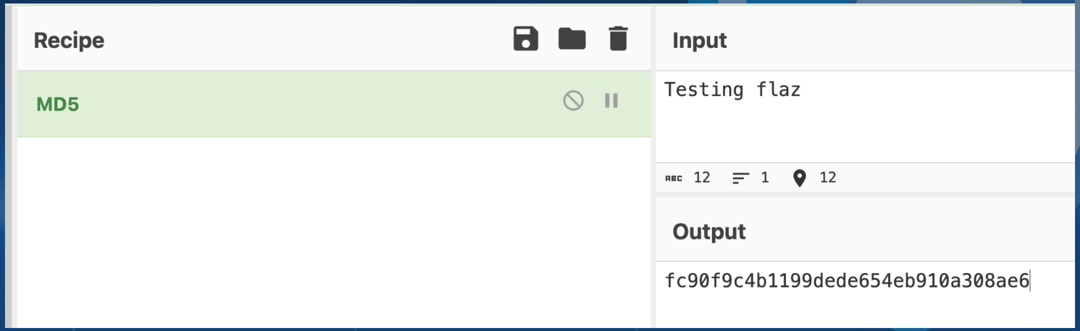
Input

```
Testing flag
```

ABC 12 ☰ 1 📍 12

Output

```
50e14a433a58d8491f366e0e92aab84d
```



Recipe

MD5

Input

```
Testing flaz
```

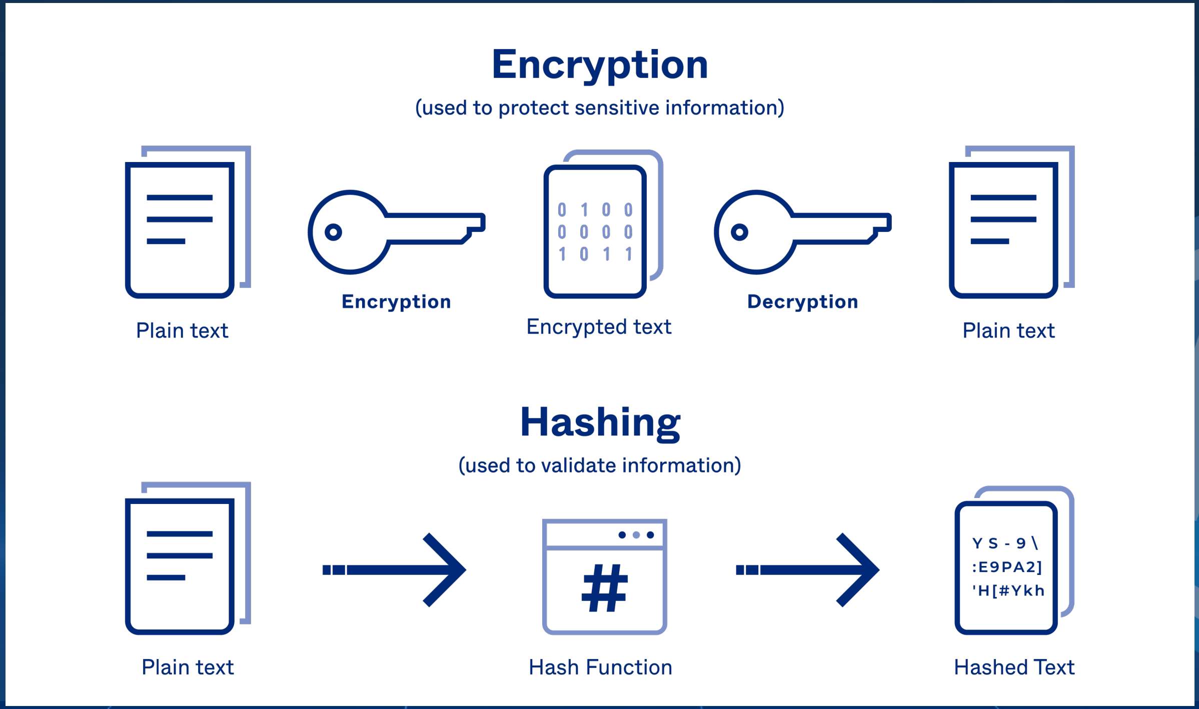
ABC 12 ☰ 1 📍 12

Output

```
fc90f9c4b1199dede654eb910a308ae6
```

Pyramid of Pain

Hash Value



Encryption

(used to protect sensitive information)

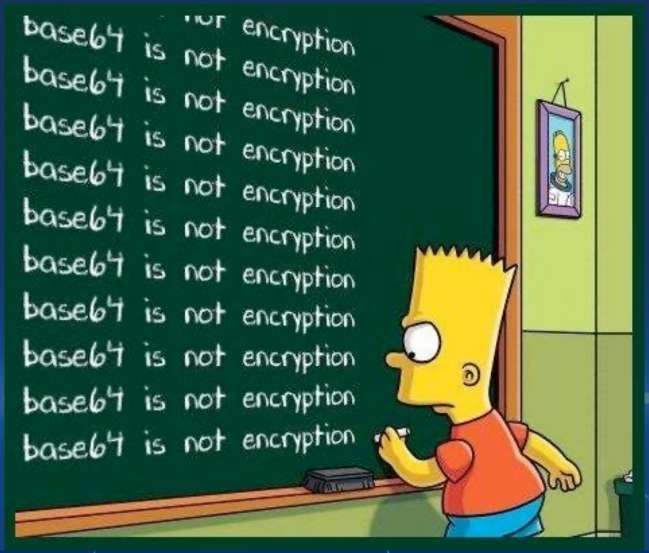
Hashing

(used to validate information)

Pyramid of Pain

Encoding vs Encryption

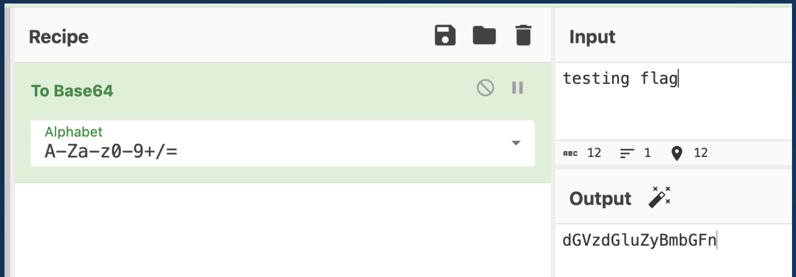
- **Encoding:** A method of converting data into a specific format for efficient transmission or storage. It can be easily reversed using the correct algorithm.
- **Encryption:** A process of converting information into a secret code to prevent unauthorized access. It requires a specific key to decrypt.



Pyramid of Pain

Base64 (Encoding) vs XOR (Encryption)

Base64



Recipe

To Base64

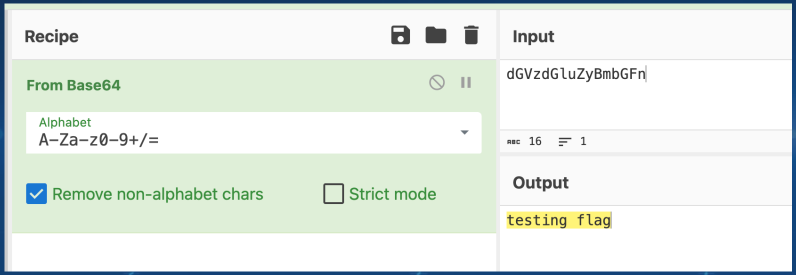
Alphabet: A-Za-z0-9+/=

Input

```
testing flag
```

Output

```
dGVzdGluZyBmbGFn
```



Recipe

From Base64

Alphabet: A-Za-z0-9+/=

☑ Remove non-alphabet chars ☐ Strict mode

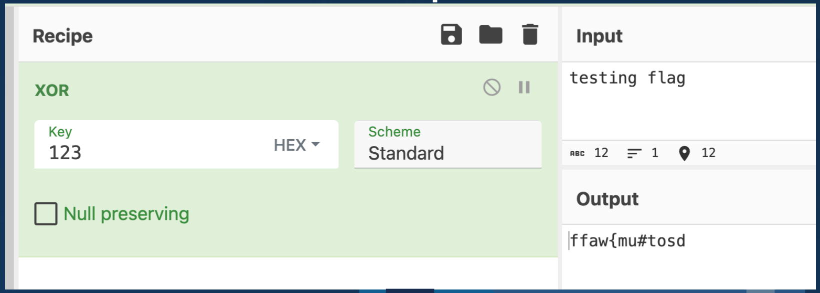
Input

```
dGVzdGluZyBmbGFn
```

Output

```
testing flag
```

XOR Cipher



Recipe

XOR

Key: 123 HEX

Scheme: Standard

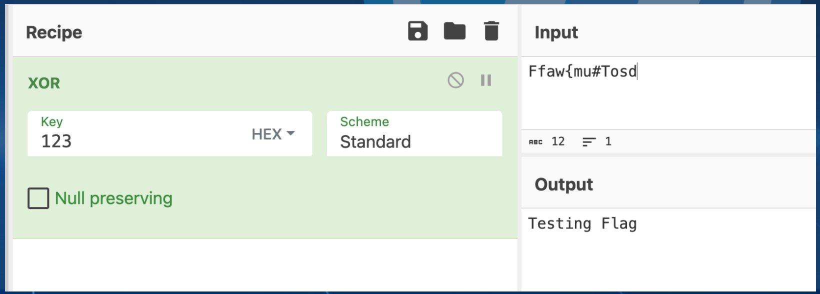
☐ Null preserving

Input

```
testing flag
```

Output

```
ffaw{mu#tosd
```



Recipe

XOR

Key: 123 HEX

Scheme: Standard

☐ Null preserving

Input

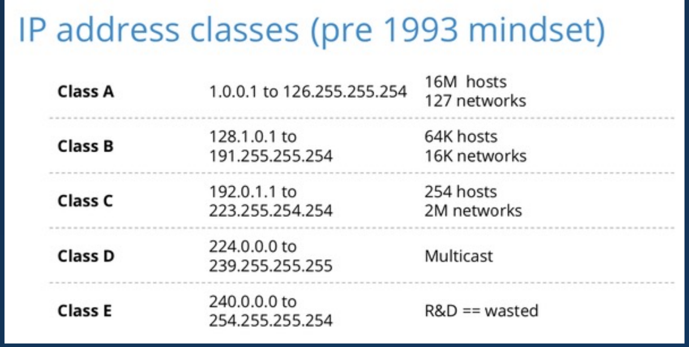
```
Ffaw{mu#Tosd
```

Output

```
Testing Flag
```

Pyramid of Pain

IP Address



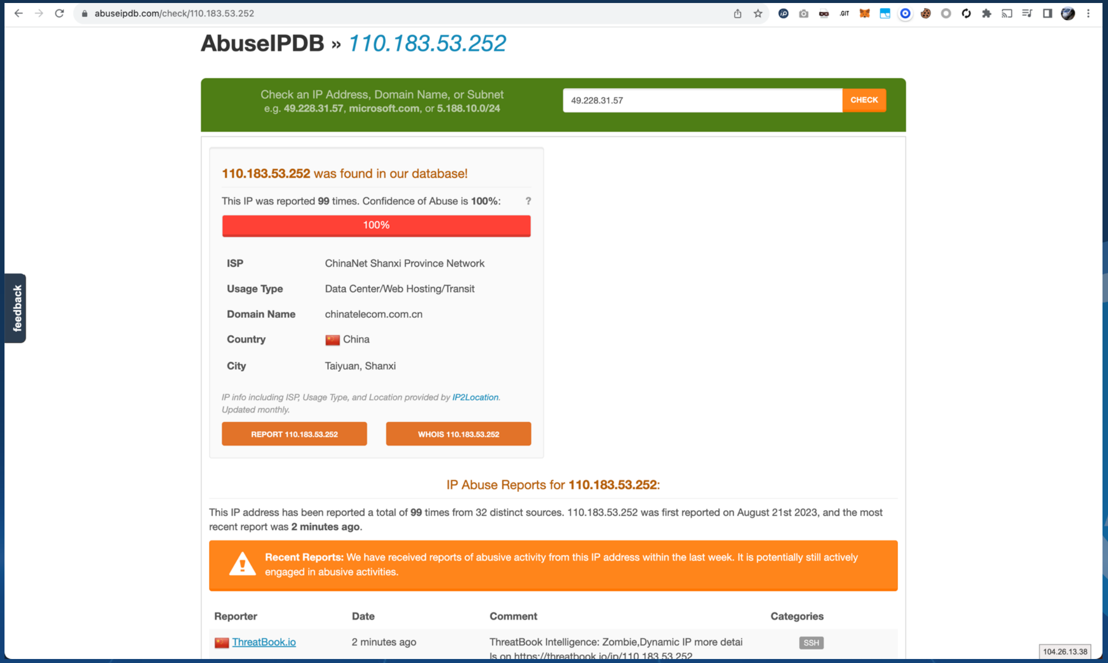
IP address classes (pre 1993 mindset)

Class A	1.0.0.1 to 126.255.255.254	16M hosts 127 networks
Class B	128.1.0.1 to 191.255.255.254	64K hosts 16K networks
Class C	192.0.1.1 to 223.255.254.254	254 hosts 2M networks
Class D	224.0.0.0 to 239.255.255.255	Multicast
Class E	240.0.0.0 to 254.255.255.254	R&D == wasted

- **Description**: Numerical labels assigned to devices connected to a network.
- **Ease for Adversaries**: Relatively easy to change by switching networks or using proxies.
- **Pain for Adversaries**: Low to moderate. Changing IPs requires some effort but is doable.
- **Use Case**: Useful for detecting known malicious sources but can lead to false positives.

Pyramid of Pain

IP Address



abuseipdb.com/check/110.183.53.252

AbuseIPDB » *110.183.53.252*

Check an IP Address, Domain Name, or Subnet
e.g. **49.228.31.57**, **microsoft.com**, or **5.188.10.0/24**

49.228.31.57 CHECK

110.183.53.252 was found in our database!

This IP was reported **99** times. Confidence of Abuse is **100%**: ?

100%

ISP	ChinaNet Shanxi Province Network
Usage Type	Data Center/Web Hosting/Transit
Domain Name	chinatelecom.com.cn
Country	🇨🇳 China
City	Taiyuan, Shanxi

IP info including ISP, Usage Type, and Location provided by IP2Location. Updated monthly.

REPORT 110.183.53.252 WHOIS 110.183.53.252

IP Abuse Reports for **110.183.53.252**:

This IP address has been reported a total of **99** times from 32 distinct sources. 110.183.53.252 was first reported on August 21st 2023, and the most recent report was **2 minutes ago**.

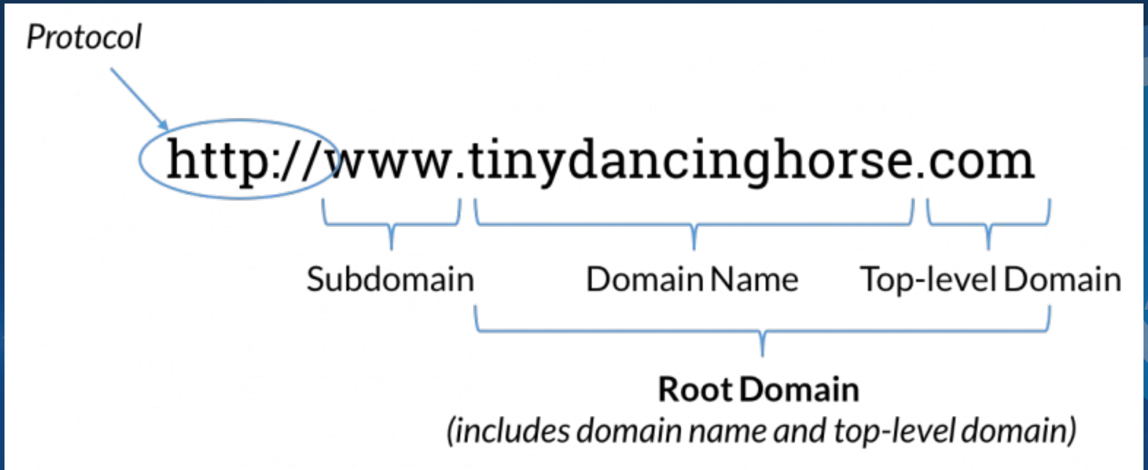
Recent Reports: We have received reports of abusive activity from this IP address within the last week. It is potentially still actively engaged in abusive activities.

Reporter	Date	Comment	Categories
🇨🇳 ThreatBook.io	2 minutes ago	ThreatBook Intelligence: Zombie,Dynamic IP more details on https://threatbook.io/ip/110.183.53.252	SSH

Pyramid of Pain

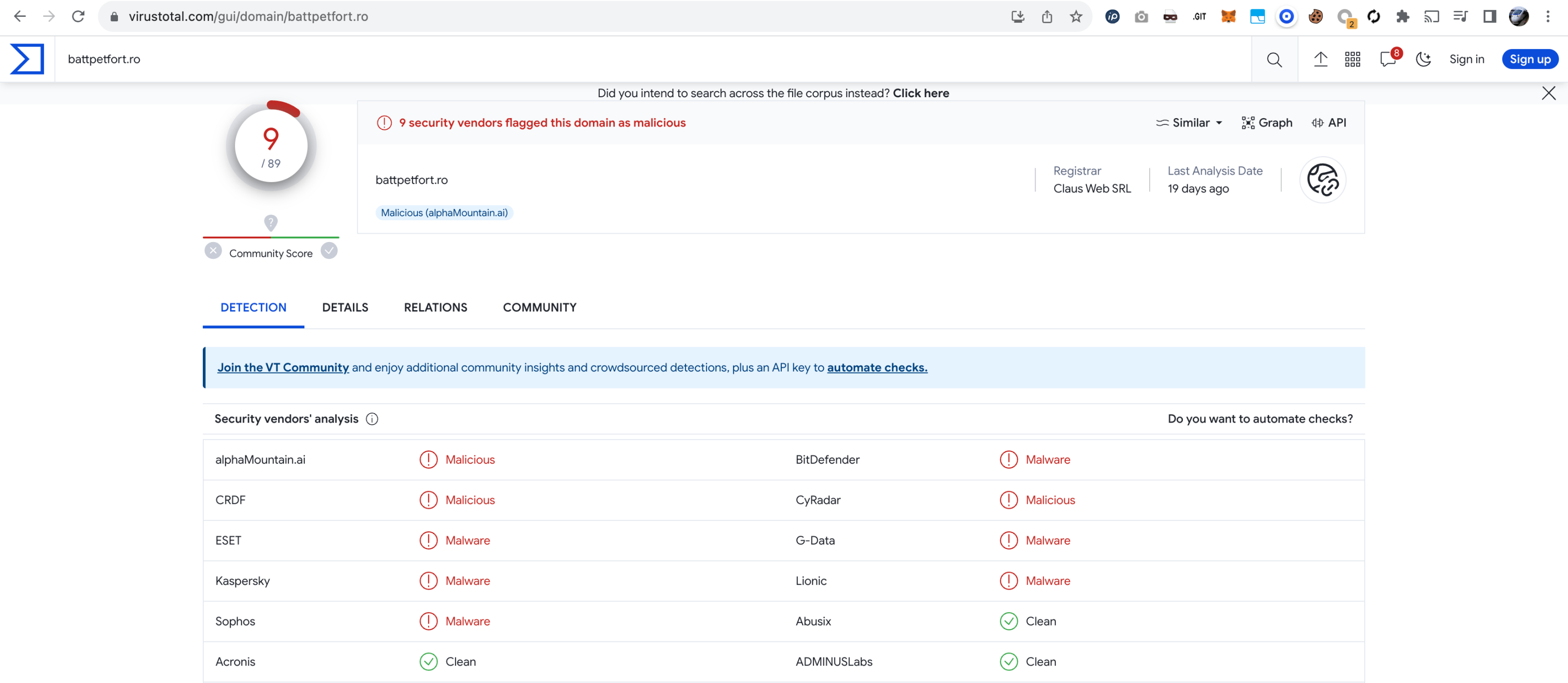
Domain Name

- **Description**: Human-readable names for IP addresses, like example.com.
- **Ease for Adversaries**: Moderate to change, involves registering new domains.
- **Pain for Adversaries**: Moderate. Requires time and possibly money to set up new domains.
- **Use Case**: Good for tracking command and control servers.


Protocol
http://www.tinydancinghorse.com
Subdomain
Domain Name
Top-level Domain
Root Domain
(includes domain name and top-level domain)

Pyramid of Pain

Domain Name



virustotal.com/gui/domain/battpetfort.ro

battpetfort.ro

Did you intend to search across the file corpus instead? **Click here**

9 / 89

Community Score

9 security vendors flagged this domain as malicious

battpetfort.ro

Malicious (alphaMountain.ai)

Registrar: Claus Web SRL

Last Analysis Date: 19 days ago

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

Security vendors' analysis

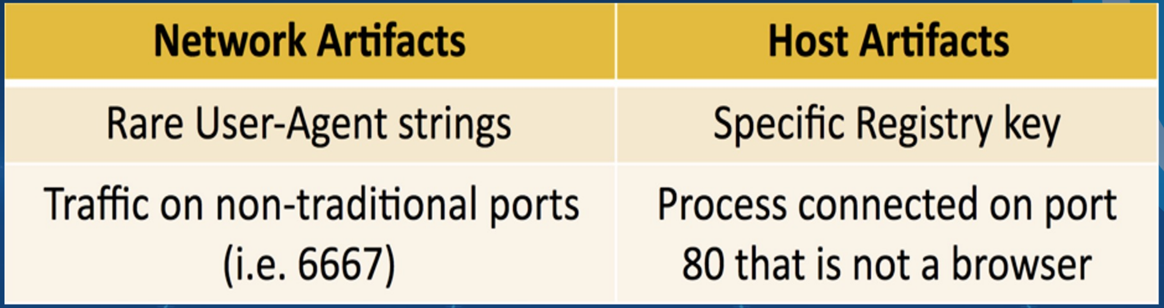
Do you want to automate checks?

alphaMountain.ai	Malicious	BitDefender	Malware
CRDF	Malicious	CyRadar	Malicious
ESET	Malware	G-Data	Malware
Kaspersky	Malware	Lionic	Malware
Sophos	Malware	Abusix	Clean
Acronis	Clean	ADMINUSLabs	Clean

Pyramid of Pain

Network/Host Artefacts

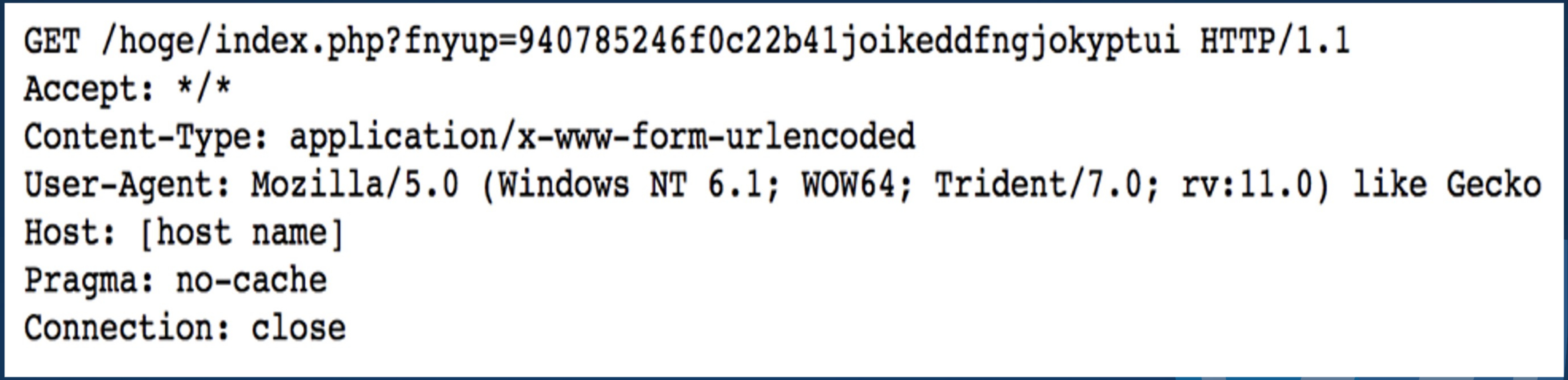
- **Description:** These include registry settings, filenames, and system modifications.
- **Ease for Adversaries:** Moderate to difficult to change without affecting functionality.
- **Pain for Adversaries:** Moderate to high. Requires re-engineering the malware or attack method.
- **Use Case:** Useful for identifying specific attack campaigns or types of malware.



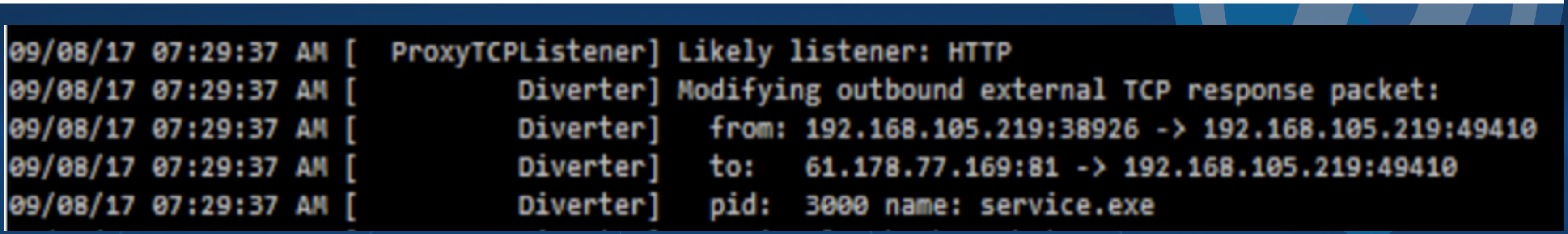
Network Artifacts	Host Artifacts
Rare User-Agent strings	Specific Registry key
Traffic on non-traditional ports (i.e. 6667)	Process connected on port 80 that is not a browser

Pyramid of Pain

Network/Host Artefacts



```
GET /hoge/index.php?fnyup=940785246f0c22b41joikeddfngjokyptui HTTP/1.1
Accept: */*
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; Trident/7.0; rv:11.0) like Gecko
Host: [host name]
Pragma: no-cache
Connection: close
```



```
09/08/17 07:29:37 AM [  ProxyTCPListener] Likely listener: HTTP
09/08/17 07:29:37 AM [          Diverter] Modifying outbound external TCP response packet:
09/08/17 07:29:37 AM [          Diverter]   from: 192.168.105.219:38926 -> 192.168.105.219:49410
09/08/17 07:29:37 AM [          Diverter]   to:   61.178.77.169:81 -> 192.168.105.219:49410
09/08/17 07:29:37 AM [          Diverter]   pid:  3000 name: service.exe
```

Pyramid of Pain

Network/Host Artefacts

- **Run, RunOnce**
  - SOFTWARE\Microsoft\Windows\CurrentVersion\Run
  - SOFTWARE\Microsoft\Windows\CurrentVersion\RunOnce
  - HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
  - HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\RunOnce

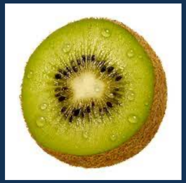
- **Services and Drivers**
  - SYSTEM\CurrentControlSet\Services
    - Services Type shall be "0x10", "0x20", "0x100";
    - Start shall be "2", "3" or "4" only
    - Services without “ObjectName” that is set to: LocalSystem, NT AUTHORITY\LocalService, or NT AUTHORITY\NetworkService
    - Services starting under the Svchost process must have an entry in SOFTWARE\Microsoft\Windows NT\CurrentVersion\svchost

- **Scheduled Tasks**
  - SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Shared Task Scheduler
  - SOFTWARE\Classes\CLSID\{GUID}

- **Browser Helper Objects**
  - SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Browser Helper Objects

Pyramid of Pain

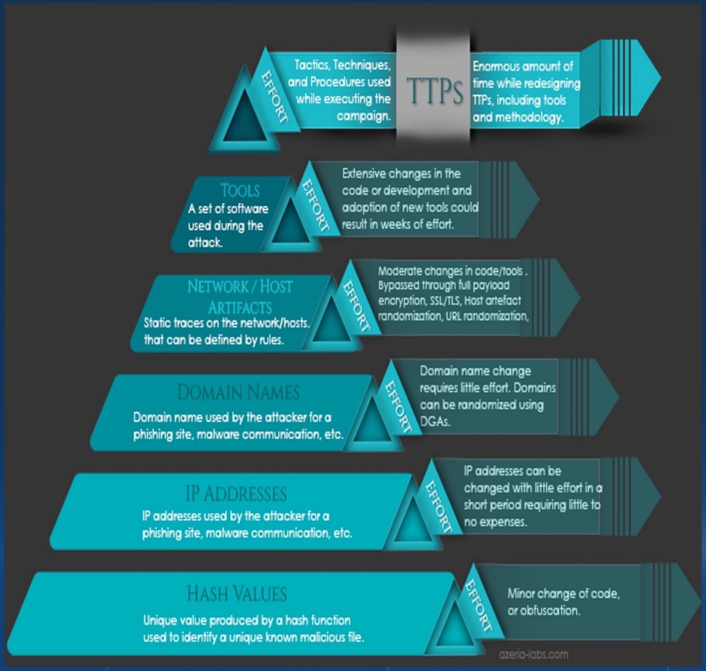
Tools



- **Description:** The software or utilities used by the attacker, like specific malware or exploit kits.
- **Ease for Adversaries:** Difficult to change without losing functionality.
- **Pain for Adversaries:** High. Requires developing or acquiring new tools.
- **Use Case:** Excellent for attributing attacks to specific groups or campaigns.

Pyramid of Pain

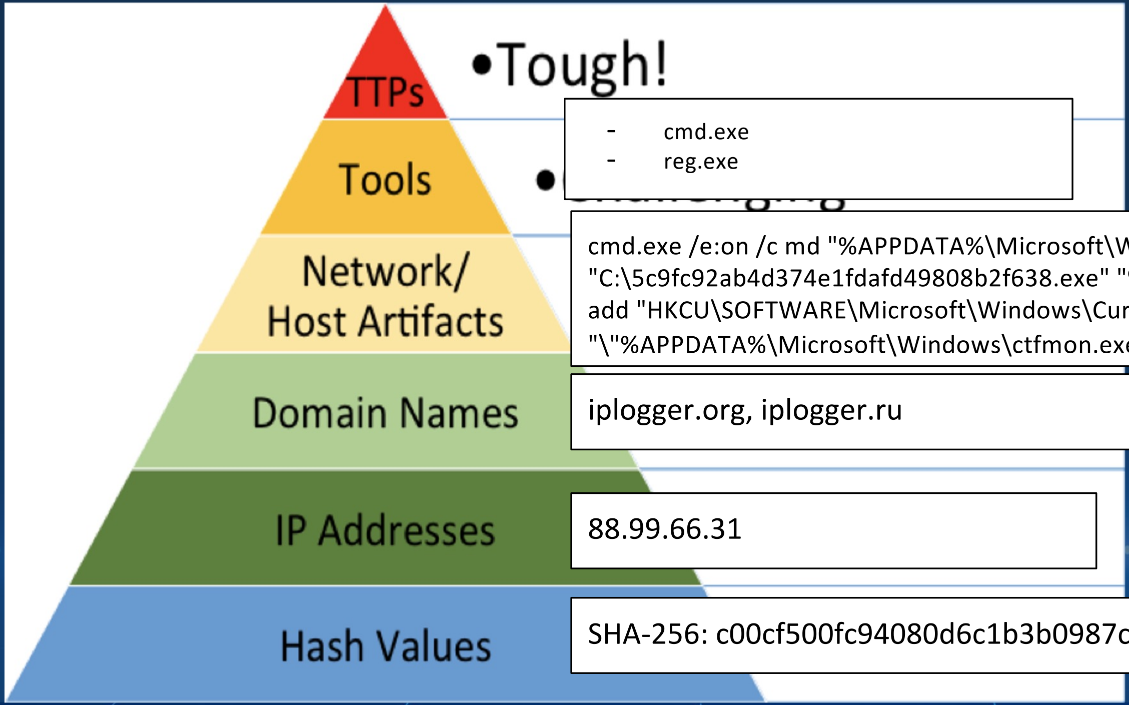
Tools, Techniques, and Procedures. (TTP)


Tactics, Techniques, and Procedures used while executing the campaign.
TTPs
Enormous amount of time while redesigning TTPs, including tools and methodology.
Tools
A set of software used during the attack.
Extensive changes in the code or development and adoption of new tools could result in weeks of effort.
Network / Host Artifacts
Static traces on the network/hosts. that can be defined by rules.
Moderate changes in code/tools . Bypassed through full payload encryption, SSL/TLS, Host artefact randomization, URL randomization,
Domain Names
Domain name used by the attacker for a phishing site, malware communication, etc.
Domain name change requires little effort. Domains can be randomized using DGAs.
IP Addresses
IP addresses used by the attacker for a phishing site, malware communication, etc.
IP addresses can be changed with little effort in a short period requiring little to no expenses.
Hash Values
Unique value produced by a hash function used to identify a unique known malicious file.
Minor change of code, or obfuscation.
Effort
azeria-labs.com

- **Description**: The "how" of the attack, including the strategies and methods used by the attacker.
- **Ease for Adversaries**: Very difficult to change without compromising effectiveness.
- **Pain for Adversaries**: Very high. Requires a fundamental change in approach.
- **Use Case**: Best for long-term defense and attribution but requires deep analysis and expertise.

Pyramid of Pain

Analysis: Closing Cost.xls



- cmd.exe
- reg.exe

```
cmd.exe /e:on /c md "%APPDATA%\Microsoft\Windows" & copy
"C:\5c9fc92ab4d374e1fdafd49808b2f638.exe" "%APPDATA%\Microsoft\Windows\ctfmon.exe" & reg
add "HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run" /V "ctfmon.exe" /t REG_SZ /F /D
"\"%APPDATA%\Microsoft\Windows\ctfmon.exe\" *"
```

iplogger.org, iplogger.ru

88.99.66.31

SHA-256: c00cf500fc94080d6c1b3b0987cc9bc3285621a3f54129d29aa0e29d2dec025d

Yara Rule

Yara Rule

What is Yara Rule

YARA rules serve as signatures for identifying files that match specific conditions, such as MD5, strings, Hex Strings, or file size. These rules can be implemented across various enterprise-level tools, including FireEye, CarbonBlack, Checkpoint, and even in antivirus software like ClamAV.

YARA rules are platform-independent and can be used across different operating systems like Windows, Linux, and MacOS without relying on any engine other than YARA binary. The text also emphasizes the current relevance of YARA rules, especially in the context of encrypted internet traffic, which makes network-level file transfer analysis challenging. This situation leads to a greater reliance on endpoint protection, such as antivirus or APT solutions.

YARA rules can be a vital part of this protection, allowing Security Operation Center (SoC) engineers to analyze malicious files by examining their characteristics and writing YARA rules for prevention. This proactive approach enables organizations to defend against threats even before vendors create specific signatures for those files.

Example

```
# yara MyRule.yar /path/to/scan
```

Yara Rule

How to write Yara Rule

Step-by-step guide to writing a YARA rule:

1. Define the Rule Name

Start by defining a name for the rule. This name should be descriptive and related to what the rule is intended to detect.

Example

```
rule MyExampleRule
```

2. Add Meta Information

You can include meta-information to describe the rule, such as the author, date, or a description of what the rule detects.

```
{
  meta:
    author = "Your Name"
    description = "Detects Example Malware"
    date = "2023-08-22"
```

Yara Rule

How to write Yara Rule

3. Define Strings to Match
In this section, you define the strings or patterns that you want to search for within the files. You can use plain text strings, hexadecimal bytes, or regular expressions.

```
strings:
   $string1 = "malicious code" nocase
   $hex_string = { E2 34 A1 FB }
   $regex = /malware[0-9]+/ nocase
```

- nocase: Makes the string match case-insensitive.
- Hexadecimal patterns are useful for matching binary data.
- Regular expressions provide flexibility in matching complex patterns.

Yara Rule

How to write Yara Rule

4. Define the Condition
The condition is the logical expression that determines if the rule is a match. You can use logical operators like and, or, and not, and refer to the strings defined earlier.

```
 condition:
    $string1 or $hex_string or $regex
}
```

5. Save the Rule
Save the rule with a .yar or .yara extension, such as MyExampleRule.yar.

6. Test the Rule
You can test the rule using the YARA command-line tool by running:

```
# yara MyExampleRule.yar /path/to/files
```

Yara Rule

How to write Yara Rule

```
rule MyExampleRule
{
  meta:
    author = "Your Name"
    description = "Detects Example Malware"
    date = "2023-08-22"
  strings:
    $string1 = "malicious code" nocase
    $hex_string = { E2 34 A1 FB }
    $regex = /malware[0-9]+/ nocase
  condition:
    $string1 or $hex_string or $regex
}
```

Yara Rule

Advanced Yara Rule

Advanced YARA rules allow for more complex and nuanced pattern matching, providing greater flexibility and precision in detecting threats. Here's a detailed look at some advanced features of YARA rules, along with examples:

1) Using Wildcards and Jumps

Wildcards (?) and jumps ([]) allow for more flexible matching of byte sequences.

- **Wildcards**: Match any single byte.
- **Jumps**: Match a range of bytes.

```
strings:
  $pattern1 = { 4D 5A ?? ?? 50 45 } // Wildcard, matches any two bytes between 4D 5A and 50 45
  $pattern2 = { 4D 5A [2-4] 50 45 } // Jump, matches 2 to 4 bytes between 4D 5A and 50 45
```

Yara Rule

Advanced Yara Rule

2. Using External Variables
External variables allow you to define conditions outside of the YARA rule itself, such as file size or file type.

```
rule Detect_PDF
{
  condition:
    filesize < 1MB and
    filetype == "pdf"
}
```

Yara Rule

Advanced Yara Rule

3. Using Functions and Modules

YARA supports various functions and modules that provide additional capabilities.

- **Math Functions**: Such as uint8(), uint16(), etc., to read integer values.
- **PE Module**: To analyze Portable Executable (PE) files.

```
import "pe"

rule Detect_Signed_PE
{
  condition:
    pe.number_of_signatures > 0
}
```

Yara Rule

Advanced Yara Rule

4. Combining Rules

You can create rules that depend on other rules, allowing for more complex logic.

```
rule Is_Executable
{
  condition:
    uint16(0) == 0x5A4D
}

rule Detect_Malware
{
  condition:
    Is_Executable and
    $malicious_string
}
```

Yara Rule

Advanced Yara Rule

4. Combining Rules

You can create rules that depend on other rules, allowing for more complex logic.

```
rule Is_Executable
{
  condition:
    uint16(0) == 0x5A4D
}

rule Detect_Malware
{
  condition:
    Is_Executable and
    $malicious_string
}
```

Yara Rule

Advanced Yara Rule

5. Using Iterators
Iterators allow you to loop through multiple occurrences of a pattern, providing powerful matching capabilities.

```
strings:
  $repeated_string = "malware" nocase

condition:
  # At least three occurrences of the string
  3 of them
```

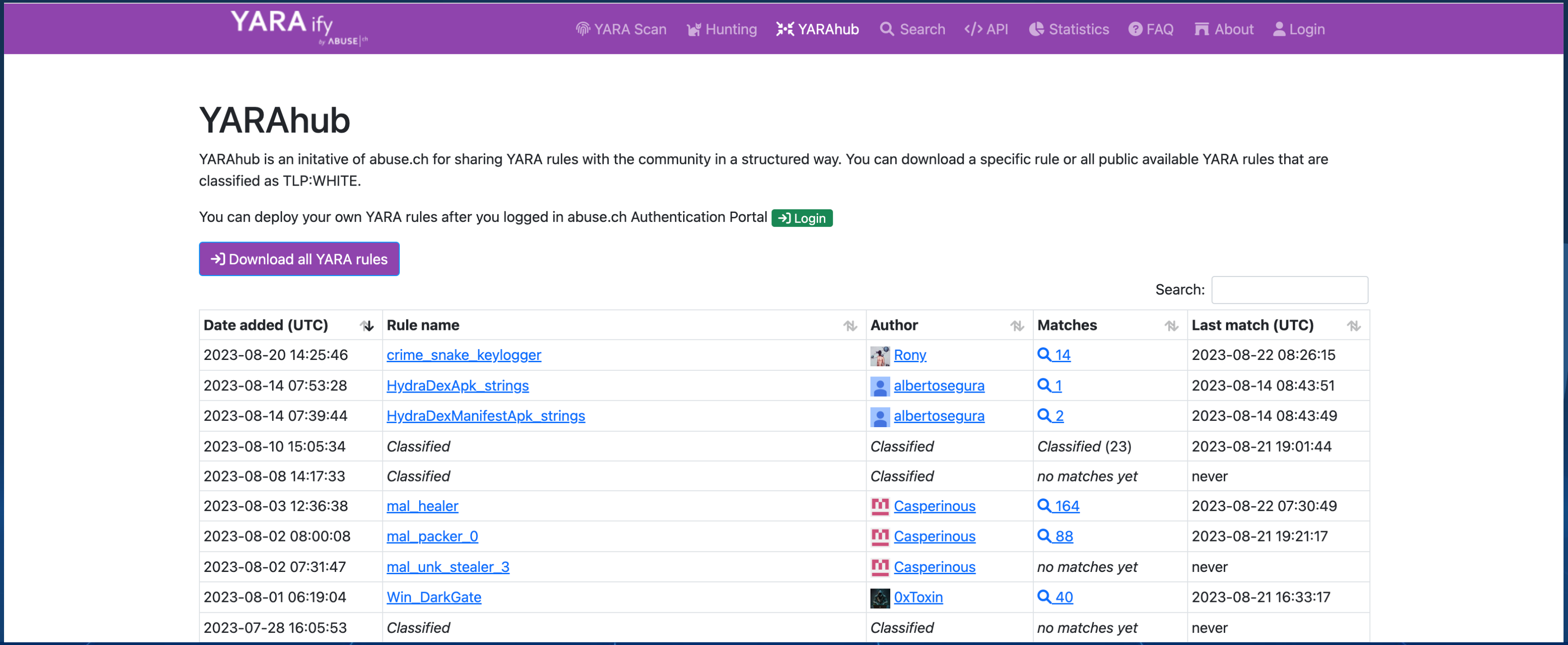
Yara Rule

Advanced Yara Rule

```
import "pe"

rule Detect_Advanced_Malware
{
  meta:
    description = "Detects a specific advanced malware"

  strings:
    $signature = { 4D 5A [2-4] 50 45 }

  condition:
    pe.is_dll() and
    filesize < 1MB and
    $signature and
    2 of them
}
```

Yara Rule

YARAify



YARAhub

YARAhub is an initative of abuse.ch for sharing YARA rules with the community in a structured way. You can download a specific rule or all public available YARA rules that are classified as TLP:WHITE.

You can deploy your own YARA rules after you logged in abuse.ch Authentication Portal Login

Download all YARA rules

Search:

Date added (UTC)	Rule name	Author	Matches	Last match (UTC)
2023-08-20 14:25:46	crime_snake_keylogger	Rony	14	2023-08-22 08:26:15
2023-08-14 07:53:28	HydraDexApk_strings	albertosegura	1	2023-08-14 08:43:51
2023-08-14 07:39:44	HydraDexManifestApk_strings	albertosegura	2	2023-08-14 08:43:49
2023-08-10 15:05:34	*Classified*	*Classified*	*Classified* (23)	2023-08-21 19:01:44
2023-08-08 14:17:33	*Classified*	*Classified*	*no matches yet*	never
2023-08-03 12:36:38	mal_healer	Casperinous	164	2023-08-22 07:30:49
2023-08-02 08:00:08	mal_packer_0	Casperinous	88	2023-08-21 19:21:17
2023-08-02 07:31:47	mal_unk_stealer_3	Casperinous	*no matches yet*	never
2023-08-01 06:19:04	Win_DarkGate	0xToxin	40	2023-08-21 16:33:17
2023-07-28 16:05:53	*Classified*	*Classified*	*no matches yet*	never

Yara Rule

Demo

https://play.secplayground.com/lab/568

Sigma Rule

Sigma Rule

What is Sigma Rule

Sigma is a generic and open signature format that allows you to describe relevant log events in a straightforward manner. The rule format is very flexible, easy to write, and applicable to any type of log file. The main purpose of Sigma is to provide a structured format in which researchers or analysts can describe their methods of detecting particular cybersecurity events. Sigma rules are typically used in Security Information and Event Management (SIEM) systems to define how to identify specific activities that are of interest for security monitoring.

Sigma Rule

Sigma Rule Structure

- **title:** A brief description of the rule.
- **id:** A unique identifier for the rule.
- **description:** A detailed explanation of what the rule does.
- **logsource:** Specifies the source of the log data.
- **detection:** Defines the conditions for the rule.
- **condition:** Specifies the logic to apply the conditions.
- **falsepositives:** Describes the potential false positives.
- **level:** Sets the severity level of the rule.

Sigma Rule

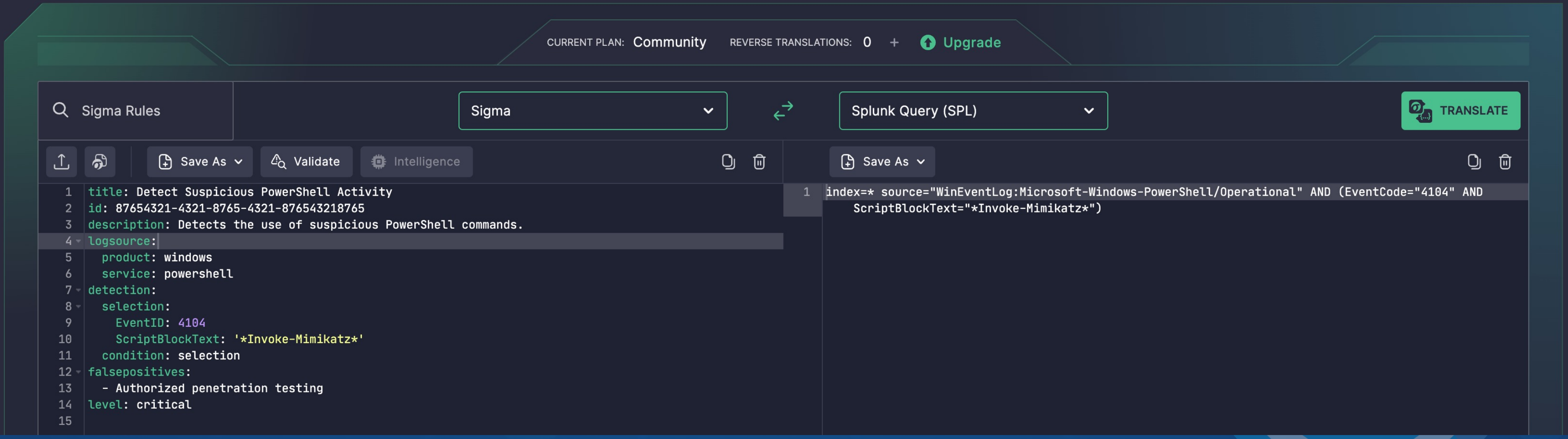
Sample of Sigma Rule

```
title: Detect Suspicious PowerShell Activity
id: 87654321-4321-8765-4321-876543218765
description: Detects the use of suspicious PowerShell commands.
logsource:
  product: windows
  service: powershell
detection:
  selection:
    EventID: 4104
    ScriptBlockText: '*Invoke-Mimikatz*'
  condition: selection
falsepositives:
  - Authorized penetration testing
level: critical
```

Sigma Rule

Sample of Sigma Rule

https://tdm.socprime.com/uncoder-ai



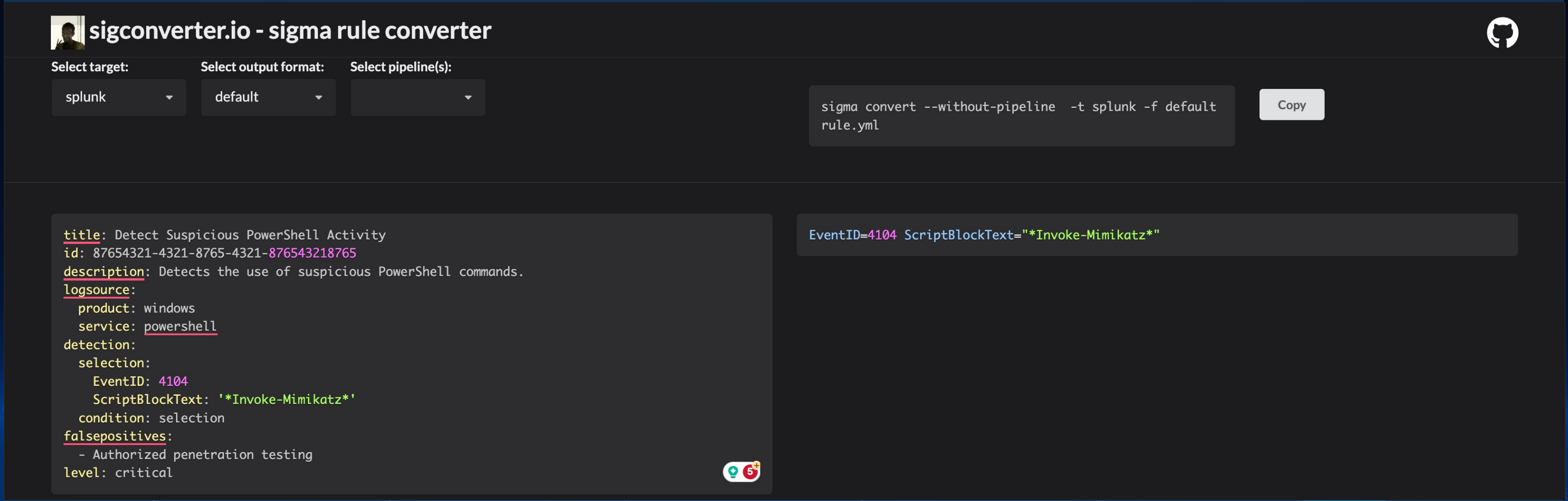
```yaml
title: Detect Suspicious PowerShell Activity
id: 87654321-4321-8765-4321-876543218765
description: Detects the use of suspicious PowerShell commands.
logsource:
  product: windows
  service: powershell
detection:
  selection:
    EventID: 4104
    ScriptBlockText: '*Invoke-Mimikatz*'
  condition: selection
falsepositives:
  - Authorized penetration testing
level: critical
```

```
index=* source="WinEventLog:Microsoft-Windows-PowerShell/Operational" AND (EventCode="4104" AND
    ScriptBlockText="*Invoke-Mimikatz*")
```

Sigma Rule

Sample of Sigma Rule

https://sigconverter.io/



sigconverter.io - sigma rule converter

Select target: splunk

Select output format: default

Select pipeline(s):

```
sigma convert --without-pipeline  -t splunk -f default
rule.yml
```

```
title: Detect Suspicious PowerShell Activity
id: 87654321-4321-8765-4321-876543218765
description: Detects the use of suspicious PowerShell commands.
logsource:
  product: windows
  service: powershell
detection:
  selection:
    EventID: 4104
    ScriptBlockText: '*Invoke-Mimikatz*'
  condition: selection
falsepositives:
  - Authorized penetration testing
level: critical
```

```
EventID=4104 ScriptBlockText="*Invoke-Mimikatz*"
```

Challenge of Threat Hunting

Challenge of Threat Hunting

Challenge

Budget

- Accurate planning
- Proving Return On Investment (ROI)
- Meaningful and Accurate Metrics

Capability

- Identifying and prioritizing gaps in capability
- Measuring current performance
  - Tools
  - Staff

Challenge of Threat Hunting

Output of Threat Hunting

Nothing Found

Benefits

- No malicious activity is present.
- Or current visibility in the organization is not enough or the tools that used by threat hunters is not good enough to help them to investigate the case.

Something Found Non-Malicious

Benefits

- Identify compliance/best practice issue
- Validate effectiveness of hunting activity

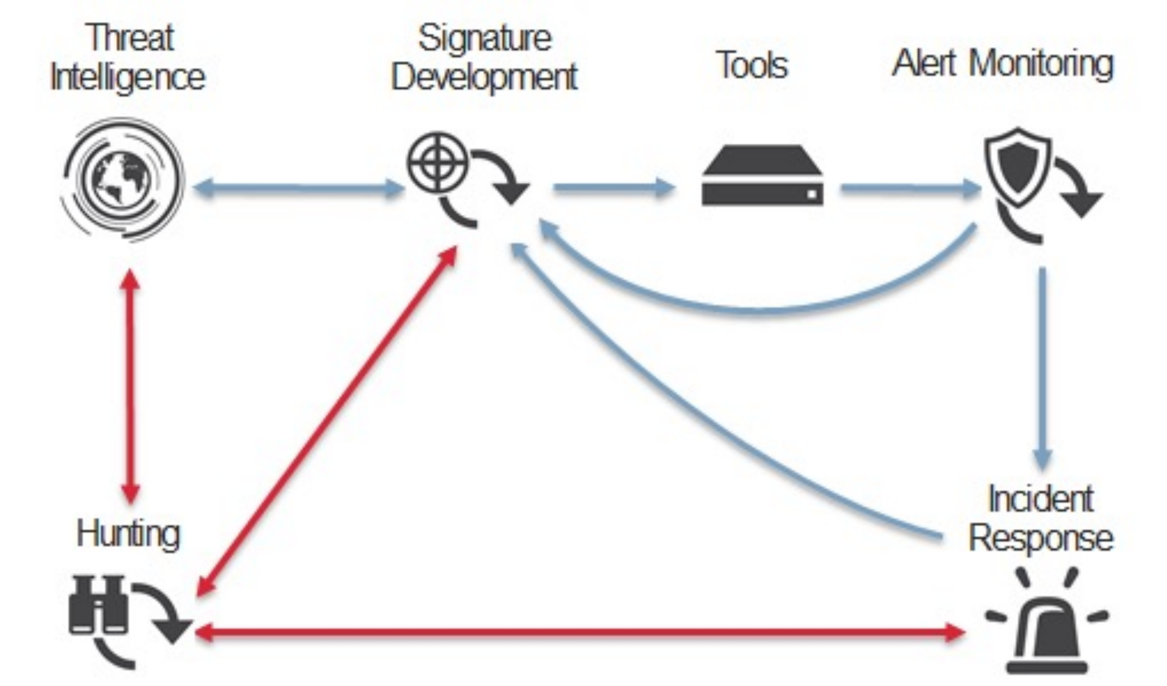
Something Found Malicious

Benefits

- Identify security incident
- Validate effectiveness of hunting activity

Challenge of Threat Hunting

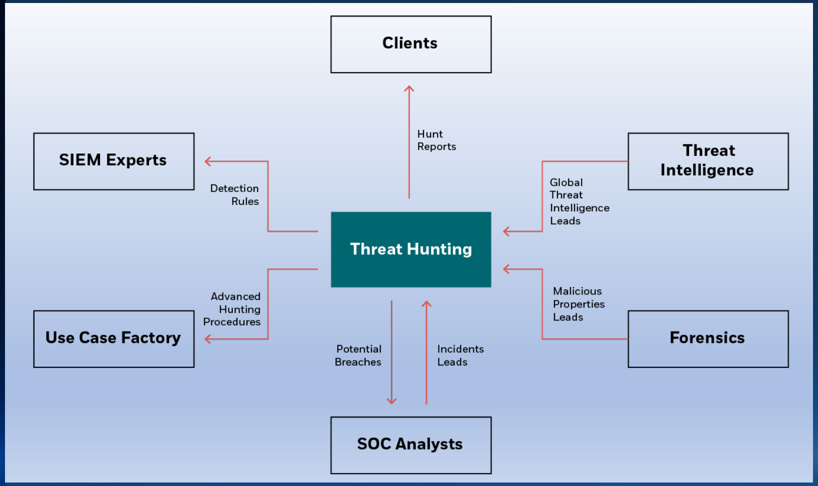
Integration



- Strategic hunting plan clearly defines roles and responsibilities
- Provides additional information flow conduits
- Documenting the Hunting Program plan is critical to success
- Metrics review creates near real time feedback loop

Challenge of Threat Hunting

Output of Threat Hunting



- New rules of SIEM for detect suspicious behavior.
- Update current rules to more effective.
- White paper.
- Signatures

Challenge of Threat Hunting

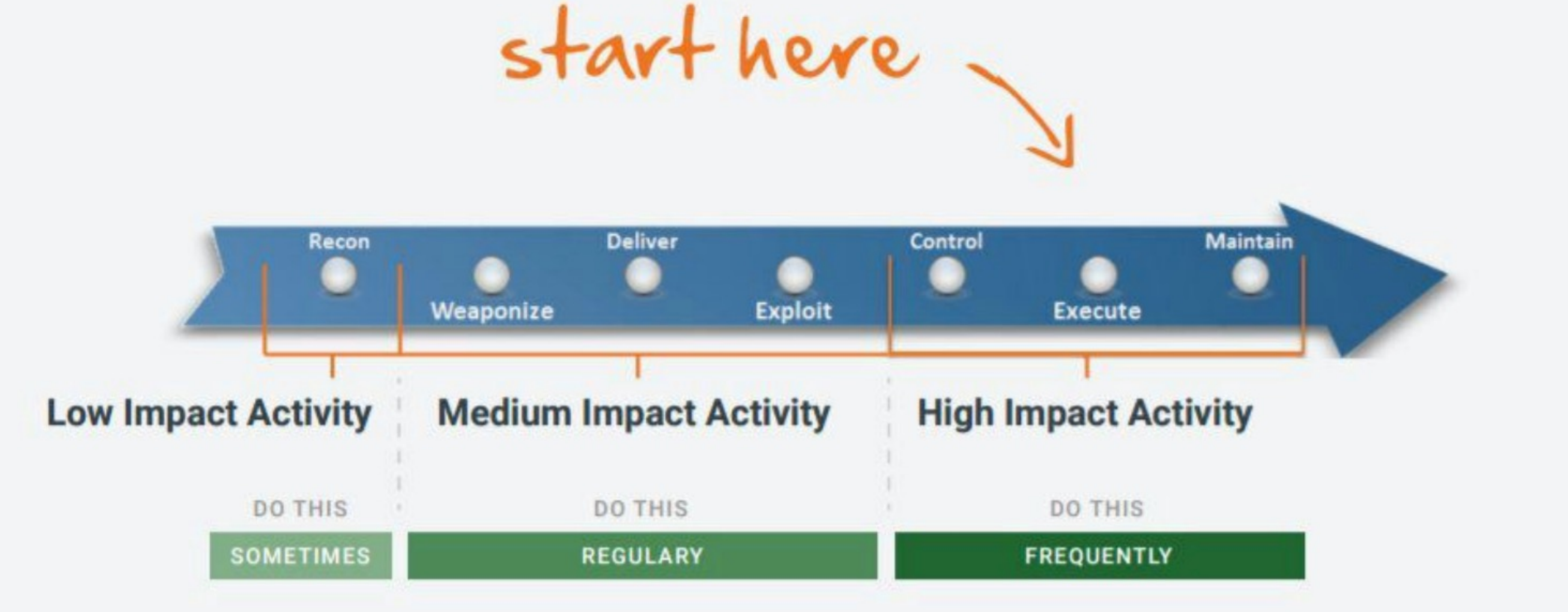
Threat Hunting Benefit

- Finding adversaries who have gotten past your current security protection
- Continuous improvement of your detection capabilities
- With your existing technology, you can not have oversight of everything that’s happening, at this point threat hunting help your organization
- Supports faster and early detection of potential compromise
- Increasing awareness of your environment and attack surface
- One of method to improve your data collection

Metrics and Visualization

Threat Hunting

Attack Kill Chain



Threat Hunting

Hunting Metrics Matrix

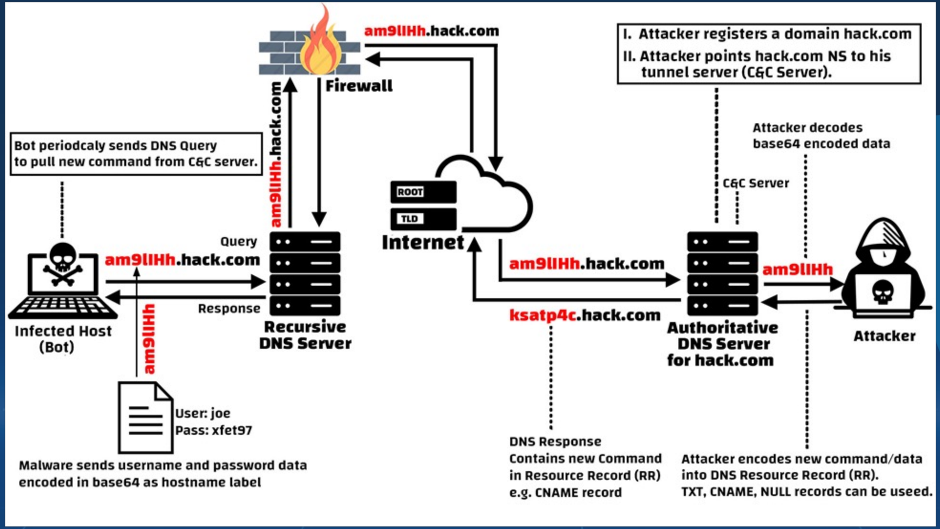
Attacker Lifecycle Phase	Hunting Goal	Hunting Activity	Priority	Tools	Data Source
Init. Compromise, Est Foothold / Escalate Priv	Detect host based artifacts of installed webshells	Analyze server web logs for evidence of installed webshell	Low	Remote Forensics Tool	DATT agent
Init. Compromise, Est Foothold / Escalate Priv / Internal Recon / Move Laterally / Maintain Presence / Complete Mission	Detect host based artifacts of malicious use of legitimate processes	Analyze hosts for signs of malicious powershell use	Med	Remote Forensics Tool	DATT agent
Escalate Privileges	Detect network based artifacts of attacker tools	Use network sensors to identify dumpers on the wire: keywords from strings, filenames, hashes	Low	Network Logs, Network Alerts	Mcafee IDS, Mcafee IPS, Firewalls, VPN logs, Proxy Logs
Escalate Privileges, Move Laterally	Detect host based artifacts of inappropriate use of privileged accounts	Identify authentication with privileged accounts that have been disabled	Med	SIEM	Domain Logs
Establish Foothold, Escalate Privileges, Move Laterally, Maintain Presence	Detect account compromise	Identify attempted execution of common dumper service names	Med	Remote Forensics Tool	DATT agent
Complete mission	Detect unauthorized movement of data in and out of the environment	Unusual traffic: size, frequency, endpoints, port/protocol	High	SIEM, Network Logs	Flow data, Raw packet data
Complete mission	Detect unauthorized movement of data in and out of the environment	Stacks of common archive formats passing over Network	High	SIEM, Network Logs	IDS, IPS, Raw packet data

Sample of Threat Hunting

Reactive Threat Hunting

DNS Tunneling

DNS tunneling involves abuse of the underlying DNS protocol. Instead of using DNS requests and replies to perform legitimate IP address lookups, malware uses it to implement a command and control channel with its handler.



I. Attacker registers a domain hack.com

II. Attacker points hack.com NS to his tunnel server (C&C Server).

Bot periodcaly sends DNS Query to pull new command from C&C server.

Attacker decodes base64 encoded data

Malware sends username and password data encoded in base64 as hostname label

DNS Response Contains new Command in Resource Record (RR) e.g. CNAME record

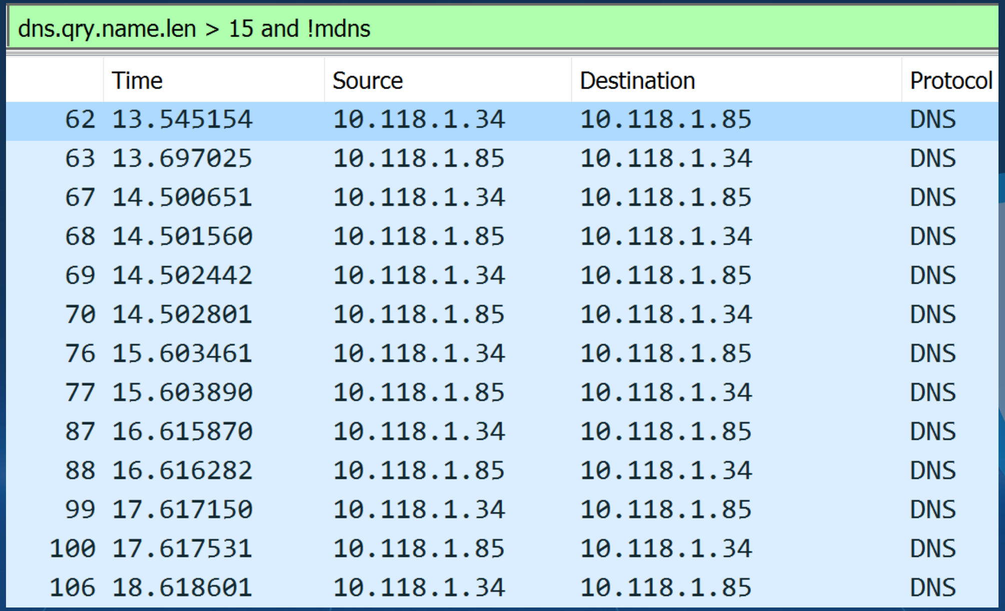
Attacker encodes new command/data into DNS Resource Record (RR). TXT, CNAME, NULL records can be useed.

https://www.socinvestigation.com/how-dns-tunneling-works-detection-response/

Reactive Threat Hunting

DNS Tunneling

Wireshark: dns.qry.name.len > 15 and !mdns



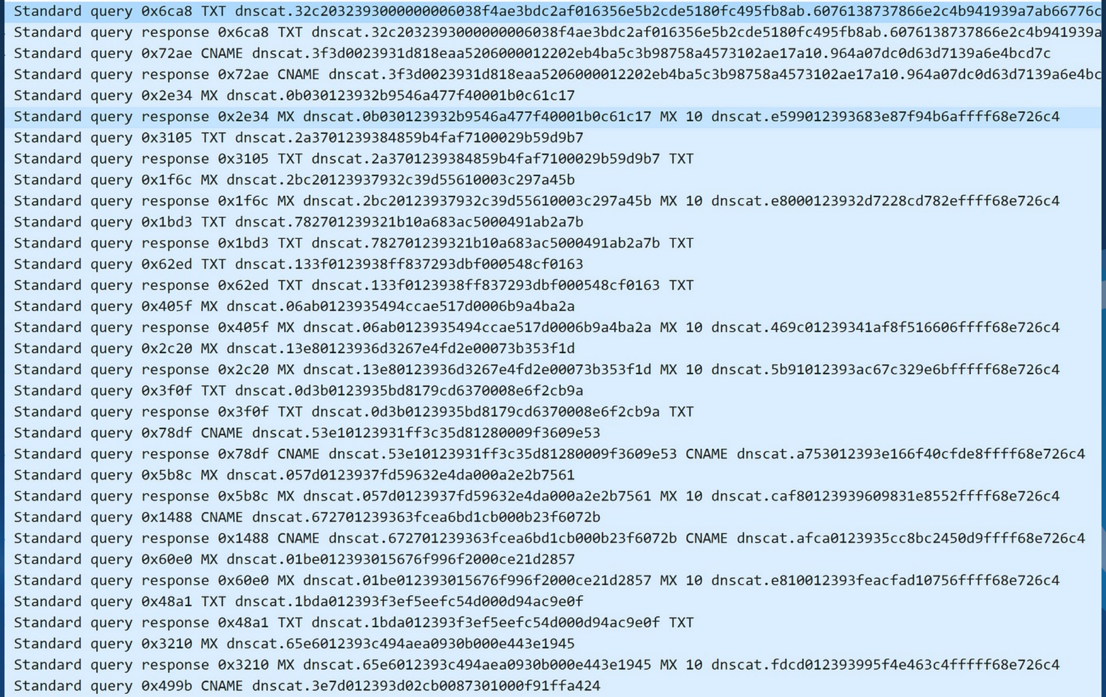
dns.qry.name.len > 15 and !mdns

	Time	Source	Destination	Protocol
62	13.545154	10.118.1.34	10.118.1.85	DNS
63	13.697025	10.118.1.85	10.118.1.34	DNS
67	14.500651	10.118.1.34	10.118.1.85	DNS
68	14.501560	10.118.1.85	10.118.1.34	DNS
69	14.502442	10.118.1.34	10.118.1.85	DNS
70	14.502801	10.118.1.85	10.118.1.34	DNS
76	15.603461	10.118.1.34	10.118.1.85	DNS
77	15.603890	10.118.1.85	10.118.1.34	DNS
87	16.615870	10.118.1.34	10.118.1.85	DNS
88	16.616282	10.118.1.85	10.118.1.34	DNS
99	17.617150	10.118.1.34	10.118.1.85	DNS
100	17.617531	10.118.1.85	10.118.1.34	DNS
106	18.618601	10.118.1.34	10.118.1.85	DNS

https://alparslanakyildiz.medium.com/detecting-dns-tunnelling-with-wireshark-71ce39cd8fe5

Reactive Threat Hunting

DNS Tunneling



```
Standard query 0x6ca8 TXT dnscat.32c2032393000000006038f4ae3bdc2af016356e5b2cde5180fc495fb8ab.6076138737866e2c4b941939a7ab66776c
Standard query response 0x6ca8 TXT dnscat.32c2032393000000006038f4ae3bdc2af016356e5b2cde5180fc495fb8ab.6076138737866e2c4b941939a
Standard query 0x72ae CNAME dnscat.3f3d0023931d818eaa5206000012202eb4ba5c3b98758a4573102ae17a10.964a07dc0d63d7139a6e4bcd7c
Standard query response 0x72ae CNAME dnscat.3f3d0023931d818eaa5206000012202eb4ba5c3b98758a4573102ae17a10.964a07dc0d63d7139a6e4bc
Standard query 0x2e34 MX dnscat.0b030123932b9546a477f40001b0c61c17
Standard query response 0x2e34 MX dnscat.0b030123932b9546a477f40001b0c61c17 MX 10 dnscat.e599012393683e87f94b6affff68e726c4
Standard query 0x3105 TXT dnscat.2a3701239384859b4faf7100029b59d9b7
Standard query response 0x3105 TXT dnscat.2a3701239384859b4faf7100029b59d9b7 TXT
Standard query 0x1f6c MX dnscat.2bc20123937932c39d55610003c297a45b
Standard query response 0x1f6c MX dnscat.2bc20123937932c39d55610003c297a45b MX 10 dnscat.e8000123932d7228cd782effff68e726c4
Standard query 0x1bd3 TXT dnscat.782701239321b10a683ac5000491ab2a7b
Standard query response 0x1bd3 TXT dnscat.782701239321b10a683ac5000491ab2a7b TXT
Standard query 0x62ed TXT dnscat.133f0123938ff837293dbf000548cf0163
Standard query response 0x62ed TXT dnscat.133f0123938ff837293dbf000548cf0163 TXT
Standard query 0x405f MX dnscat.06ab0123935494ccae517d0006b9a4ba2a
Standard query response 0x405f MX dnscat.06ab0123935494ccae517d0006b9a4ba2a MX 10 dnscat.469c01239341af8f516606ffff68e726c4
Standard query 0x2c20 MX dnscat.13e80123936d3267e4fd2e00073b353f1d
Standard query response 0x2c20 MX dnscat.13e80123936d3267e4fd2e00073b353f1d MX 10 dnscat.5b91012393ac67c329e6bfffff68e726c4
Standard query 0x3f0f TXT dnscat.0d3b0123935bd8179cd6370008e6f2cb9a
Standard query response 0x3f0f TXT dnscat.0d3b0123935bd8179cd6370008e6f2cb9a TXT
Standard query 0x78df CNAME dnscat.53e10123931ff3c35d81280009f3609e53
Standard query response 0x78df CNAME dnscat.53e10123931ff3c35d81280009f3609e53 CNAME dnscat.a753012393e166f40cfde8ffff68e726c4
Standard query 0x5b8c MX dnscat.057d0123937fd59632e4da000a2e2b7561
Standard query response 0x5b8c MX dnscat.057d0123937fd59632e4da000a2e2b7561 MX 10 dnscat.caf80123939609831e8552ffff68e726c4
Standard query 0x1488 CNAME dnscat.672701239363fcea6bd1cb000b23f6072b
Standard query response 0x1488 CNAME dnscat.672701239363fcea6bd1cb000b23f6072b CNAME dnscat.afca0123935cc8bc2450d9ffff68e726c4
Standard query 0x60e0 MX dnscat.01be012393015676f996f2000ce21d2857
Standard query response 0x60e0 MX dnscat.01be012393015676f996f2000ce21d2857 MX 10 dnscat.e810012393feacfad10756ffff68e726c4
Standard query 0x48a1 TXT dnscat.1bda012393f3ef5eefc54d000d94ac9e0f
Standard query response 0x48a1 TXT dnscat.1bda012393f3ef5eefc54d000d94ac9e0f TXT
Standard query 0x3210 MX dnscat.65e6012393c494aea0930b000e443e1945
Standard query response 0x3210 MX dnscat.65e6012393c494aea0930b000e443e1945 MX 10 dnscat.fdcd012393995f4e463c4fffff68e726c4
Standard query 0x499b CNAME dnscat.3e7d012393d02cb0087301000f91ffa424
```

https://alparslanakyildiz.medium.com/detecting-dns-tunnelling-with-wireshark-71ce39cd8fe5

Reactive Threat Hunting

DNS Tunneling Lab

https://play.secplayground.com/lab/996

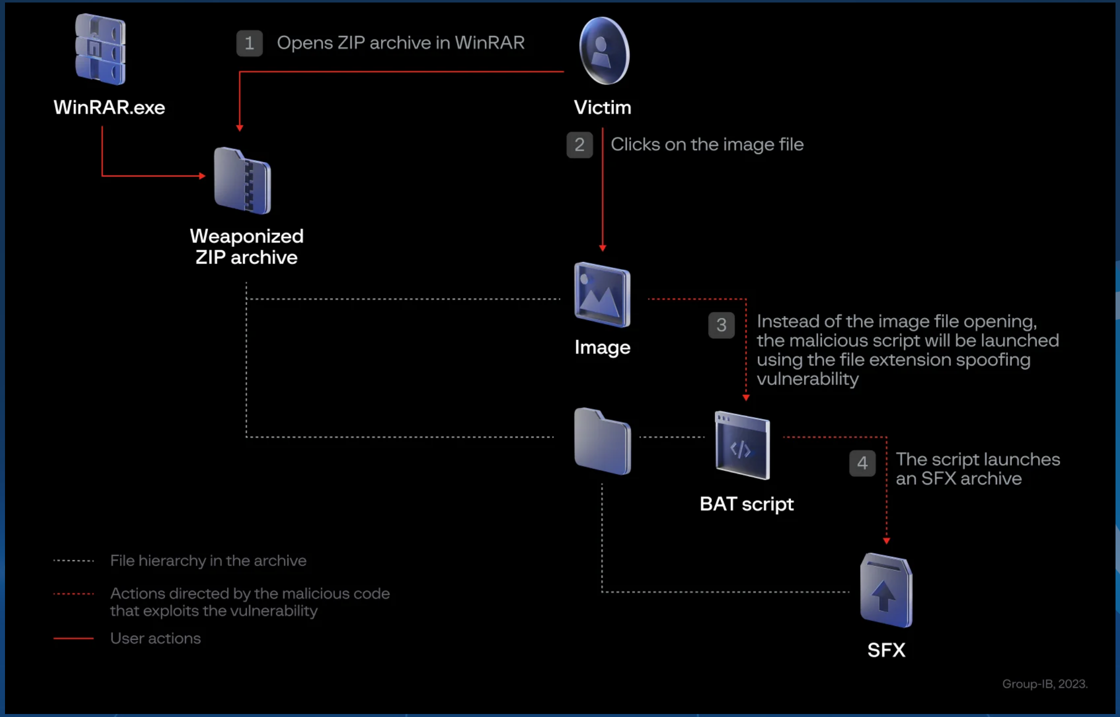
Proactive Threat Hunting

WINRAR CVE-2023-38831

https://www.youtube.com/watch?v=gkwMb1hjmIA&ab_channel=TacticalAdversary

Proactive Threat Hunting

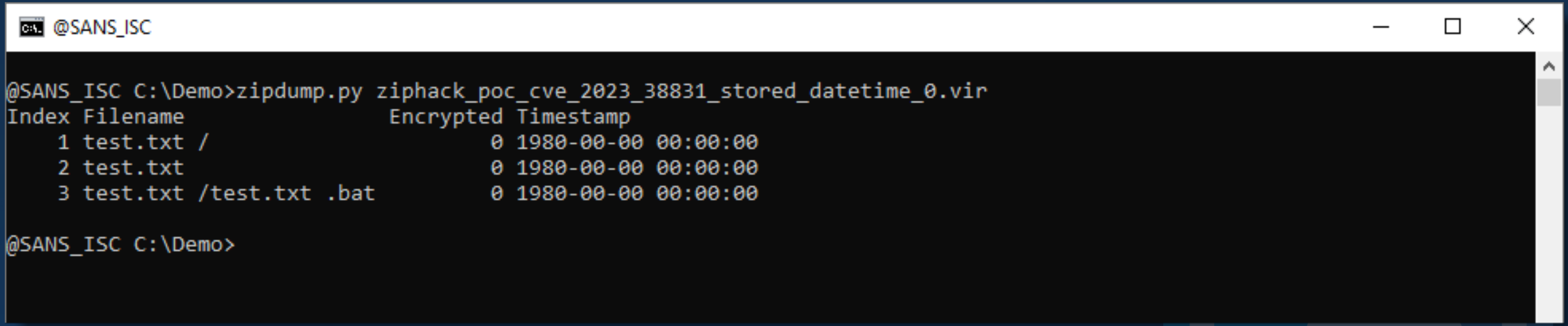
WINRAR CVE-2023-38831



WinRAR.exe
1 Opens ZIP archive in WinRAR
Victim
Weaponized ZIP archive
2 Clicks on the image file
Image
3 Instead of the image file opening, the malicious script will be launched using the file extension spoofing vulnerability
BAT script
4 The script launches an SFX archive
SFX
File hierarchy in the archive
Actions directed by the malicious code that exploits the vulnerability
User actions
Group-IB, 2023.

Proactive Threat Hunting

WINRAR CVE-2023-38831

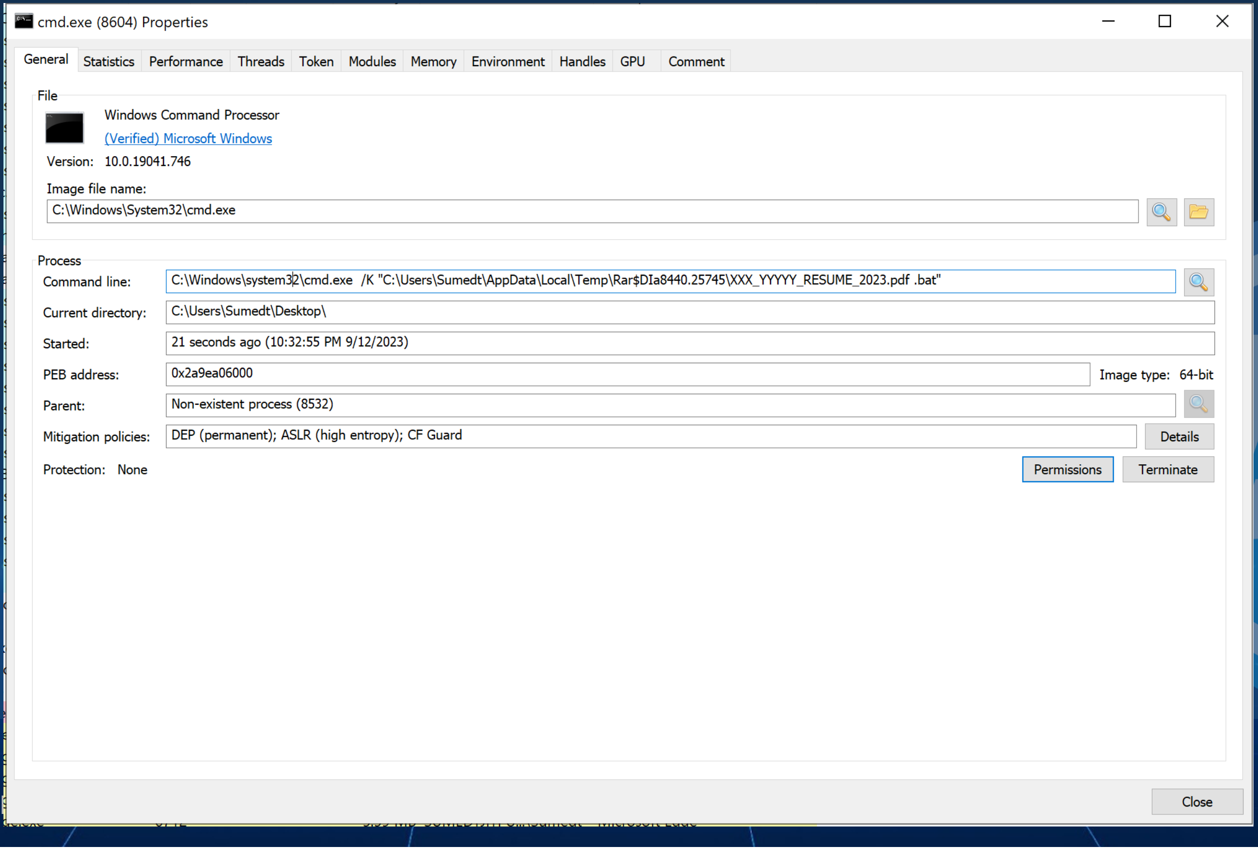


```
@SANS_ISC C:\Demo>zipdump.py ziphack_poc_cve_2023_38831_stored_datetime_0.vir
Index Filename                Encrypted Timestamp
    1 test.txt /                      0 1980-00-00 00:00:00
    2 test.txt                        0 1980-00-00 00:00:00
    3 test.txt /test.txt .bat         0 1980-00-00 00:00:00

@SANS_ISC C:\Demo>
```

https://isc.sans.edu/diary/Analysis+of+RAR+Exploit+Files+CVE202338831/30164/

Proactive Threat Hunting

WINRAR CVE-2023-38831



File

Windows Command Processor

(Verified) Microsoft Windows

Version: 10.0.19041.746

Image file name:
C:\Windows\System32\cmd.exe

Process

Command line: C:\Windows\system32\cmd.exe /K "C:\Users\Sumedt\AppData\Local\Temp\Rar$DIa8440.25745\XXX_YYYYY_RESUME_2023.pdf .bat"

Current directory: C:\Users\Sumedt\Desktop\

Started: 21 seconds ago (10:32:55 PM 9/12/2023)

PEB address: 0x2a9ea06000 Image type: 64-bit

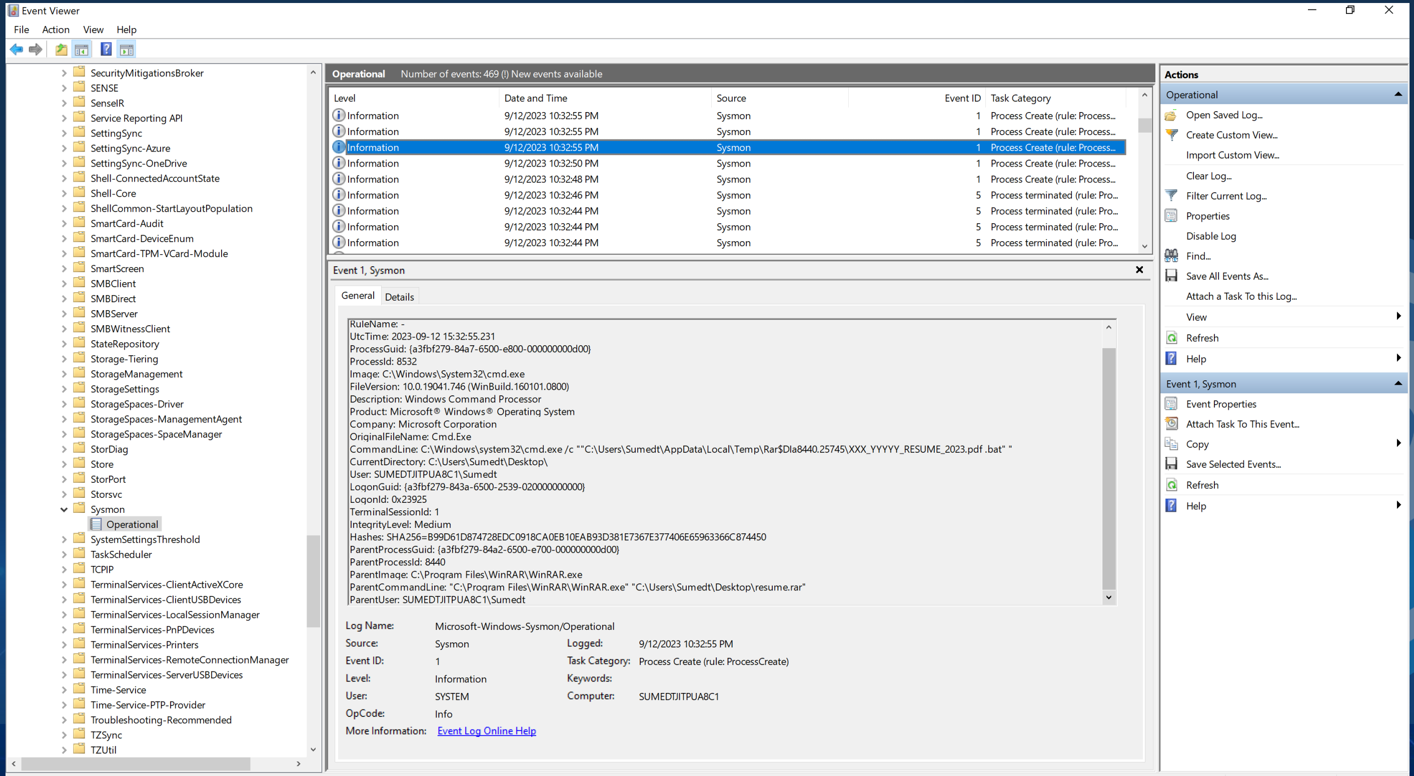
Parent: Non-existent process (8532)

Mitigation policies: DEP (permanent); ASLR (high entropy); CF Guard

Protection: None

Proactive Threat Hunting

WINRAR CVE-2023-38831



Operational Number of events: 469 (!) New events available

Level	Date and Time	Source	Event ID	Task Category
Information	9/12/2023 10:32:55 PM	Sysmon	1	Process Create (rule: Process...
Information	9/12/2023 10:32:55 PM	Sysmon	1	Process Create (rule: Process...
Information	9/12/2023 10:32:55 PM	Sysmon	1	Process Create (rule: Process...
Information	9/12/2023 10:32:50 PM	Sysmon	1	Process Create (rule: Process...
Information	9/12/2023 10:32:48 PM	Sysmon	1	Process Create (rule: Process...
Information	9/12/2023 10:32:46 PM	Sysmon	5	Process terminated (rule: Pro...
Information	9/12/2023 10:32:44 PM	Sysmon	5	Process terminated (rule: Pro...
Information	9/12/2023 10:32:44 PM	Sysmon	5	Process terminated (rule: Pro...
Information	9/12/2023 10:32:44 PM	Sysmon	5	Process terminated (rule: Pro...

Event 1, Sysmon

```
RuleName: -
UtcTime: 2023-09-12 15:32:55.231
ProcessGuid: {a3fbf279-84a7-6500-e800-000000000d00}
ProcessId: 8532
Image: C:\Windows\System32\cmd.exe
FileVersion: 10.0.19041.746 (WinBuild.160101.0800)
Description: Windows Command Processor
Product: Microsoft® Windows® Operating System
Company: Microsoft Corporation
OriginalFileName: Cmd.Exe
CommandLine: C:\Windows\system32\cmd.exe /c ""C:\Users\Sumedt\AppData\Local\Temp\Rar$DIa8440.25745\XXX_YYYYY_RESUME_2023.pdf .bat" "
CurrentDirectory: C:\Users\Sumedt\Desktop\
User: SUMEDTJITPUA8C1\Sumedt
LogonGuid: {a3fbf279-843a-6500-2539-020000000000}
LogonId: 0x23925
TerminalSessionId: 1
IntegrityLevel: Medium
Hashes: SHA256=B99D61D874728EDC0918CA0EB10EAB93D381E7367E377406E65963366C874450
ParentProcessGuid: {a3fbf279-84a2-6500-e700-000000000d00}
ParentProcessId: 8440
ParentImage: C:\Program Files\WinRAR\WinRAR.exe
ParentCommandLine: "C:\Program Files\WinRAR\WinRAR.exe" "C:\Users\Sumedt\Desktop\resume.rar"
ParentUser: SUMEDTJITPUA8C1\Sumedt
```

Log Name: Microsoft-Windows-Sysmon/Operational
Source: Sysmon
Logged: 9/12/2023 10:32:55 PM
Event ID: 1
Task Category: Process Create (rule: ProcessCreate)
Level: Information
Keywords:
User: SYSTEM
Computer: SUMEDTJITPUA8C1
OpCode: Info
More Information: Event Log Online Help

Proactive Threat Hunting

WINRAR CVE-2023-38831

https://play.secplayground.com/lab/997

Sigma Rule: https://github.com/SigmaHQ/sigma/blob/master/rules-emerging-threats/2023/Exploits/CVE-2023-38831/file_event_win_exploit_cve_2023_38331_winrar_susp_double_ext.yml

Yara Rule:
- https://isc.sans.edu/diary/Analysis+of+RAR+Exploit+Files+CVE202338831/30164/
- https://yaraify.abuse.ch/yarahub/rule/EXPLOIT_WinRAR_CVE_2023_38831_Aug23/

Secure D

your secure daemon

Threat Hunting

Reference

- Hunting: Discovering Hidden Threats
- Cyber Threat Hunting Workshop