Malware Analysis Using ANY.RUN

Technical Requirements:

Internet connection

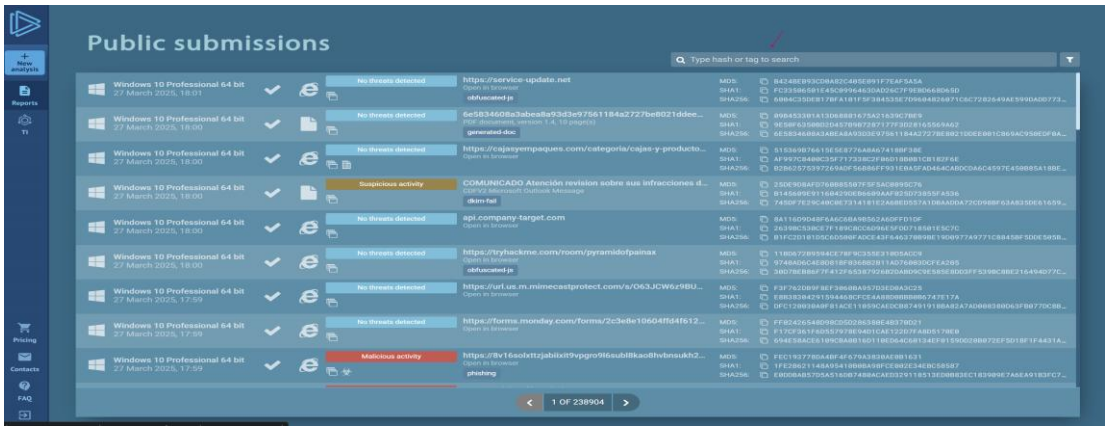
Web browser

Access to: https://app.any.run/submissions

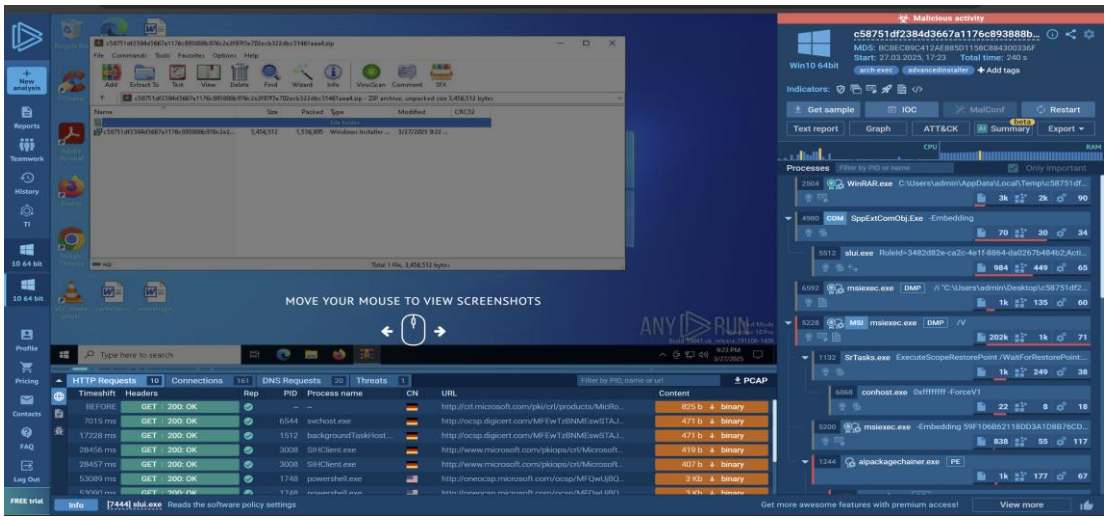
Objective:

Task 1:

In the top search bar, paste the following SHA-256 hash:
9a5b81c373040c445e9d521e7ef6faa42af378b1033d60aa93c56e7bbe90508b

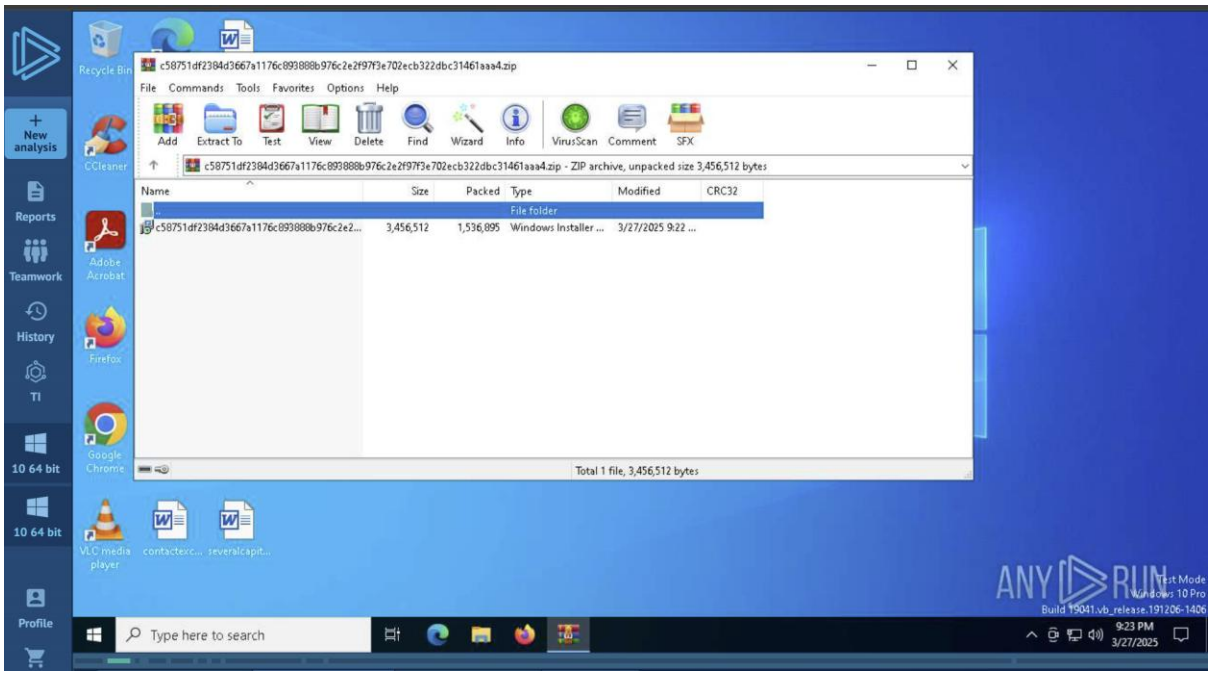


Once you search for a malware hash and find the result, click on the corresponding entry. You will then be redirected to a new page where you'll land on an interactive sandbox environment.



Task 2: Understanding the ANY.RUN Interface

Virtual Screen (Middle Area):

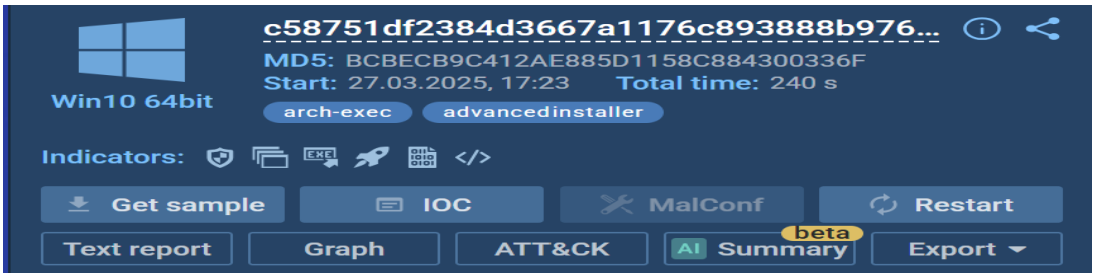


In this section, you'll see the malware running on a virtual Windows desktop. For this sample, a ZIP file is extracted using WinRAR, and then an MSI file inside it is executed. This screen helps you follow the basic steps: first, the archive is opened, and then the installer inside is launched. While you won't see anything too advanced here, it's still useful for confirming how the malware begins its execution.

PCAP / Network Panel (Bottom):

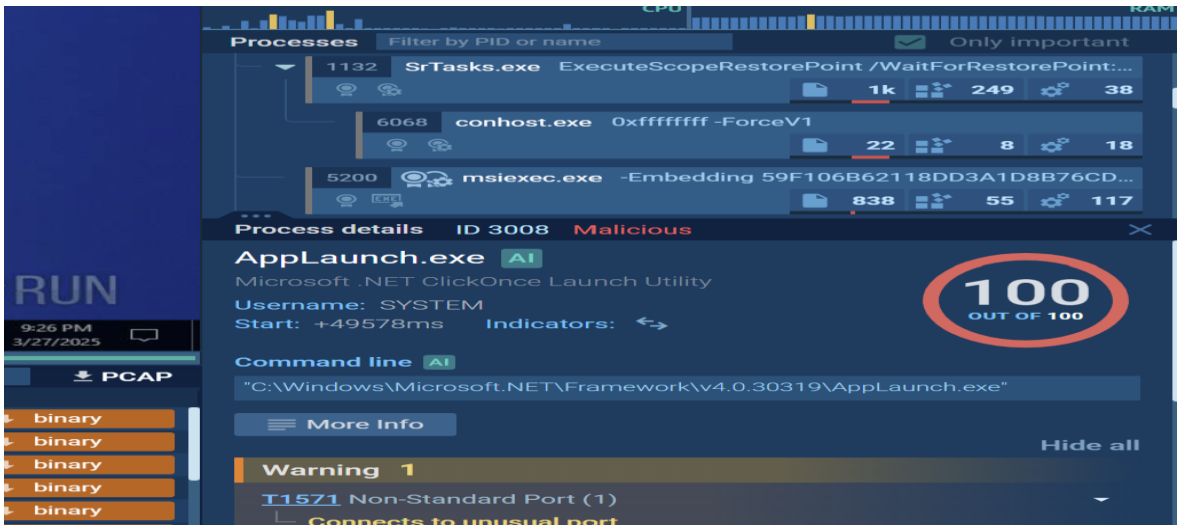
In this section, you can see all the network-related information generated during the malware execution. This includes details like HTTP requests made by the malware, domains it contacted, all the connections it made, DNS queries, and observed threats during network communication. It also shows which process was responsible for each network action. This helps you understand whether the malware tried to communicate with external servers, download additional files, or perform any suspicious network activity. You can also download the full PCAP (packet capture) file from this section if you want to analyze the network traffic in a tool like Wireshark.

File Info (Top-Right):



This section gives you key information about the file being analyzed. You can see the file name, when the analysis started, and how long it ran. At the top, there's a verdict bar that gives a quick indication of whether the file behaved suspiciously or showed signs of unusual activity. You'll also notice tags that describe how the file is packaged or executed, such as whether it's part of an installer or archive. Just below that, there are helpful shortcuts to view indicators of compromise (IOCs), download reports, or explore how the file's behaviour maps to known attack techniques using the MITRE ATT&CK framework. Overall, this section gives you a clear summary of what the file is and what it tried to do during the analysis.

Process Tree (Right):



This section shows a list of all the processes that were launched during the file's execution. They are arranged in a tree-like structure so you can easily see which process started another. Each process includes technical details like its ID and indicators of how it behaved. You will also see a score next to each one. A higher score usually means the process is acting more suspiciously. This makes it easier to identify the parts of the execution that may be linked to unusual or harmful activity.

Task 3: Use the virtual screen, verdict section, process tree, and network/PCAP panel to answer the following questions.

1) From the top-right verdict section, what does the analysis say about the file? (Malicious, Suspicious, etc.

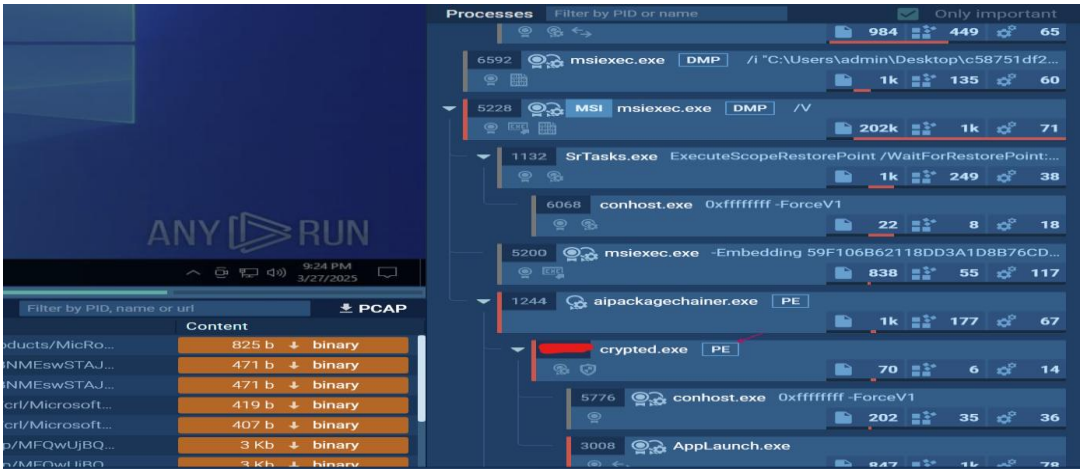
A) Malicious Activity

2)Using VirusTotal, check if any of the domains or IPs contacted by the sample are marked as suspicious or untrusted. (Hint: Go to the Network section>Connections tab and look for connections initiated by the application with the name AppLaunch.exe.)

A) AppLaunch.exe connected to IP 171.22.28.236, which is flagged as malicious by 11 security vendors.

3) In the Process Tree (right panel), scroll down and find the process named crypted.exe. What is its Process ID (PID)?

A) 6972



Further, view the PE (Portable Executable) information by clicking the PE button next to the crypted.exe process in the screenshot above. This opens a window showing detailed static information about the file.

In the Main tab, you can see file hashes (MD5, SHA1, SHA256), basic file type, and how it was compiled.

If you scroll down to the EXE section, you'll find technical details such as the Machine Type, which tells you what kind of system the executable is designed to run on. In this case, it shows Intel 386 or later, meaning it's built for 32-bit Windows systems.

Use that hash to search for the file on VirusTotal.

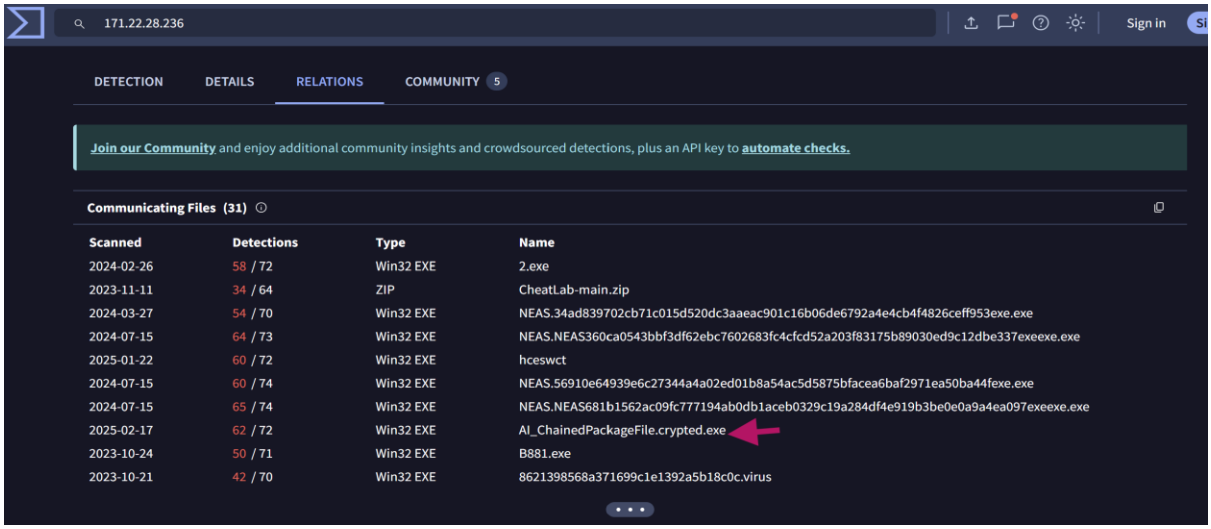
4) What malware family is this file related to? Describe its behavior based on the and Virus Total result and how many vendors flagged it as malicious.

The file crypted.exe is linked to the RedLine stealer and Mikey malware families. It acts like a trojan, meaning it secretly performs harmful actions like stealing information. VirusTotal shows that 62 out of 72 security tools detected it as malicious.

Once again, go to VirusTotal and search for the IP address that AppLaunch.exe tried to connect to, as shown in the Network panel of ANY.RUN. After opening the IP report, click on the Relations tab to see a list of files that have also communicated with this IP. Find the SHA256 hash of crypted.exe from ANY.RUN and compare it with the file hashes listed there. To view a file's hash in detail, simply click on its name. This will open a new tab with full information about that file on VirusTotal.

5) Does any file in the Relations tab match the hash of crypted.exe from ANY.RUN? If yes, provide the matching file name, and a screenshot as evidence.

Yes, the hash of crypted.exe matches the file AI_ChainedPackageFile.crypted.exe listed in the IP's Relations tab in Virus Total, confirming it's the same sample.



Task 4: At the top-right of the ANY.RUN interface, where the File Info section is located, you'll notice a button labeled "ATT&CK". This section is connected to the MITRE ATT&CK framework, which is used by security analysts to classify how malware behaves.

Clicking this button will open a new view that shows which tactics (goals) and techniques (methods) the malware sample used during execution. For example, it may list tactics like Defense Evasion or Execution, and techniques such as Process Injection or PowerShell abuse.

This helps you understand the real-world impact of the malware and how it fits into a full attack scenario.

Once you click on it, you'll be taken to a layout where tactics are listed across the top and related techniques appear underneath. Each item is clickable for more detail.

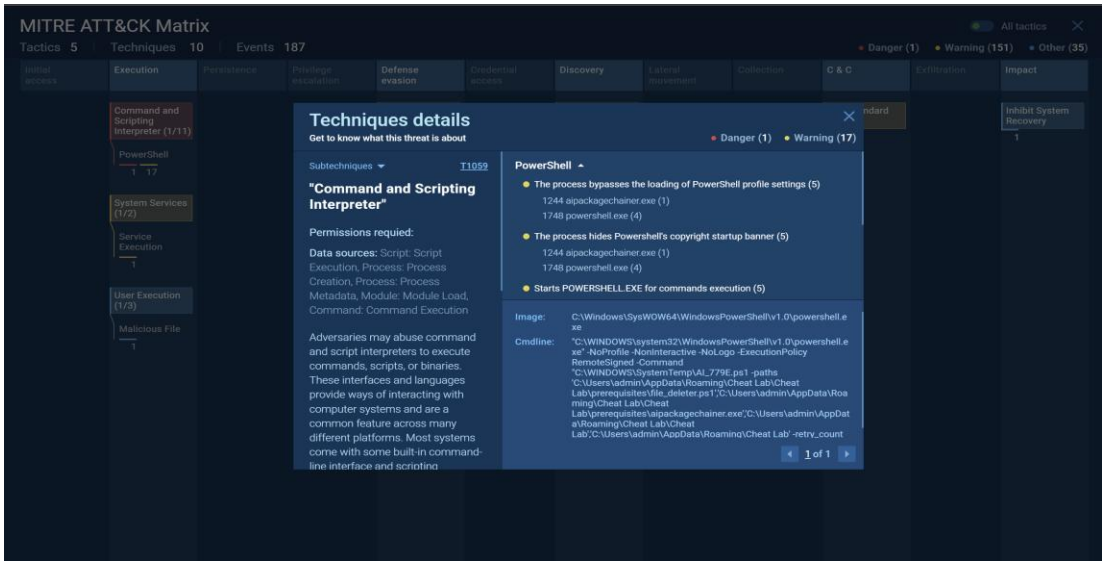
6) Based on the MITRE ATT&CK framework view, how many tactics and techniques are used by this file during its execution?

The sample uses 5 tactics and 10 techniques.

While exploring the MITRE ATT&CK section, you'll see that different tactics and techniques are highlighted using various colours. These colours help indicate how serious or confirmed the behavior is. Red or dark orange usually points to clearly suspicious or dangerous activity. Yellow or light orange suggests behaviour that might be suspicious but isn't fully confirmed. Blue is often used to show general or background activity. It may not be harmful on its own but is still useful for understanding the bigger picture of what the file was doing. Then, click on the Command and Scripting Interpreter technique, which is marked in red. It will display detailed information explaining why this execution behaviour is considered dangerous.

This will open a panel showing detailed behavior related to the use of PowerShell. As you can see in the screenshot below, the process uses PowerShell with parameters like -NoProfile, -NoLogo, and -ExecutionPolicy RemoteSigned, and it runs a script that deletes files and folders.

These options are often used to make PowerShell run quietly, avoid detection, and hide signs of execution.



7)Based on what you observe in this view, do you think the malware is using any anti-forensics techniques? Briefly explain your answer using examples from the command line or indicators shown in this section.

Yes, the malware uses PowerShell with options like NoProfile, NonInteractive, and ExecutionPolicy set to RemoteSigned to run quietly and avoid drawing attention. It also launches a script named AI_779E.ps1, which deletes itself, the original installer (aipackagechainer.exe), and the folders where the malware was stored. This is a clear attempt to hide its tracks and avoid being detected later a typical anti-forensics strategy.

Further, you can download the analysis report from the File Info Section (Text Report) here

End of Lab