

# Unveiling shadows: key tactics for tracking cyber threat actors, attribution, and infrastructure analysis

Hossein Jazi

Senior Threat Intelligence Specialist



# Hossein Jazi

@h2jazi

## THREAT INTELLIGENCE SPECIALIST

- APT researcher
- Malware reverse engineer
- Threat Hunter
- Cyber crime investigator





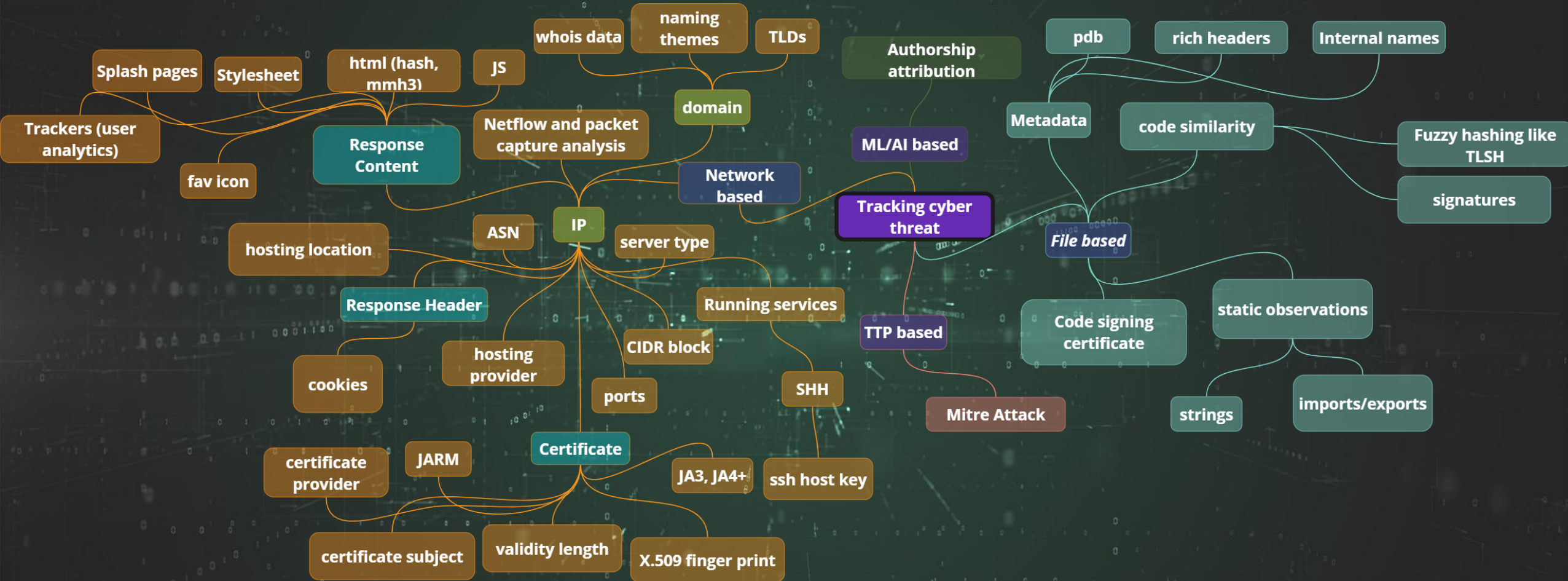
# APT Tracking





# APT Tracking

High Level Detail





# File-based Tracking





# File-based Tracking

Static Indicators – Strings



## DACLS RAT

- **Static Indicators:**
  - Extracted from static analysis of a toolset or malware used by threat actors.
- **Key use:**
  - Identifying and tracking malware variants associated with specific threat actors.
- **Specific Strings:** Unique strings within the malware can serve as indicators to detect and link new variants to known threats.
- **DACLS RAT:**
  - Certificate name and private key
  - “c\_2910.cls” and “k\_3872.cls,”





# File-based Tracking

Static Indicators – Imports/Exports

## IMPORTS/EXPORTS

- Examining import/export names in malware to uncover new malware families linked to the same threat actor.
- **Imphashing for Malware Tracking:**
  - Generates a hash from the Import Address Table (IAT) of an executable file.
  - Comparing imphashes reveals related malware samples and variants.
- **Benefits:**
  - Identifies new malware variants by detecting similar or identical imphashes.
  - Useful in tracking malware with code reuse by threat actors.
- **LocalPotato Exploit (FIN11):**
  - Imphash analysis helps track different variants of this NTLM exploitation tool, which exploits CVE-2023-21746 for privilege escalation.





# File-based Tracking

## Code Similarity

### LAZARUS APT

- **Code similarity:**
  - Compares malware samples to identify commonalities and relationships between different threats.
- **Key concepts:**
  - Code Reuse: Malware authors often reuse snippets or entire modules, linking malware to specific authors or groups.
  - Structural Analysis: Examining control flow graphs and function calls reveals hidden similarities.
  - Behavioral Patterns: Techniques for persistence, evasion, or exploitation may indicate a common origin.
- **APT Tracking:**
  - APTs reuse encryption keys, algorithms, API hashing, and C2 communications.
  - Reverse Engineering: Advanced techniques identify unique patterns for tracking via YARA rules.

```
signed_int64 __fastcall String_decoder(int64 a1, int64 a2, Int64a3)
(
    __int64 v3; // r10
    char v4; // ril
    signed__int64 result; // rax
    unsigned int v6; // er9
    __int64 v7; //rbx
    char v8; // cl

    a1 = (signed int)a3;
    v3 = a2;
    v4 = -124;
    result = 1461817411164;
    v6 = 162112194;
    if ((signed int)a3 > 0i64)
    {
        v7 = a1 - a2;
        do
        {
            v8 = *(_BYTE *) (v7 * v3++);
            *(BYTE )(v3 - 1) = v4 ^ result ^ v6 ^ v8;
            v4 = v4 & result ^ v6 & (v4 ^ result);
            v6 = (v6 >> 8) | (((unsigned int16)v6 ^ (unsigned int16)(8 * v6)) & 0x7F8) << 20;
            result = ((unsigned int)result >> 8) | (((_DWORD)result << 7) ^ ((unsigned int)result ^ 16
                * ((unsigned int)result ^ 2 * (_DWORD) result)) & 0xFFFFFFFF80) << 17;

            --a3;
        }
        while ( 43 );
    }
    return result;
}
```







# File-based Tracking

Fuzzy hashing

## GAMAREDON APT

- Fuzzy Hashing:
  - Identifies similarities between files or data, even when they are not identical.
  - Unlike traditional hashing, fuzzy hashes reflect degrees of similarity, useful for detecting variations of malicious files.
- Key Fuzzy Hashing Algorithms:
  - ssdeep: Detects similarities in spam emails and malware.
  - sdhash: Generates similarity digests for digital forensics, useful for file comparison.
  - TLSH: Robust against minor changes, used for clustering similar malware samples and identifying new variants.
- APT Tracking:
  - Links common tools used by APTs across incidents.
  - Compares hashes of files/scripts to track shared infrastructure and identify common servers/resources.

	Similarity	Detections	Size	First seen	Last seen	Submitters
12_00_12.12.2023.xhtml	61.11%	1 / 60	25.98 KB	13:55:57	14:12:11	1
ace3107a14f66685bb363fa6a30127015d58e26cd4... ...66685bb363fa6a30127015d58e26cd44cca61e6172fb375b068ab.bin	59.72%	22 / 59	30.92 KB	2023-01-19 07:18:28	2023-02-22 18:10:18	3
c40822ac3386c2cc0690e9a5dbf0e74bdf945466cd... No meaningful names	59.72%	8 / 63	29.10 KB	2023-10-03 13:09:24	2023-10-03 13:09:24	1
9289c37d19731eb04c25dc055748407691a93bc564... 2202300000000275.xhtml	58.33%	0 / 59	29.22 KB	2023-09-05 04:52:29	2023-09-05 04:52:29	1
c255b222f74da4737821cce386f223918241547862... ...da4737821cce386f223918241547862c6acc7df0acf682066322b.bin	58.33%	19 / 60	27.70 KB	2022-10-25 14:41:29	2022-12-22 09:00:36	





# File-based Tracking

Fuzzy Hashing



## OILRIG -SAITAMA BACKDOOR

tlsh:T1B9454B3BD722FDDAD7BF3D7090142D621C883D57A7714768FA4829EA26B7200DF1A168

Smart search 🔍 ⬆️

FILES - 9 / 9

	Similarity	Detections	Size	First seen	Last seen	Submitters	
<input type="checkbox"/> Confirmation Receive Document.xls <a href="#">xls</a> <a href="#">handle-file</a> <a href="#">run-dll</a> <a href="#">auto-open</a> <a href="#">environ</a> <a href="#">powershell</a> <a href="#">macros</a> <a href="#">create-ole</a> <a href="#">write-file</a> <a href="#">exe-pattern</a> ...	100%	41 / 66	1.17 MB	2022-04-26 10:21:39	2022-04-26 10:21:39	1	
<input type="checkbox"/> dttcodexgigas.d57ff42e1e53341cd34ded32960dfe902168c8ce <a href="#">xls</a> <a href="#">open-file</a> <a href="#">auto-open</a> <a href="#">create-dir</a> <a href="#">exe-pattern</a> <a href="#">handle-file</a> <a href="#">url-pattern</a> <a href="#">macros</a> <a href="#">enum-windows</a> <a href="#">environ</a> ...	97.22%	34 / 61	1.17 MB	2022-05-11 02:46:30	2022-05-11 02:46:30	1	
<input type="checkbox"/> 84ca7cdceed0cc0731f7cf6af3d19d87c49331fc19aae65db444b... <a href="#">doc</a> <a href="#">open-file</a> <a href="#">create-dir</a> <a href="#">exe-pattern</a> <a href="#">handle-file</a> <a href="#">malware</a> <a href="#">macros</a> <a href="#">run-dll</a> <a href="#">environ</a> <a href="#">calls-wmi</a> <a href="#">write-file</a> ...	83.33%	32 / 60	1.16 MB	2021-04-02 07:09:38	2021-07-27 09:50:31	2	
<input type="checkbox"/> -DF37582D178C55641E.TMP <a href="#">doc</a> <a href="#">write-file</a> <a href="#">run-dll</a> <a href="#">environ</a> <a href="#">create-dir</a> <a href="#">exe-pattern</a> <a href="#">create-ole</a> <a href="#">handle-file</a> <a href="#">open-file</a> <a href="#">macros</a>	83.33%	36 / 62	1.16 MB	2023-09-30 17:48:48	2023-09-30 17:48:48	1	
<input type="checkbox"/> -WRF{82955F28-CB09-406B-B05A-4CB840248AB1}.tmp <a href="#">doc</a>	81.94%	2 / 61	1.16 MB	2023-09-30 17:48:48	2023-09-30 17:48:48	1	
<input type="checkbox"/> -DF1B050D0A62810725.TMP <a href="#">doc</a>	73.61%	1 / 61	1.06 MB	2023-09-30 17:48:48	2023-09-30 17:48:48	1	
<input type="checkbox"/> 11dae9f628d35e003d6c08669cf9f4f35ce6e0fdd376bd757942... <a href="#">doc</a> <a href="#">open-file</a> <a href="#">create-dir</a> <a href="#">exe-pattern</a> <a href="#">handle-file</a> <a href="#">runtime-modules</a> <a href="#">malware</a> <a href="#">checks-network-adapters</a> ...	73.61%	39 / 62	1.26 MB	2021-04-02 07:32:03	2021-07-27 09:50:38	3	
<input type="checkbox"/> M0.xls <a href="#">xls</a> <a href="#">obfuscated</a> <a href="#">open-file</a> <a href="#">auto-open</a> <a href="#">handle-file</a> <a href="#">create-file</a> <a href="#">detect-debug-environment</a> <a href="#">macros</a> <a href="#">calls-wmi</a> ...	59.72%	29 / 62	1.86 MB	2021-09-29 12:51:57	2021-09-29 12:51:57	1	
<input type="checkbox"/> MailFirst.txt <a href="#">vba</a> <a href="#">obfuscated</a> <a href="#">handle-file</a> <a href="#">run-dll</a> <a href="#">write-file</a> <a href="#">anti-analysis</a>	50%	13 / 59	1.82 MB	2021-09-29 12:57:22	2021-09-29 12:57:22	1	





# File-based Tracking

Signature – YARA rules

## YARA RULES

- What are YARA Rules?
  - A powerful tool used to identify and classify malware based on patterns.
  - **Features:**
    - **Patterns:** Matches textual or binary patterns within files.
    - **Formats:** Includes plain text, hexadecimal strings, regular expressions, and wildcards.
    - **Conditions:** Incorporates complex conditions and boolean logic for precise targeting.
    - **Modularity:** Reusable and effective for various tasks.
- Applications of YARA Rules
  - **Detection:** Identify unique encryption algorithms or code reuse.
  - **Tracking:** Link malware samples to APT groups like Lazarus.
  - **Variants:** Detect and respond to known malware variants.

## TURLA APT

```
rule SFX_TUR_AND_KOPILUWAK
[
  meta:
    author = "H2J"
    description = "Strings contained in SFX"

  strings:
    $str1 = "Setup=c:\\windows\\system32\\wscript.exe /b"
    $str2 = "Silent=1"

  condition:
    all of them
]
```





# File-based Tracking

AV Signatures

## LAZARUS APT

- Specialized Signatures developed by companies can be used to track APT-linked malware.
- Lazarus APT Campaign
- **Malware:** BeaverTail
- **Detection:** AV signatures used to identify new samples associated with the attack campaign.

eset\_nod32:JS/Spy.NukeSped.A

Smart search

FILES - 20 / 117

First seen desc X

Sort by Filter by Export Tools Help

		Detections	Size	First seen	Last seen	Submitters	
<input type="checkbox"/>	c189c82ef3c1e986c2ba599d68505fa88f74236a629f00061bdb06d8b951e5... index.js <small>javascript   idle   long sleeps</small>	15 / 62	8.21 KB	2024-06-13 08:58:14	2024-06-13 08:58:14	1	
<input type="checkbox"/>	87d5917f5c0113d7b2db511538f3a386717a0bf9fd2b2f494516d5e08564aa... test.zip <small>zip   sets process name   detect debug environment</small>	19 / 66	5.58 MB	2024-06-13 06:42:16	2024-06-13 06:42:16	1	
<input type="checkbox"/>	44cc9f16ac993080653f5017cb4bc2ad01111a9fac8277b848b56cd5e8eacc... server.zip <small>zip</small>	8 / 65	31.02 KB	2024-06-12 20:33:16	2024-06-12 21:06:33	1	
<input type="checkbox"/>	523ac26438c647e5aa63977da7243049fad5b83f08ef33da37a115402d5893... vegan-robs-dao-dapp-master.zip <small>zip</small>	8 / 65	1.62 MB	2024-06-12 19:46:49	2024-06-13 08:56:48	3	
<input type="checkbox"/>	77f4e7e51767b78f86588563feb95fbc1f465160e447f475684fe7742c0e9e... vegan-robs-dao-main.zip <small>zip</small>	3 / 63	1.95 MB	2024-06-07 12:55:46	2024-06-07 12:55:46	1	
<input type="checkbox"/>	cf6cf894c35b8f84bbae815c1f41bed41f1acb870cb6cda57b3ee1e74e0c5... start-star.zip <small>zip   sets process name   detect debug environment</small>	19 / 65	5.58 MB	2024-06-06 09:18:32	2024-06-06 09:18:32	1	





# File-based Tracking

Code signing certificate

## LEAF MINER - WINTAPIX DRIVER

- Digital certificates used to sign software, ensuring authenticity and integrity.
- Attackers use stolen certificates to sign malware, making it appear legitimate and bypassing security measures.
- Analyzing certificate details (e.g., issuers, serial numbers) can help link malware to specific actors.
- WinTAPIX code signing certificates from:
  - "Beijing JoinHope Image Technology Ltd."
  - "VeriSign Class 3 Public Primary Certification Authority - G5."

		Detections	Size	First seen	Last seen
<input type="checkbox"/>	99b59f619388993695a7ef9cba74d8b9c0964b018245bb84a1cb8aeaf1e7d8... Pasting.gj_with_Memory/Driver/KernalDriver.sys peexe assembly overlay signed 64bits native	30 / 72	17.73 KB	2023-04-23 15:35:53	2024-02-03 13:55:50
<input type="checkbox"/>	8578bff36e3b02cc71495b647db88c67c3c5ca710b5a2bd539148550595d03... WinTapix.sys peexe assembly invalid-signature signed overlay native 64bits	56 / 74	1.13 MB	2023-02-12 08:56:12	2023-05-30 04:21:12
<input type="checkbox"/>	1485c0ed3e875cbd26d18ea9d31727deb8df290a1c00c780419a... WinTapix.sys peexe invalid-signature 64bits signed assembly native overlay	55 / 74	1.13 MB	2022-09-01 06:45:18	2023-11-22 07:55:48
<input type="checkbox"/>	d8120e28d4fa324d51977dcf36c8fa6f1f61f0fbfdcc233854d88c4783d2a1... C:\Windows\system32\drivers\apld.sys peexe assembly overlay signed 64bits native	46 / 69	40.98 KB	2021-07-23 12:10:33	2023-06-16 16:25:50
<input type="checkbox"/>	8b93df65fc1dc3c5246c872c41935c1176af5e4ac1765716667da9c3bf2293... Monitor.exe peexe overlay signed	3 / 73	3.11 MB	2021-01-28 05:34:28	2024-03-16 06:20:17





# File-based Tracking

## METADATA

### METADATA

- Auxiliary information embedded within malware files.
- Helps track, analyze, and attribute malicious activities.
- **File Metadata:**
  - **Timestamps:** Creation, modification, and access times.
  - **File Properties:** Size, version information, and digital signatures.
- **Compiler Metadata:**
  - **Rich Headers:** Information about the compiler, linker, and build environment.
  - **PDB Paths:** Debugging symbols or paths to Program Database files.



# File-based Tracking

Rich headers

## TURLA APT

- Encoded metadata in Portable Executable (PE) files that contains information about compiler version, linked libraries, and build environment details.
- **Rich Header Hashing**
  - Generate a cryptographic hash from the decoded Rich header data.
  - Acts as a fingerprint for the build environment.
- **Applications for Tracking APTs**
- **Clustering:** Identifies malware samples with similar build environments.
- **Attribution:** Links new samples to known threat actors based on shared development practices.

```
rule APT_RU_Turla_Gazer_Embedded_Resource_RichHeader
{
  meta:
    description = "lets get weird - track embedded Gazer payloads based on their shared rich headers"
    hash = "d0b169d2e753191a5c366a863d216bc5a9eb5e173f0bd5a61f126c4fd16484ac"
    hash = "473aa2c3ace12abe8a54a088a08e00b7bd71bd66cda16673c308b903c796bec0"
    hash = "a65bc4adbd61c098acf40ef81dc8b6b10269af0d9ebbd18b48439df76c18cb3"
    DaysofYARA_day = "94/100"
    author = "Greg Lesnewich"
  condition:
    for any var_rsrc in pe.resources: (
      uint16be(var_rsrc.offset) == 0x4d5a and
      hash.md5(var_rsrc.offset+0x80, 0x80) == "dd006bd9a43c05c9457a9e6b9e1636ca")
    )
}
```







# File-based Tracking

PDB

## APT 29

- Program Database files that store debugging information.
- Symbols, source line information, and stack traces.
- **Function & Variable Names:** Reveals malware structure and behavior.
- **Unstripped PDB Files:** Provides deeper insights if included by accident.
- **PDB Strings and Paths**
- Specific information within PDB files.
- Paths to source code files, directory structures, usernames, project names.

The screenshot shows a search interface for the file "googledrivesucks\drive.pdb". It displays two search results in a table format. The first result is for a file with ID 295452a87c0fbb48eb87be9... and the second is for a file with ID cd54155a6b240de1b610653... Both results show detection counts, sizes, and dates.

		Detections	Size	First seen	Last seen	Submitters	
<input type="checkbox"/>	GoogleDrive.exe peexe assembly runtime-modules detect-debug-environment ...	46 / 72	1.24 MB	2022-10-21 13:35:45	2022-10-21 13:35:45	1	
<input type="checkbox"/>	GoogleDrive.exe peexe assembly overlay runtime-modules ...	17 / 70	2.00 MB	2022-08-05 08:27:20	2022-08-05 08:27:20	1	



# TTP-based Tracking





# TTP-based Tracking

Mitre Attack

Reconnaissance 10 techniques	Resource Development 8 techniques	Initial Access 10 techniques	Execution 14 techniques	Persistence 20 techniques	Privilege Escalation 14 techniques	Defense Evasion 43 techniques	Credential Access 17 techniques	Discovery 32 techniques	Lateral Movement 9 techniques
Active Scanning (0/3)	Acquire Access	Content Injection	Cloud Administration Command	Account Manipulation (0/6)	Abuse Elevation Control Mechanism	Abuse Elevation Control Mechanism (0/6)	Adversary-in-the-Middle (0/3)	Account Discovery (0/4)	Exploitation of Remote Services
Gather Victim Host Information (0/4)	Acquire Infrastructure (0/8)	Drive-by Compromise	Command and Scripting Interpreter (0/10)	BITS Jobs	Abuse Elevation Control Mechanism (0/6)	Access Token Manipulation (0/5)	Brute Force (0/4)	Application Window Discovery	Internal Spearphishing
Gather Victim Identity Information (0/3)	Compromise Accounts (0/3)	Exploit Public-Facing Application	Container Administration Command	Boot or Logon Autostart Execution (0/14)	Access Token Manipulation (0/5)	BITS Jobs	Credentials from Password Stores (0/6)	Browser Information Discovery	Lateral Tool Transfer
Gather Victim Network Information (0/6)	Compromise Infrastructure (0/8)	External Remote Services	Deploy Container	Boot or Logon Initialization Scripts (0/5)	Account Manipulation (0/6)	Build Image on Host	Exploitation for Credential Access	Cloud Infrastructure Discovery	Remote Service Session Hijacking (0/2)
Gather Victim Org Information (0/4)	Develop Capabilities (0/4)	Hardware Additions	Exploitation for Client Execution	Browser Extensions	Boot or Logon Autostart Execution (0/14)	Debugger Evasion	Forced Authentication	Cloud Service Dashboard	Remote Services (0/8)
Phishing for Information (0/4)	Establish Accounts (0/3)	Phishing (0/4)	Inter-Process Communication (0/3)	Compromise Host Software Binary	Boot or Logon Initialization Scripts (0/5)	Deobfuscate/Decode Files or Information	Forge Web Credentials (0/2)	Cloud Service Discovery	Replication Through Removable Media
Search Closed Sources (0/2)	Obtain Capabilities (0/7)	Replication Through Removable		Create Account (0/2)		Deploy Container	Input Capture	Cloud Storage Object Discovery	Software
						Direct Volume Access		Container and Resource Discovery	
						Domain or Tenant Policy Modification (0/2)			





# TTP-based Tracking

Monitor unique behaviors



## LAZARUS

behaviour\_processes:"pcaua.exe -a %APPDATA%" not tag: peexe

Files	Detections	Size	First seen	Last seen	Submitters
ff44d3dabb82467cd21187039789314bf70c15777fcdc93ada1059ed4c28... veafdsag.msi runtime-modules direct-cpu-clock-access	21 / 62	543.00 KB	2022-07-29 09:18:16	2022-09-28 06:28:54	4
8eec3ac9f7d1ac64fc7397ba57cdac4f56959d1512f71dded60e831a26e076... unknown long-command-line-arguments hiding-window uri-pattern runtime-modules high-entropy direct-cpu-clock-access ...	32 / 61	388.28 KB	2022-07-27 23:43:20	2022-08-03 22:01:01	3
eaeef808e1ac99d13481b23b9dbdb6d246b14fea35db0b592412dc213e619c... unknown long-command-line-arguments hiding-window uri-pattern runtime-modules detect-debug-environment ...	25 / 60	2.09 KB	2022-07-27 23:48:15	2022-08-03 19:03:31	2
73f684b87139927012db12ec8d92824bd2404102e17ec427b302c79a4314d4... C:\Users\user\AppData\Local\Temp\hob.lnk long-command-line-arguments hiding-window uri-pattern runtime-modules direct-cpu-clock-access	26 / 61	2.09 KB	2022-08-03 13:50:47	2022-08-03 13:50:47	1
bcafd808237f1f29bdae4e45d75d925fb4fa3d7fe7f85abb953ba3f33aa29e... C:\Users\user\AppData\Local\Temp\hob.lnk long-command-line-arguments hiding-window uri-pattern runtime-modules detect-debug-environment ...	9 / 51	2.09 KB	2022-08-01 09:37:24	2022-08-01 09:37:24	1
2e0abe6352af0924f9d9cc098230fa25e1f05cf4ae43a2d0b71405dedfe029... unknown uri-pattern hiding-window long-command-line-arguments executes-dropped-file	31 / 60	2.08 KB	2022-07-29 13:25:14	2022-08-01 01:10:39	2
e40347e7cd335b43a0d27b33521684754f66d1e3be714f22a68e27d396a63f... unknown long-command-line-arguments hiding-window uri-pattern runtime-modules detect-debug-environment high-entropy ...	24 / 61	276.65 KB	2022-07-29 09:13:10	2022-07-30 23:52:17	4
9dc813afe2ff8963696691d5092b9ea779048c63ba1a8232e8ae81f0f1a476... C:\Users\user\AppData\Local\Temp\abo.lnk long-command-line-arguments hiding-window uri-pattern runtime-modules detect-debug-environment ...	15 / 61	2.09 KB	2022-07-29 23:51:29	2022-07-29 23:51:29	1
bcfa523b1d55fcadc89bfee8f7c9abac3497f7760485843e6f58fe666984aa... C:\Users\user\AppData\Local\Temp\abo.lnk long-command-line-arguments hiding-window uri-pattern runtime-modules direct-cpu-clock-access executes-dropped-file	16 / 61	2.07 KB	2022-07-29 09:18:13	2022-07-29 09:18:13	1





# Network-based Tracking

Infrastructure tracking





# Network-based Tracking

## INFRASTRUCTURE TRACKING

- Monitoring and analyzing APT components and systems (domains, IP addresses, servers, communication channels, etc.)
- Uncover APT infrastructure, identify patterns, and link attacks to the same threat actor
- Key Techniques
  - Passive DNS Analysis
  - SSL/TLS Certificate Analysis
  - WHOIS Data Analysis
  - Network Traffic Analysis
  - Fingerprinting Techniques
- Challenges
  - **Continuous Monitoring:** Need for up-to-date intelligence
  - **Data Correlation:** Complexity in linking disparate sources
  - **Evasion Tactics:** APTs can rapidly change infrastructure to avoid detection





# Network-based Tracking

Infrastructure Reuse

## LAZARUS - BEAVERTAIL

- Some APT groups reuse IP addresses and domains due to resource constraints
- Reused infrastructure is common across different campaigns
- VirusTotal nethunt: A new feature that enhances infrastructure tracking
  - Purpose: Allows users to create Yara rules
  - Function: Identifies samples using the same infrastructure
- **Campaign Detection:** Detect new campaigns that reuse existing infrastructure
- **Campaign Analysis:** Identify if samples are part of ongoing or previously analyzed campaigns

```
rule C2_IPs_Lazarus
{
  meta:
    author = "Fortinet TI"
    description = "Rule to find files that contact malicious IPs used by North Korea's Lazarus group"
    target_entity = "ip_address"
  condition:
    // CTI-194
    vt.net.ip.raw matches /^67.203.7.171/ or
    vt.net.ip.raw matches /^147.124.212.89/ or
    vt.net.ip.raw matches /^147.124.214.129/ or
    vt.net.ip.raw matches /^147.124.214.131/ or
    vt.net.ip.raw matches /^147.124.214.237/
}
```





# Network-based Tracking

## Passive DNS

### PASSIVE DNS

- Collecting and storing historical DNS resolution data passively
- Domain names, IP addresses, query types, timestamps
- **Infrastructure Insights:** Analyze domain name resolutions and IP address associations
- **Malicious Domain Identification:** Detect patterns and connections to known APT campaigns
- **Attribution:** Map out related domains and IPs to attribute attacks to specific threat groups
- **Example: Monitoring APT42**
- Initial Domain: jpostpress.com (Created January 2022)
- Resolved IP: 91.195.240.12
- Additional Domains Identified:
- themedealines.org (September 2022)
- maariv.net (September 2022)
- khaleejtimes.org (March 2023)
- Insight: Historical data reveals APT infrastructure and timelines



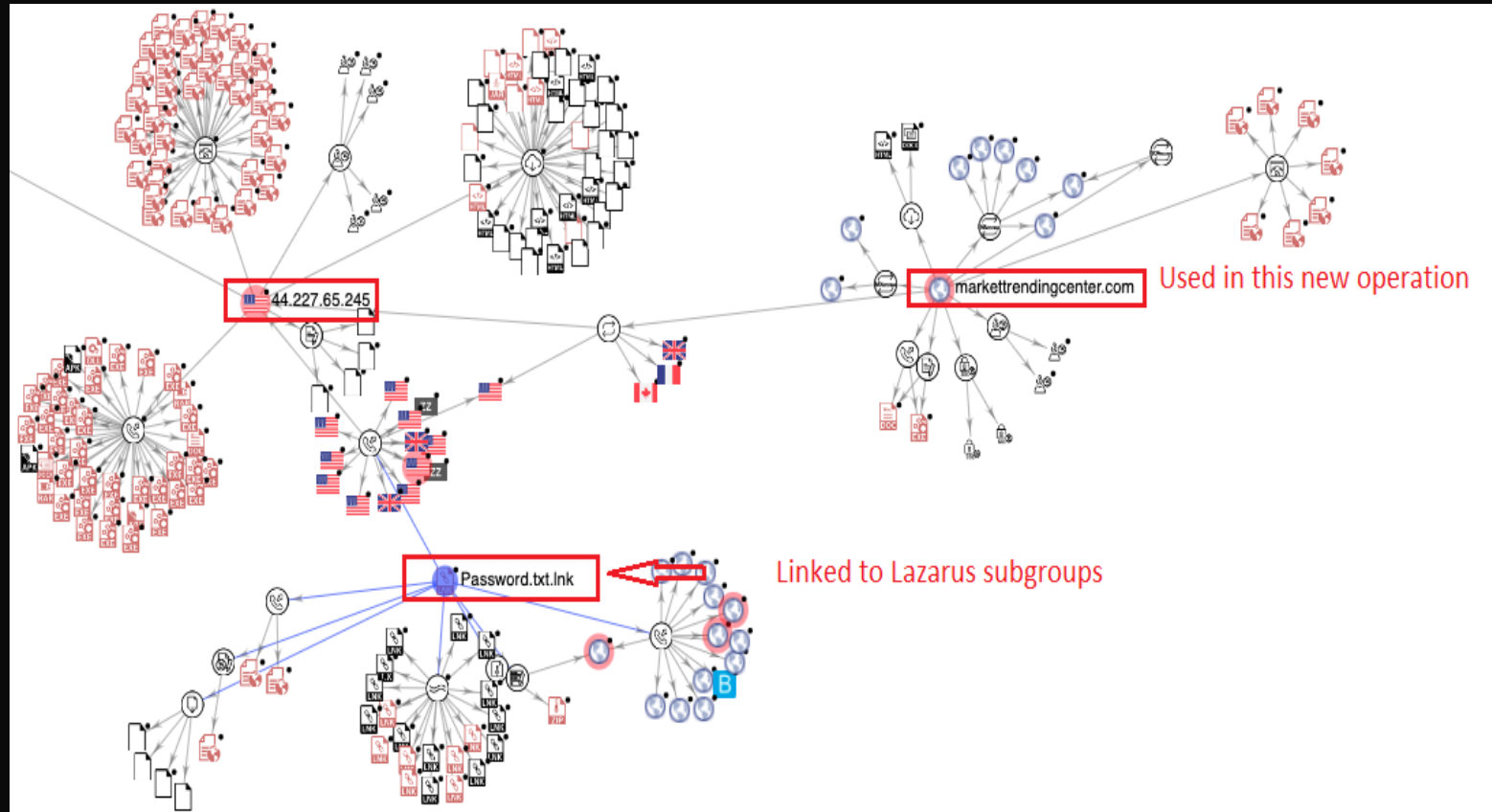


# Network-based Tracking

Passive DNS



LAZARUS





# Network-based Tracking

## IP Addresses

### IP ADDRESSES

- Key Information from IP Addresses:  
WHOIS Data
  - Autonomous System Numbers (ASNs)
  - Hosting Providers
  - CIDR Blocks
  - Server Types
  - Running Services and Ports
- **Tracking Techniques:**
  - Combine Multiple Methods: Actors may consistently use specific ASNs, hosting providers, and port sets
  - Example: ASN 20473 (CHOOPA) used by Chinese APTs like Vicious Panda and IndigoZebra
- Leveraging HTTP/HTTPS Services:
  - Artifacts in HTTP/HTTPS Services
    - Response Headers, Response Content, Certificates





# Network-based Tracking

## Certificates and Fingerprints

### SSL/TLS CERTS

- SSL/TLS Overview
  - Purpose: Secure communication over networks (data privacy & integrity)
- SSL/TLS Certificates:
  - Contains public key & identity information
  - Verified by a Certificate Authority (CA)
  - Ensures authentication & encrypted communication
- Fingerprinting Techniques
  - JA3: Creates a fingerprint of TLS clients using fields from the Client Hello message
  - JA3+: Extends JA3 with more handshake details for better accuracy
  - JA4/JA4+: Focus on fingerprinting TLS servers
  - JARM: Developed by Salesforce, fingerprints TLS servers based on responses to specific Client Hello messages
- Identifying APTs: Unique TLS handshake patterns help track APT tools & techniques



# Network-based Tracking

## Certificate values/hash

### KIM SUKY

- **Common Name (CN):** The primary identifier, often representing the domain name or server name.
- **Subject Alternative Names (SANs):** Additional domain names or IP addresses for which the certificate is valid.
- **Issuer:** The entity (CA) that issued the certificate, providing trustworthiness information.
- **Serial Number:** A unique identifier assigned by the issuer for distinguishing the certificate.
- **Validity Period:** The time frame in which the certificate is valid.
- **Public Key:** Used for encryption and authentication, associated with the private key
- **Signature Algorithm:** Specifies the security algorithm used to sign the certificate.
- **Key Usage:** Defines the cryptographic operations allowed for the public key.
- **Extended Key Usage (EKU):** Specifies additional purposes beyond the basic key usage.

The screenshot shows a search interface for hosts. The search query is `services.certificate="9de541b039cfdb96c7810df49efd958b28cc2df73e314f67c1a9"`. The results are displayed under the heading "Hosts" with "Results: 4" and "Time: 0.32s".

Two host entries are visible:

- 74.50.84.190:** Linux, IS-AS-1 (19318), New York, United States. Services include database, remote-access, and open-dir. 1 Matched Service: 443/HTTP. 2 Other Services: 22/SSH, 3306/MYSQL.
- 206.72.198.157:** Linux, IS-AS-1 (19318), New Jersey, United States. Services include remote-access. 1 Matched Service: 443/HTTP. 2 Other Services: 22/SSH, 80/HTTP.





# Network-based Tracking

## JARM Fingerprint

### STORM0558

- Use JARM fingerprint consistent with SoftEther VPN
- x509 certificate has expiration date of December 31, 2037
- Subject information within the x509 certificate does not contain "softether"



services.jarm.fingerprint: 06d06d07d06d06d06c42d42d000000cdb95e27fd8f9fee4 Search

**1 Matched Service**  
1765/HTTP

**1 Other Service**  
1723/PPTP

**66.57.75.86 (syn-066-057-075-086.biz.spectrum.com)**  
TWC-11426-CAROLINAS (11426) North Carolina, United States  
atlassian-jira-issue-collector login-page

**1 Matched Service**  
443/HTTP

**2 Other Services**  
81/HTTP 82/HTTP

**78.156.125.21**  
Microsoft Windows NORLYS-FIBERNET (39642) Central Jutland, Denmark  
email

**1 Matched Service**  
5555/HTTP

**3 Other Services**  
25/SMTP 80/HTTP 110/POP3



# Network-based Tracking

## HTTP Response Headers

### HTTP RESPONSE HEADERS

- Metadata sent by a web server to a client, providing instructions on how the client should handle the received data.
  - Sent as part of an HTTP response
  - Structured as name-value pairs.
- **APT Tracking Using Response Headers:**  
Response headers offer clues in tracking Advanced Persistent Threats (APTs) through:
  - **User-Agent Analysis:** Detecting spoofed or unusual user-agent strings.
  - **Host & Referer Tracking:** Identifying suspicious requests.
  - **Cookie Analysis:** Uncovering irregular cookie patterns.
  - **Custom Headers:** Identifying non-standard headers in malware traffic.
  - **ETag & Cache-Control:** Monitoring ETag hashes or caching rules.
  - **HTTP Method Anomalies:** Tracking unusual HTTP method usage.
  - **Content-Type & Encoding:** Inspecting content types for malware.



# Network-based Tracking

## HTTP Response Headers

### EXAMPLES OF APT TRACKING VIA HTTP HEADERS

- **Unique Header Combinations & Patterns**
  - Some APT groups use distinctive header combinations, aiding identification.
  - Regex patterns help identify abnormal headers in network traffic.
- **Case Studies**
  - **MuddyWater (Iranian Cyber-Espionage Group)**  
Utilizes unique ETag hashes on its VPS-hosted web servers:
    - **ETag hashes:** 2aa6-5c939a3a79153, 2aa6-5b27e6e58988b, 2aa6-5c939a773f7a2
    - Traces back to servers used in malicious activities.
  - **Cobalt Strike Framework**  
Utilizes the 'CS-Bid' header for C2 communication:
    - Detecting outbound HTTP traffic with this header can reveal compromised hosts and C2 server connections.



# Network-based Tracking

## HTTP Response Content

### UNDERSTANDING HTTP RESPONSE CONTENT IN APT TRACKING

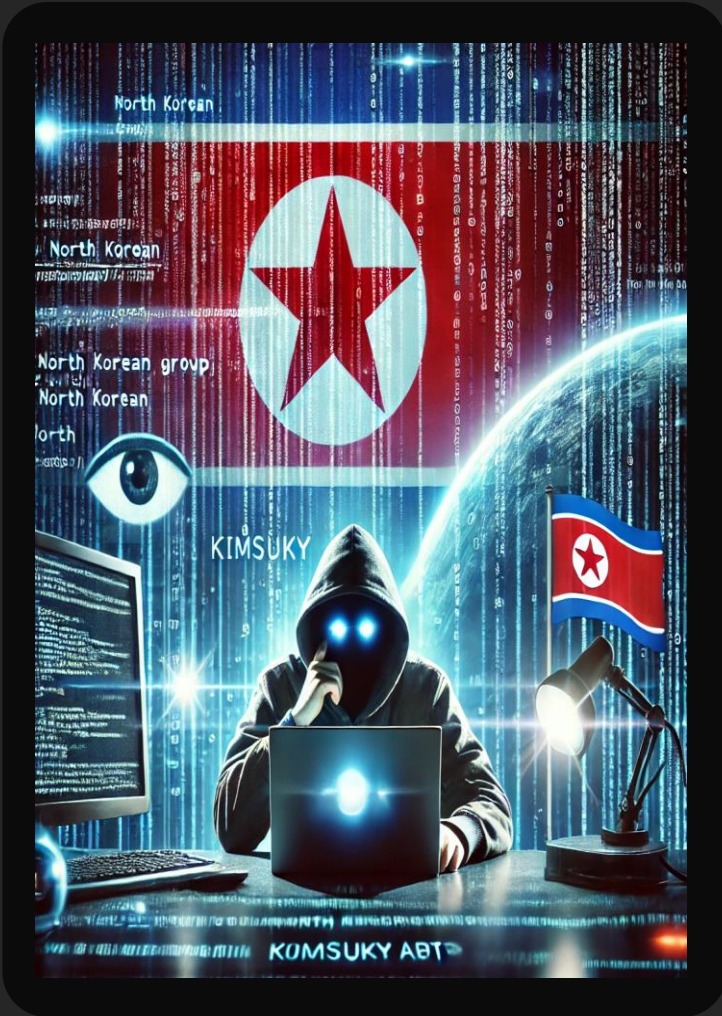
- HTTP response content refers to data sent by a web server in response to a client's request.
- Why It Matters: Key elements of HTTP responses (HTML, JavaScript, CSS, etc.) can reveal critical details about APT activities, helping to track and identify malicious actors.
- **Key Components:**
  - HTML Hashes
  - CSS Patterns
  - JavaScript Code
  - Splash Pages
  - Favicons
  - Resource URLs
  - Custom Error Pages
  - Dynamic Content





# Network-based Tracking

HTTP Response Content



KIMSUKY

services.http.response.body="Million OK !!!!"

**221.143.46.49**  
Microsoft Windows SKB-AS SK Broadband Co Ltd (9318) Seoul, South Korea  
remote-access network-administration  
2 Matched Services  
80/HTTP 4433/HTTP  
6 Other Services  
443/HTTP 2301/HTTP 2381/HTTP 2401/UNKNOWN 3389/RDP  
47001/HTTP

**118.193.68.225**  
Microsoft Windows UCLOUD-HK-AS-AP UCLOUD INFORMATION TECHNOLOGY HK LIMITED (135377) Seoul, South Korea  
network-administration remote-access  
2 Matched Services  
80/HTTP 443/HTTP  
1 Other Service  
3389/RDP

**118.42.186.25**  
Microsoft Windows KIXS-AS-KR Korea Telecom (4766) Chungcheongnam-do, South Korea





# Network-based Tracking

## HTTP Response Content

### Q B O T

- Qbot Malware: Researchers used unique HTML titles in HTTP response content to track servers related to Qbot malware operations.

The screenshot shows a network analysis tool interface with a search bar containing the query: `services.http.response.html_title:"Slack is your productivity platform | Slack"`. The results are displayed for three IP addresses:

- 194.50.233.216 (ip-195-50-233-216-ae.rack400.com)**
  - OS: Ubuntu Linux, ASN: M247 (9009), Location: Dubai, United Arab Emirates
  - Tags: clearbit-reveal, onetrust, remote-access
  - 1 Matched Service: 443/HTTP
  - 4 Other Services: 80/HTTP, 4369/EPMD, >\_ 9011/SSH, >\_ 39744/SSH
- 37.27.93.218 (static.218.93.27.37.clients.your-server.de)**
  - ASN: HETZNER-AS (24940), Location: Uusimaa, Finland
  - Tags: onetrust, moment.js, vue.js
  - 1 Matched Service: 80/HTTP
  - 7 Other Services: 443/HTTP, 8080/HTTP, 2053/HTTP, 8443/HTTP, 2083/HTTP, 2096/HTTP
- 45.77.55.4**
  - OS: Ubuntu Linux, ASN: AS-CHOOPA (20473), Location: Hesse, Germany
  - Tags: onetrust, remote-access



# Network-based Tracking

Tracking and Attribution using Netflow

## NETFLOW ANALYSIS

- **Source/Destination IP Addresses:**
  - Identifying the IPs involved can reveal connections to known malicious actors or infrastructure.
- **Port Numbers:**
  - Specific port usage patterns (e.g., uncommon or high ports) can indicate malicious activity tied to certain malware.
- **Flow Duration:**
  - Long-duration connections or frequent short connections can signal APT activity, especially when tied to C2 traffic.
- **Traffic Volume:**
  - Anomalous spikes in data transfer can indicate exfiltration.
- **Protocol:**
  - Identifying specific protocols like DNS tunneling or unusual HTTP activity aids attribution.



# Tracking Threat Actors with Visual Artifacts





# APT Tracking

## Visual Artifacts

### VISUAL ARTIFACTS

- Threat actors often reuse icons, logos, or screenshots in malware campaigns.
- An example is the use of the same logo across phishing campaigns attributed to a specific APT group.
- **Image Hashing:** Using hashing techniques, analysts can generate unique hashes for images, detecting reused visual artifacts across different malware campaigns, which helps link activity to specific APT groups.
- **Example:**
  - **PDF files analysis:**
    - Generates a BMP image of the first page of PDFs, stored in specific directories.
    - Analysts can use BMP images as dropped files to identify similar PDFs, linking various documents to threat actors (e.g., Blind Eagle).



# APT Tracking

## Visual Artifacts



## LAZARUS

main\_icon\_dhash:1100353b1b800000 Smart search

↻ FILES - 6 / 6

		Sort by	Filter by	Export	Tools	Help	
		Detections	Size	First seen	Last seen	Submitters	
<input type="checkbox"/>	<a href="#">0d01b24f7666f9bccf0f16ea97e41e0bc26f4c49cdfb7a4dabcc0a494b44ec...</a> <a href="#">0d01b24f7666f9bccf0f16ea97e41e0bc26f4c49cdfb7a4dabcc0a494b44ec9b.docx</a> <a href="#">doc</a> <a href="#">macros</a> <a href="#">run-dll</a> <a href="#">persistence</a> <a href="#">long-sleeps</a> <a href="#">detect-debug-environment</a> <a href="#">url-pattern</a> <a href="#">auto-open</a> <a href="#">open-file</a> <a href="#">exe-pattern</a> ...	40 / 64	2.27 MB	2022-01-18 16:13:22	2024-09-10 04:29:14	61	<input type="checkbox"/>
<input type="checkbox"/>	<a href="#">87f9f137687187e8eecf98393a7885fdd55cf573d80d94443e9324daa5f49...</a> <a href="#">932.docx</a> <a href="#">doc</a> <a href="#">calls-wmi</a>	0 / 63	353.50 KB	2024-01-03 13:19:04	2024-09-09 05:52:21	7	<input type="checkbox"/>
<input type="checkbox"/>	<a href="#">b3218dbbc4417667bee19f637235e25afe0c5ff521151e5ee5bca5ec1d0455...</a> <a href="#">C:\Users\user\Desktop\readme.doc (copy)</a> <a href="#">doc</a> <a href="#">open-file</a> <a href="#">url-pattern</a> <a href="#">exe-pattern</a> <a href="#">malware</a> <a href="#">macros</a> <a href="#">run-dll</a> <a href="#">calls-wmi</a> <a href="#">auto-open</a>	39 / 66	4.24 MB	2024-08-08 19:23:36	2024-08-08 19:23:36	1	<input type="checkbox"/>
<input type="checkbox"/>	<a href="#">3c5d6b1e22ccd420ecbfc21354929604c8fbd1f3c66d25b89cb9bf13062f3...</a> <a href="#">C:\Users\user\AppData\Local\Temp\~DF269013A291D288E9.TMP</a> <a href="#">doc</a> <a href="#">open-file</a> <a href="#">exe-pattern</a> <a href="#">macros</a> <a href="#">run-dll</a> <a href="#">url-pattern</a>	32 / 63	4.29 MB	2022-11-20 14:15:06	2022-11-20 14:15:06	1	<input type="checkbox"/>
<input type="checkbox"/>	<a href="#">ef8f4d31e16cfb399b4c69e3d2c43ab33863d3149cb701209061aa5d575747...</a> <a href="#">C:\Users\user\Desktop\attachment.doc (copy)</a> <a href="#">doc</a> <a href="#">open-file</a> <a href="#">exe-pattern</a> <a href="#">macros</a> <a href="#">run-dll</a> <a href="#">url-pattern</a>	35 / 62	4.24 MB	2022-06-22 03:02:51	2022-06-22 03:02:51	1	<input type="checkbox"/>
<input type="checkbox"/>	<a href="#">d1b5961f259b9af90d8c191f679b996c7321d00a17363a7639eb74fa9428b9...</a> <a href="#">clean.doc</a> <a href="#">doc</a> <a href="#">open-file</a> <a href="#">exe-pattern</a> <a href="#">url-pattern</a> <a href="#">malware</a> <a href="#">macros</a> <a href="#">run-dll</a>	31 / 61	2.26 MB	2022-04-26 07:16:00	2022-04-26 07:16:00	1	<input type="checkbox"/>







# TOOLS

## TOOLS

- **Shodan:** A search engine for internet-connected devices, identifying vulnerabilities in servers, routers, and more.
- **Censys:** Scans and monitors internet devices, aiding in detecting malicious activities.
- **DomainTools:** Provides domain and DNS-related data, including WHOIS and historical records, to trace malicious domain infrastructure.
- **FOFA:** A search engine for discovering internet assets based on keywords, assisting in finding attack surfaces and IOCs.
- **VirusTotal:** Analyzes files and URLs to detect malware, aggregating insights from multiple antivirus engines.
- **Maltego:** A tool for data mining and link analysis, visualizing complex relationships between online entities.
- **ThreatConnect:** Integrates data from multiple sources to provide threat intelligence for collaborative analysis.
- **PassiveTotal:** Specializes in passive DNS and SSL analysis, helping track APT infrastructure and correlate threat indicators.







# Attribution of APTs

Connecting the Dots

## ATTRIBUTION

- Attribution identifies the individuals, groups, or nation-states behind cyberattacks based on evidence gathered from tracking APT activities.
- Key Elements of Attribution:
  - Technical Indicators:
    - Malware & Network Traffic Analysis
    - Forensic Investigation
  - Contextual Factors:
    - Geopolitical Insights
    - Historical Attack Patterns
    - Victimology
- How APT Tracking Supports Attribution:
  - Tracking Tools: Passive DNS, threat intelligence platforms, and IoCs help build detailed threat actor profiles.
  - Beyond Technical: Behavioral analysis, motives, and geopolitical context enhance accuracy in identifying threat actors.





# Attribution Challenges

## CHALLENGES

- Over-Reliance on IoCs
- Failure to correlate data across sources
- Attribution based on insufficient information
- Neglecting the role of False Flags and misdirection
- Ignoring evolving tactics and techniques
- Lack of contextual understanding
- Overlooking nontechnical evidence
- Inadequate mapping of attack infrastructure
- Overlooking relationships between infrastructure components
- Ignoring indicators of infrastructure evolution
- Inadequate pivoting strategies
- Over reliance of passive analysis
- Failure to use contextual information

**FORTINET®**