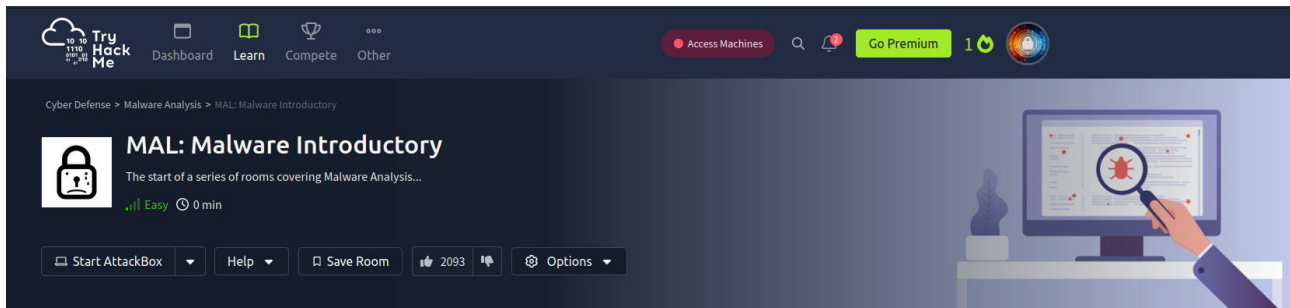




Eric Mwenda

Malware Introductory

<https://tryhackme.com/p/Ericm>



Purpose of Malware Analysis

Malware analysis not only a form of incidence response, but it is also useful in understanding how the behaviours of variants of malware result in their respective categorisation.

It is important to consider the following while carrying out Malware Analysis.

Point of Entry (PoE) - Was it through spam that our e-mail filtering missed and the user opened the attachment? Let's review our spam filters and train our users better for future prevention!

Indicators that malware has even been executed on a machine? Are there any files, processes, or perhaps any attempt of "un-ordinary" communication?

How does the malware perform? Does it attempt to infect other devices? Does it encrypt files or install anything like a backdoor / Remote Access Tool (RAT)?

Most importantly - can we ultimately prevent and/or detect further infection?!

Attacks can generally be classified into two types: **Targeted and Mass Campaign.**

Targeted Attack – This are malware attacks that are created for a specific purpose against a specific target.

Mass campaign – This are the most common type of attacks. The entire purpose of this type of Malware is to infect as many devices as possible and perform whatever it may - regardless of target.

Answer the questions below

What is the famous example of a targeted attack-esque Malware that targeted Iran?

Stuxnet

What is the name of the Ransomware that used the Eternalblue exploit in a "Mass Campaign" attack?

Wannacry

Identifying a malware attack has taken place.

The ultimate process of a malware attack can be broken down into a few broad steps:

- 1. Delivery** - This could be of many methods, to name a few: USB (Stuxnet!), PDF attachments through "Phishing" campaigns or vulnerability enumeration.
- 2. Execution** – This is the main part, what does it actually do? If it encrypts files - it's Ransomware! If it records information like keystrokes or displays adware - we can classify it as Spyware.
- 3. Maintaining persistence (not always the case!)**
- 4. Propagation (not always!)**

There are two categories of fingerprints that malware may leave behind on a Host after an attack:

- Host-Based Signatures
- Network-Based Signatures

Host-Based Signatures

These are generally speaking the results of execution and any persistence performed by the Malware.

Network-Based Signatures

This are observation of any networking communication taking place during delivery, execution and propagation.

Name the first essential step of a Malware Attack?

Delivery

Now name the second essential step of a Malware Attack?

Execution

What type of signature is used to classify remnants of infection on a host?

Host-Based Signatures

What is the name of the other classification of signature used after a Malware attack?

Network-Based Signatures

Static vs Dynamic Analysis

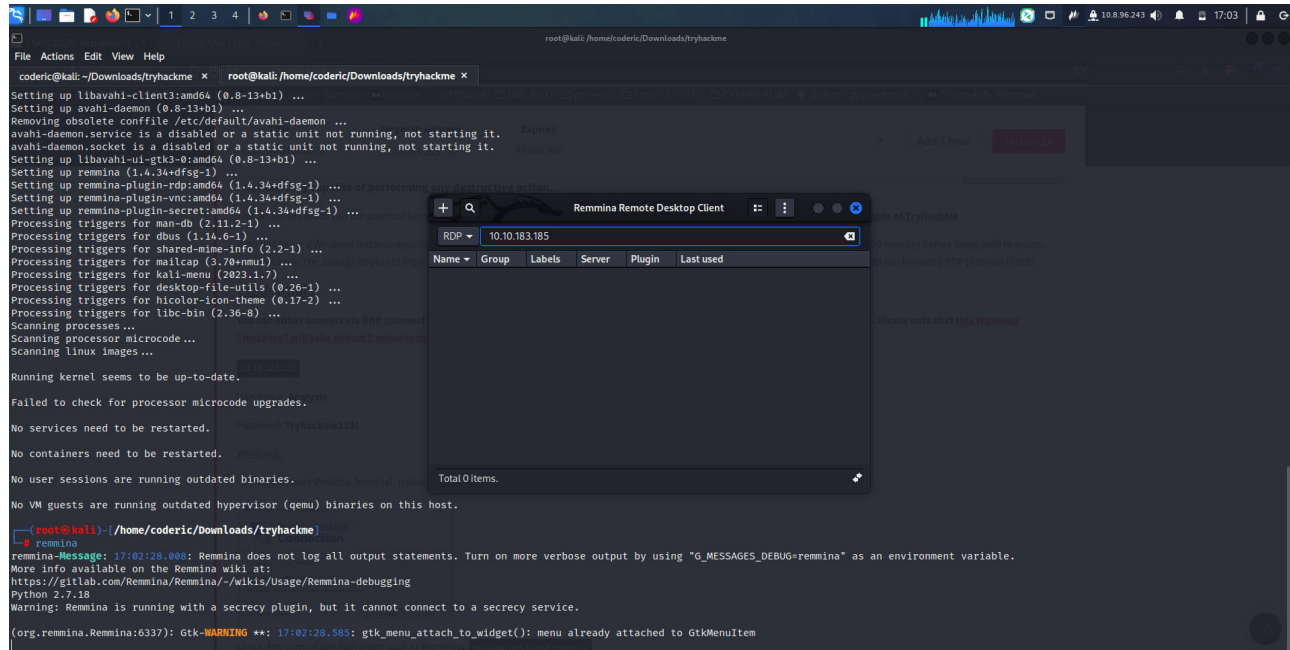
Static Analysis - This method analyses the sample at the state it presents itself as, without executing the code.

Dynamic Analysis - essentially involves executing the sample and observing what happens.

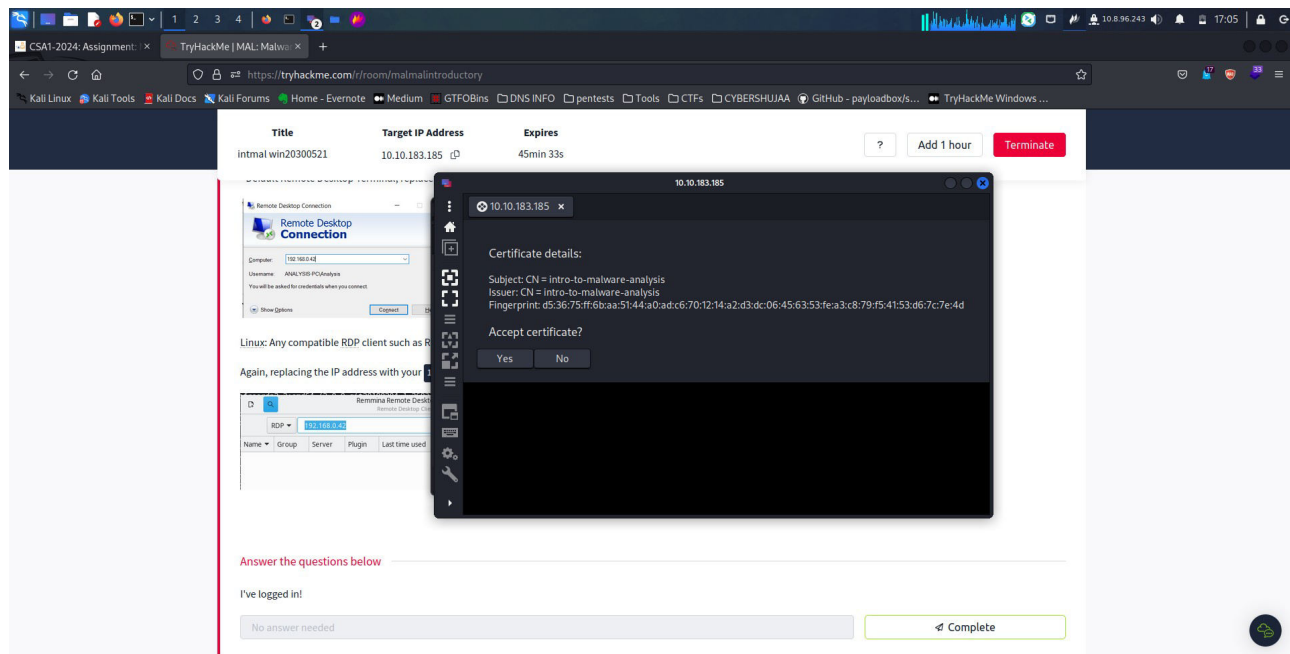
First was to start my machine to get the target machine.

Title	Target IP Address	Expires			
intmal win20300521	10.10.183.185	48min 5s	?	Add 1 hour	Terminate

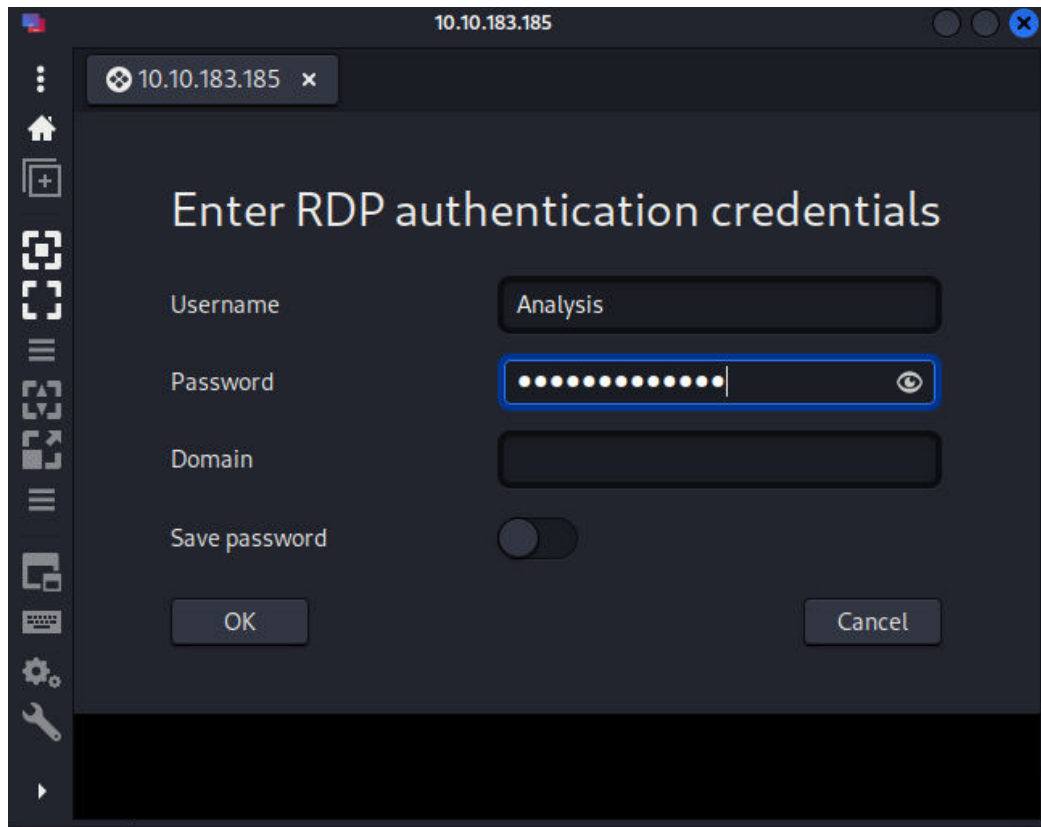
Once done I installed Remmina tool in kali to connect to the machine remotely.



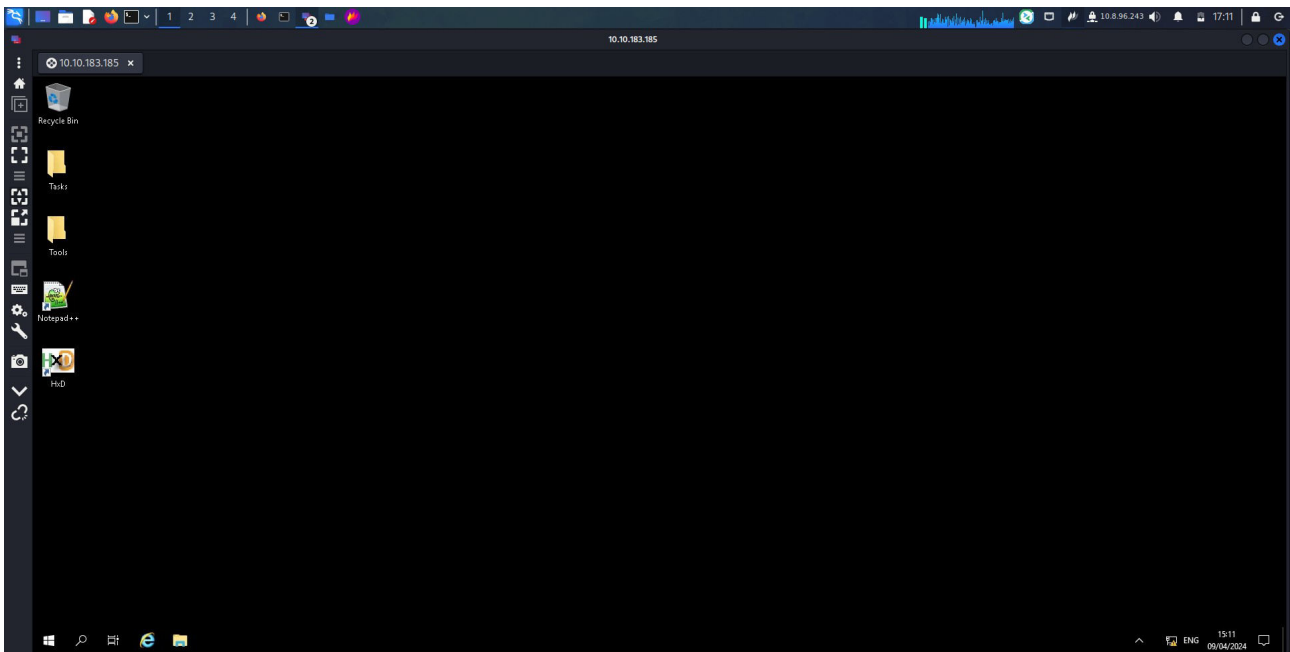
Connecting...



Keying in credentials.



Click OK to login.



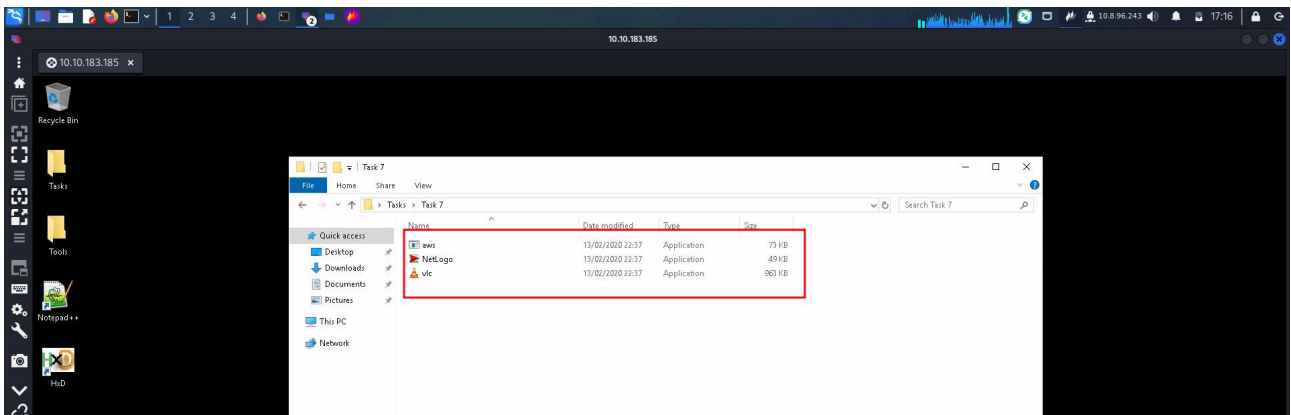
Am now logged in.

Obtaining MD5 Checksums of Provided Files

MD5 "Checksums" are a prominent attribute in the malware Community. Because there can be many variants of a family of Ransomware, these MD5 "Checksums" are cryptographic "fingerprints" of the files. This allows a uniformed identification throughout the community - especially with automated Sandboxes.

MD5 reveals its true identity.

Navigate to the "Tasks" Folder on the Desktop, and then enter the "Task 7" Directory, where there will be three files:

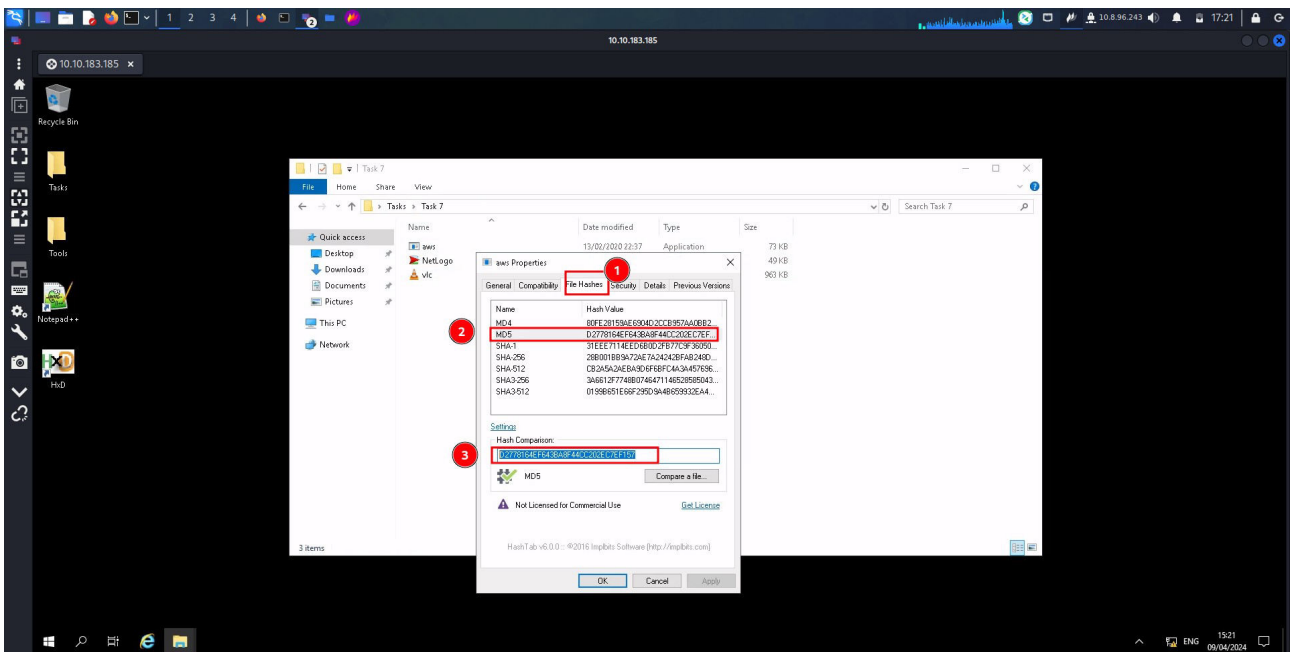


Answer the questions below

The MD5 Checksum of aws.exe

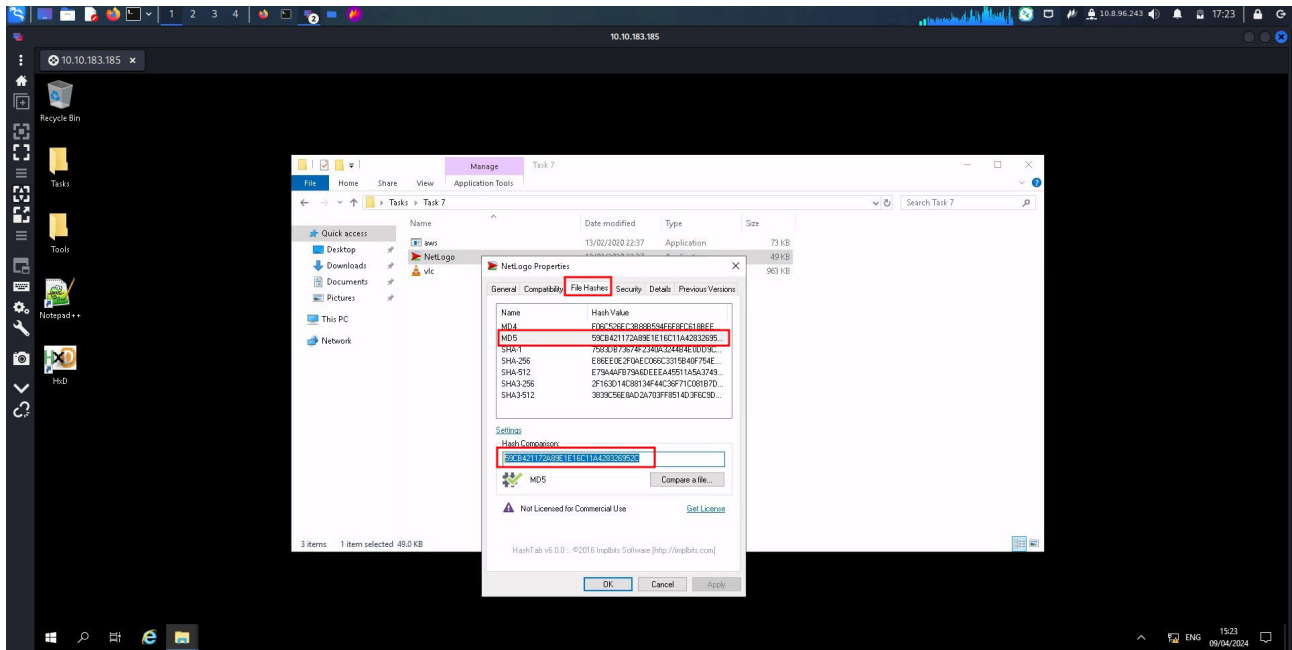
D2778164EF643BA8F44CC202EC7EF157

To find the MD5 Checksum you highlight the executable file > Right click and choose properties then check for the value under File Hashes Option.



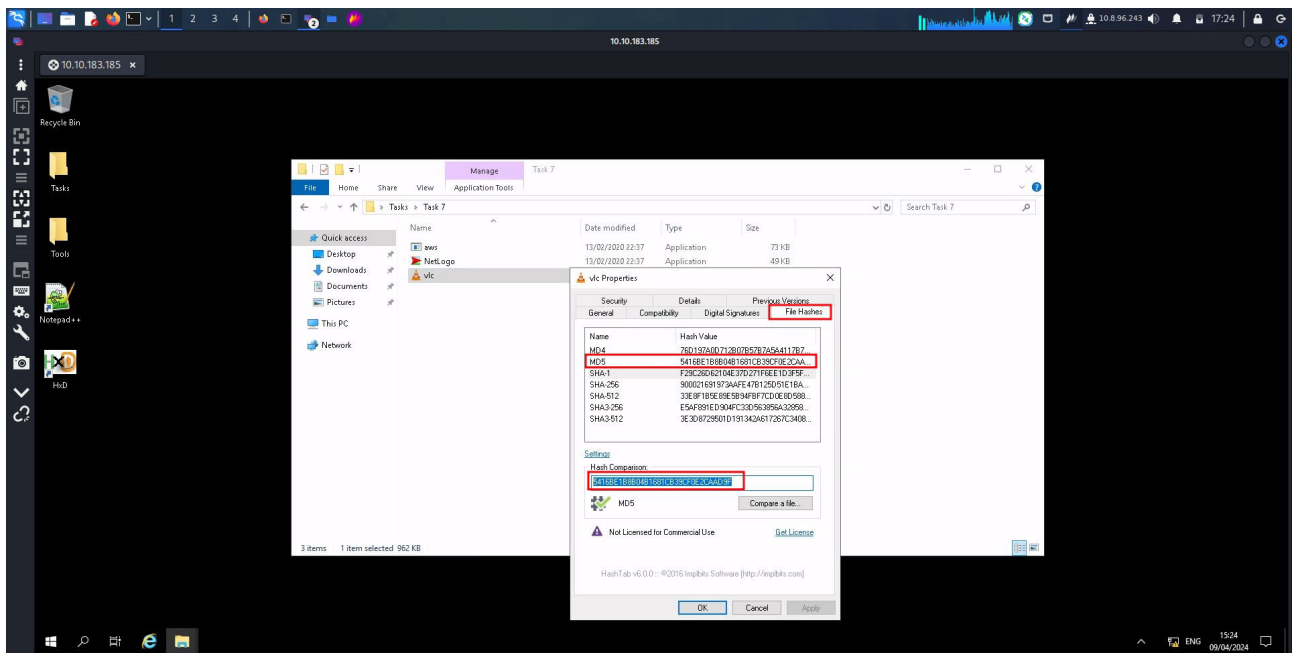
The MD5 Checksum of Netlogo.exe

59CB421172A89E1E16C11A428326952C



The MD5 Checksum of vlc.exe

5416BE1B8B04B1681CB39CF0E2CAAD9F

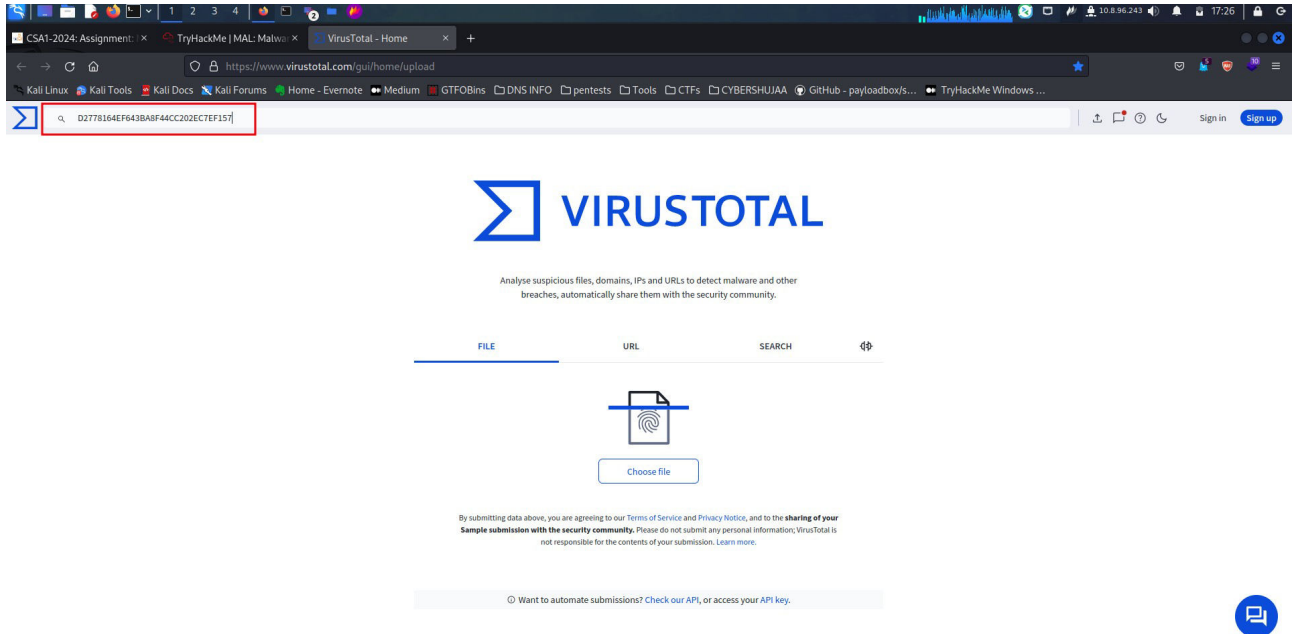


Now lets see if the MD5 Checksums have been analysed before

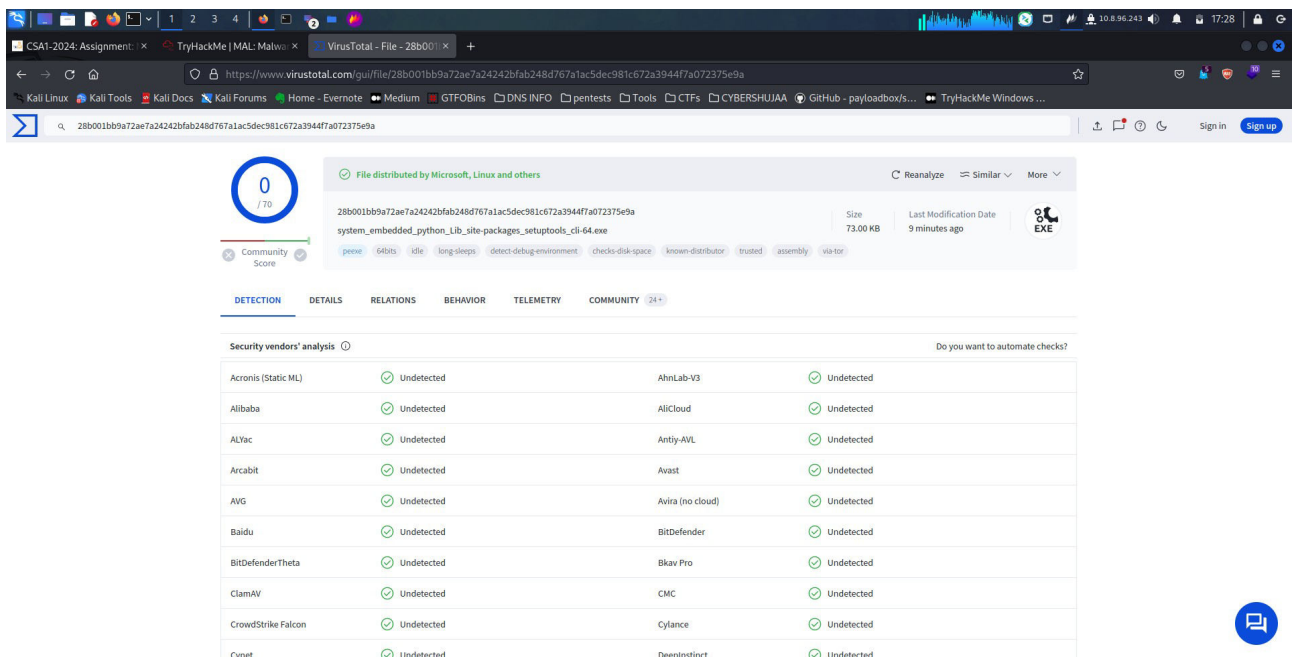
Answer the questions below

Does Virustotal report this MD5 Checksum / file aws.exe as malicious? **Nay**

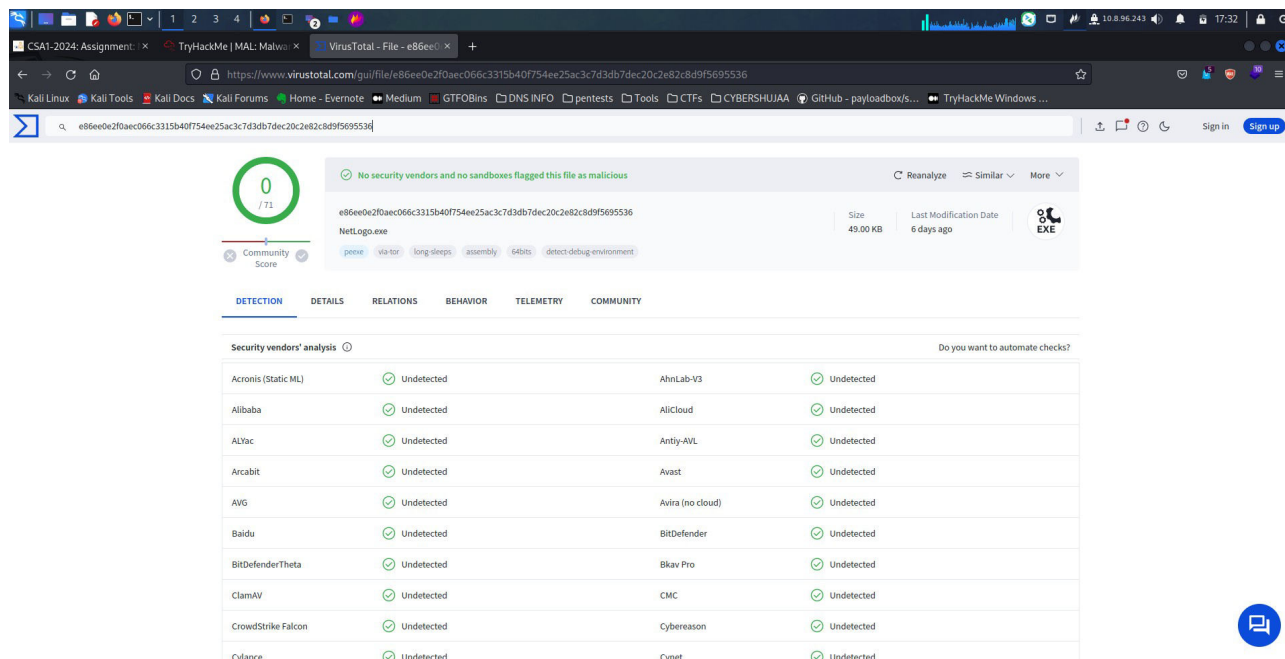
First thing is to copy and paste the MD5 Checksum obtained earlier for each file on the virus total web application to check for the report.



No report



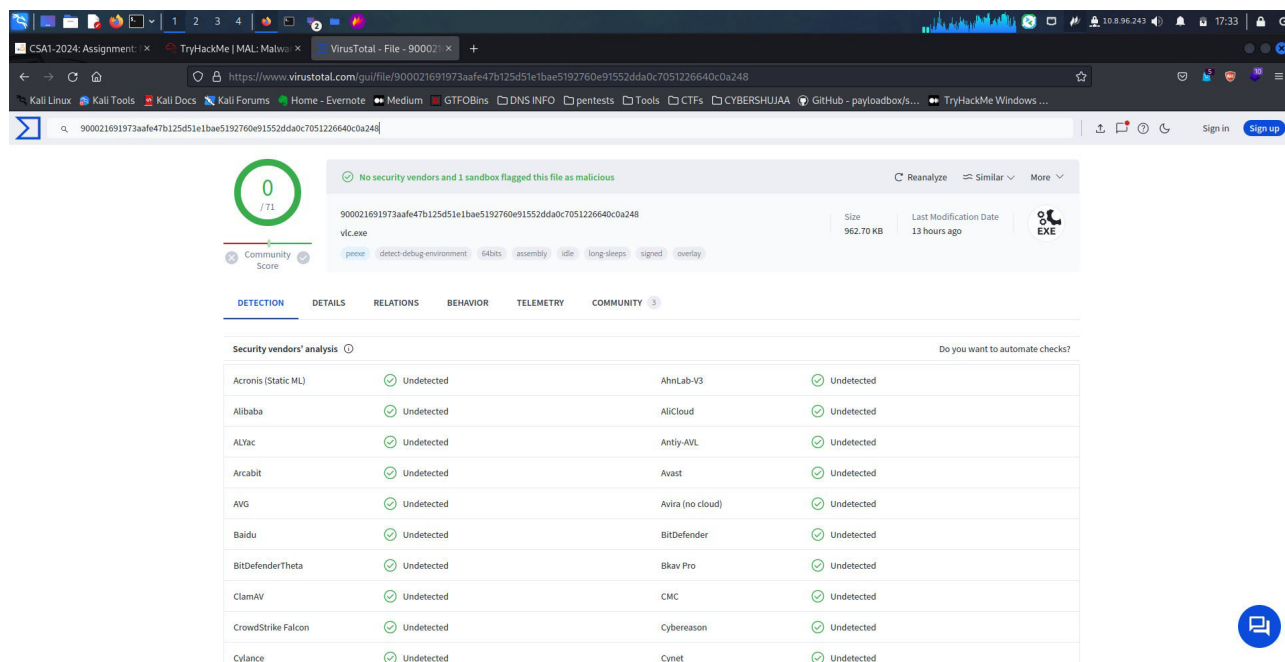
Does Virustotal report this MD5 Checksum / file Netlogo.exe as malicious? Nay



The screenshot shows the VirusTotal web interface for a file named NetLogo.exe. The file's MD5 checksum is e86ee0e2f0aec066c3315b40f754ee25ac3c7d3db7dec20c2e82c8d9f5695536. The file size is 49.00 KB, and it was last modified 6 days ago. The community score is 0/71. A green banner at the top states "No security vendors and no sandboxes flagged this file as malicious". Below this, a table titled "Security vendors' analysis" lists 16 vendors, all of whom report the file as "Undetected".

Vendor	Result
Acronis (Static ML)	Undetected
Alibaba	Undetected
ALYac	Undetected
Arcabit	Undetected
AVG	Undetected
Baidu	Undetected
BitDefenderTheta	Undetected
ClamAV	Undetected
CrowdStrike Falcon	Undetected
Cybereason	Undetected
Cyren	Undetected
AhnLab-V3	Undetected
AllCloud	Undetected
Antiy-AVL	Undetected
Avast	Undetected
Avira (no cloud)	Undetected
BitDefender	Undetected
Bkav Pro	Undetected
CMC	Undetected

Does Virustotal report this MD5 Checksum / file vlc.exe as malicious? Nay



The screenshot shows the VirusTotal web interface for a file named vlc.exe. The file's MD5 checksum is 900021691973aaf47b125d51e1bae5192760e915526da0c7051226640c0a248. The file size is 962.70 KB, and it was last modified 13 hours ago. The community score is 0/71. A green banner at the top states "No security vendors and 1 sandbox flagged this file as malicious". Below this, a table titled "Security vendors' analysis" lists 16 vendors, all of whom report the file as "Undetected".

Vendor	Result
Acronis (Static ML)	Undetected
Alibaba	Undetected
ALYac	Undetected
Arcabit	Undetected
AVG	Undetected
Baidu	Undetected
BitDefenderTheta	Undetected
ClamAV	Undetected
CrowdStrike Falcon	Undetected
Cybereason	Undetected
Cyren	Undetected
AhnLab-V3	Undetected
AllCloud	Undetected
Antiy-AVL	Undetected
Avast	Undetected
Avira (no cloud)	Undetected
BitDefender	Undetected
Bkav Pro	Undetected
CMC	Undetected

Identifying if the Executables are obfuscated / packed

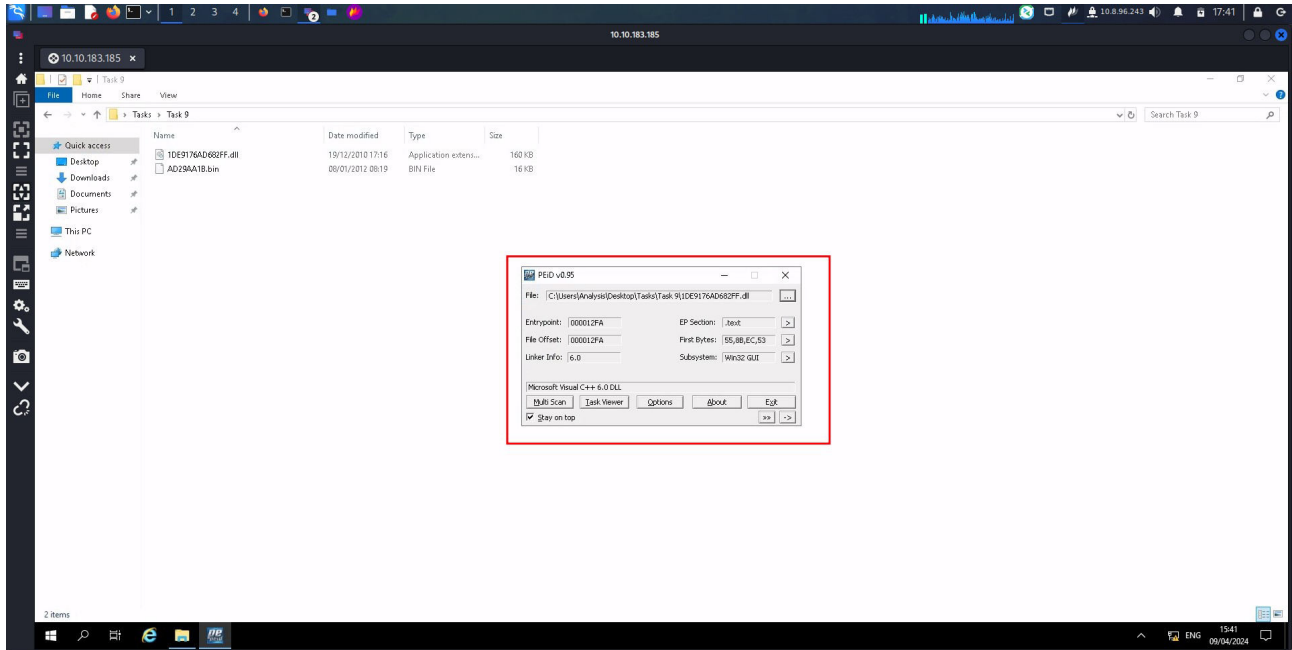
Just because a file doesn't have the ".exe" extension, doesn't mean it isn't an actual executable! For instance, it can have the ".jpg" extension and still be an executable piece of code.

The hex value for an executable is always "4D 5A". So if a file with a ".jpg" file has the hex header of "4D 5A", then it is obviously not a jpg file.

Answer the questions below

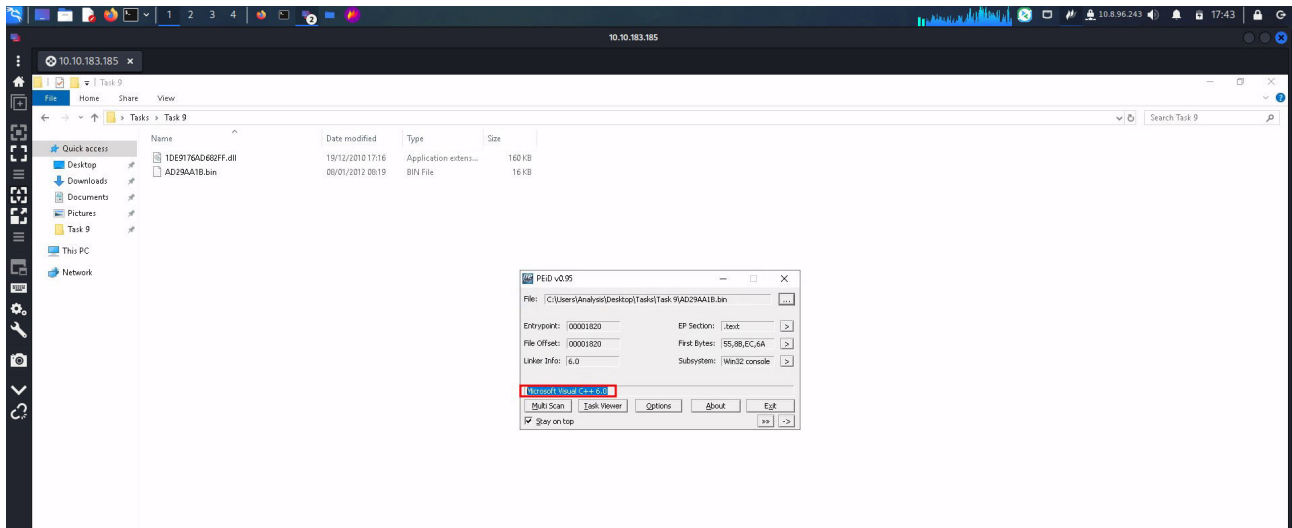
What does PeID propose 1DE9176AD682FF.dll being packed with?

Microsoft Visual C++ 6.0 DLL



What does PeID propose AD29AA1B.bin being packed with?

Microsoft Visual C++ 6.0



What is Obfuscation / Packing?

Packing is one form of obfuscation that malware Authors employ to prevent the analysis of programmes. There are both legitimate and malicious reasons as to why the Author of a program will want to prevent the decompiling of their program.

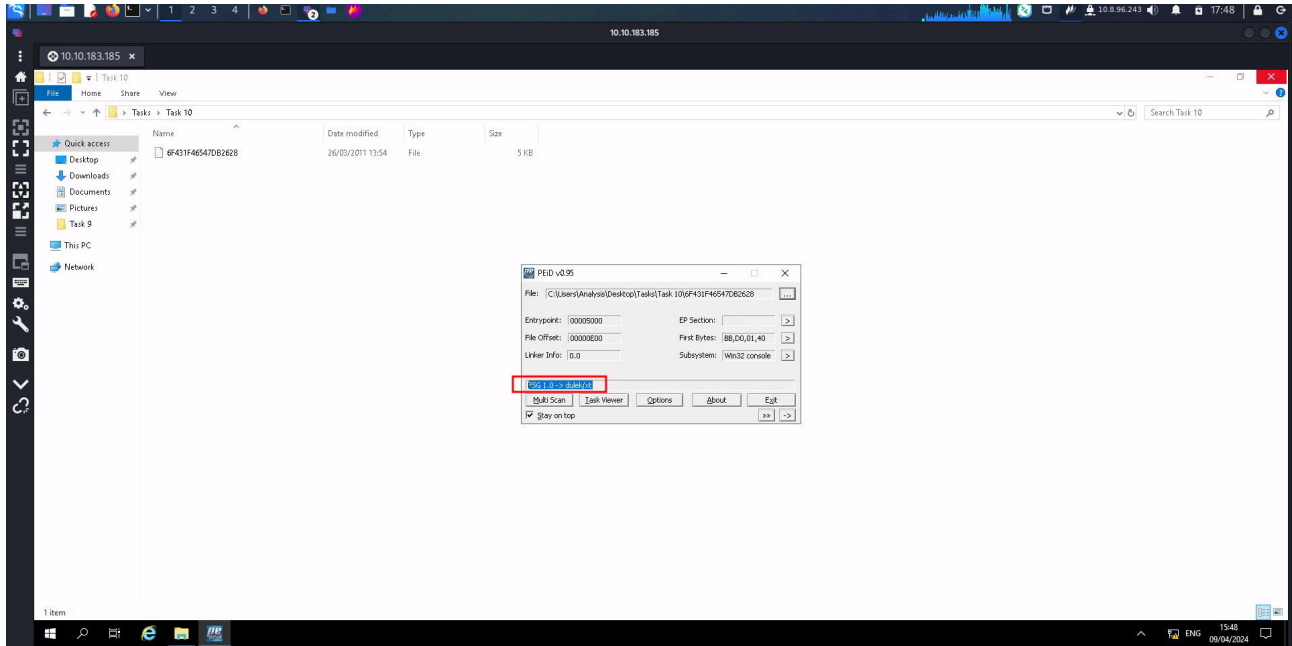
For example, a legitimate reason is the protection of intellectual property

malware Authors employ obfuscation techniques such as packing with the intent to prevent Ethical analysts reversing it to understand its behaviours and ultimately with the aims of achieving infection.

Answer the questions below

What packer does PeID report file "6F431F46547DB2628" to be packed with?

FSG 1.0 -> dulek/xt

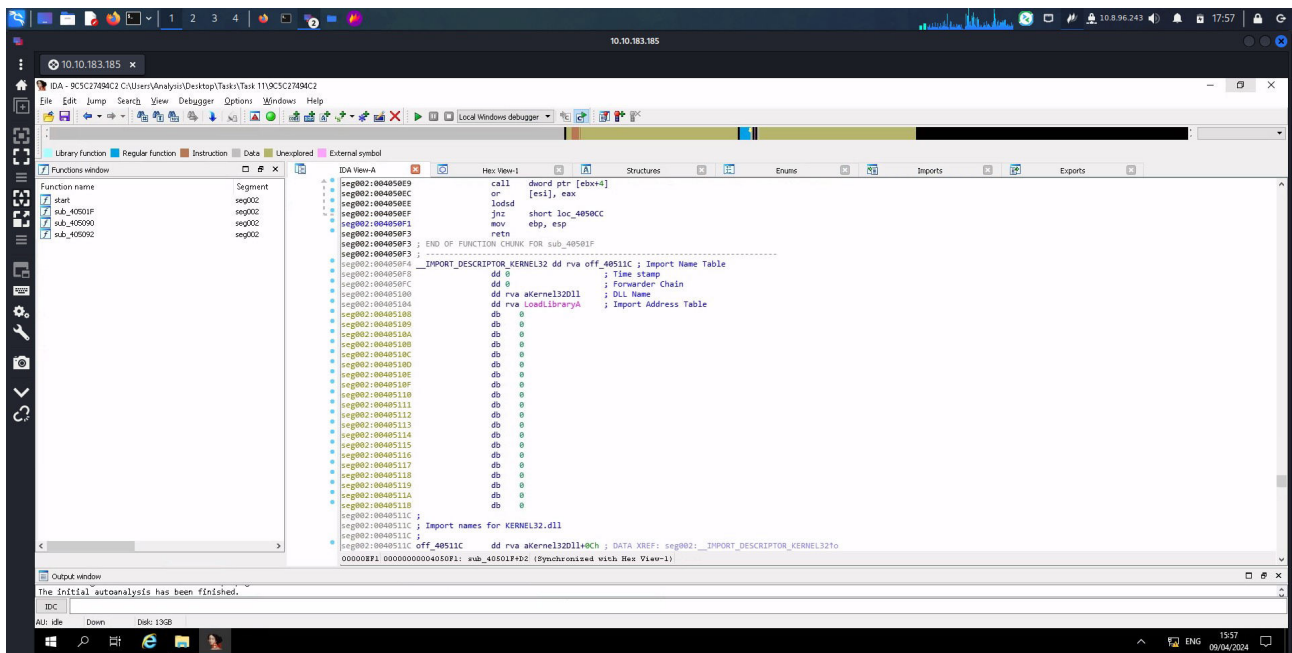


Visualising the Differences Between Packed & Non-Packed Code

PeID tool has a huge database, it doesn't have every packer out there! Especially say if an Author has written their own - PeID will have no way of identifying the packer used.

PeID is capable of detecting the possibility of packers being used, it is not able to automatically de-obfuscate them. This is a process we will have to do manually.

After confirming that this file is indeed packed, we can utilize a tool called IDA Freeware to disassemble and check for obfuscation.



Introduction to Strings

"Strings" are essentially the ASCII / Text contents of a program...this could be anything from passwords for self-extracting zips, to bitcoin addresses in ransomware samples.

when analysing the contents of these strings, we can sometimes paint a fairly indicative picture of the behaviours of the programme - bitcoin wallets being used in ransomware.

Task:

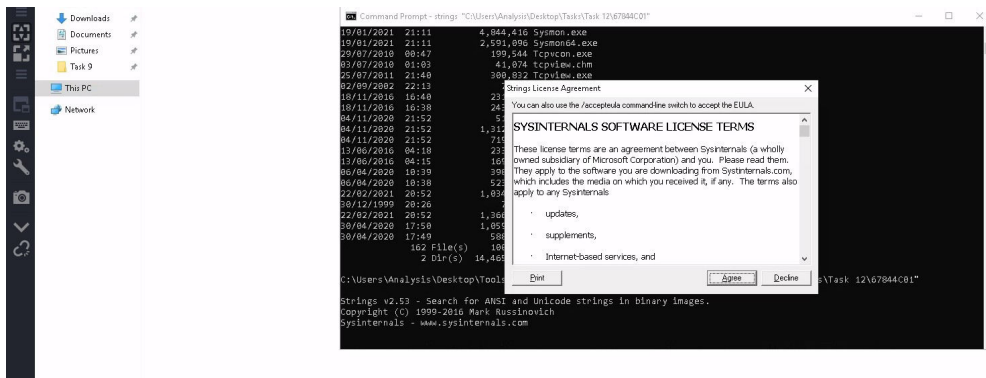
Open a Command prompt on the Windows Machine and navigate to the directory "Tools\SysinternalsSuite"

cd C:\Users\Analysis\Desktop\Tools\SysinternalsSuite

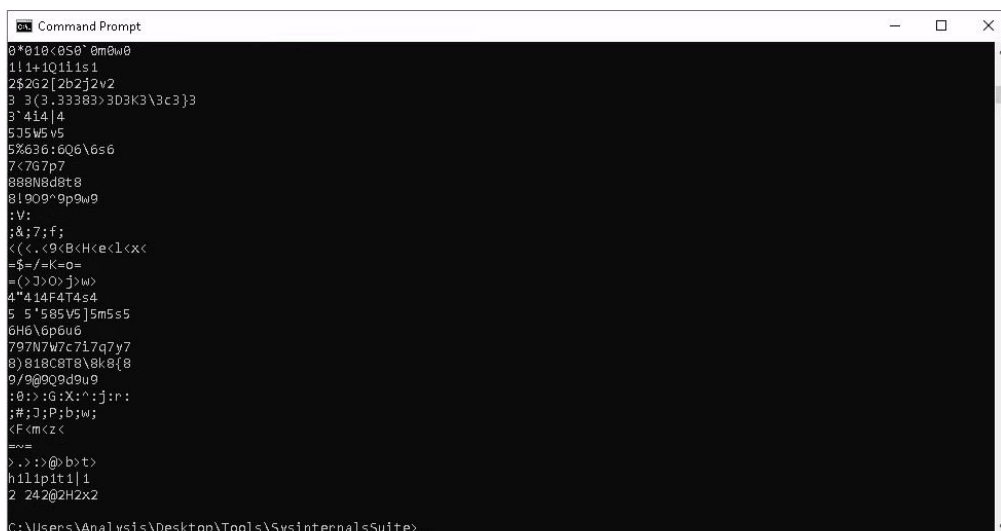
Keep this terminal open.

We're going to use Microsoft's Sysinternals "Strings" program to output the retained strings within the specified file in "Task 12". We can do this by:

strings "C:\Users\Analysis\Desktop\Tasks\Task 12\67844C01"



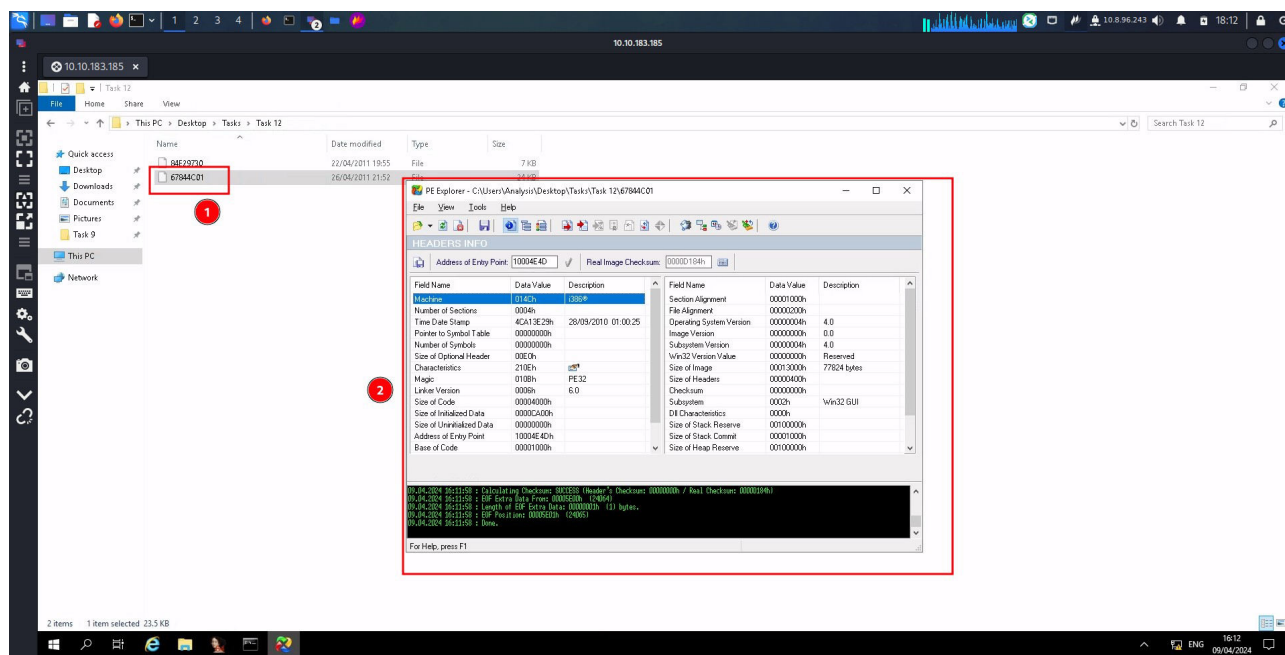
You will receive a whole load of text.



You'll find that programs often contain large amount of strings and using the "strings" tool from sysinternals may only display 10% of these...

It's not exactly practical scrolling up through a terminal for stuff like this. There's a GUI tool for that in windows.

Launch the application within "Tools/Static/PE Tools/PE Explorer" and drag and drop the same file "67844C01" from the previous question into the application.

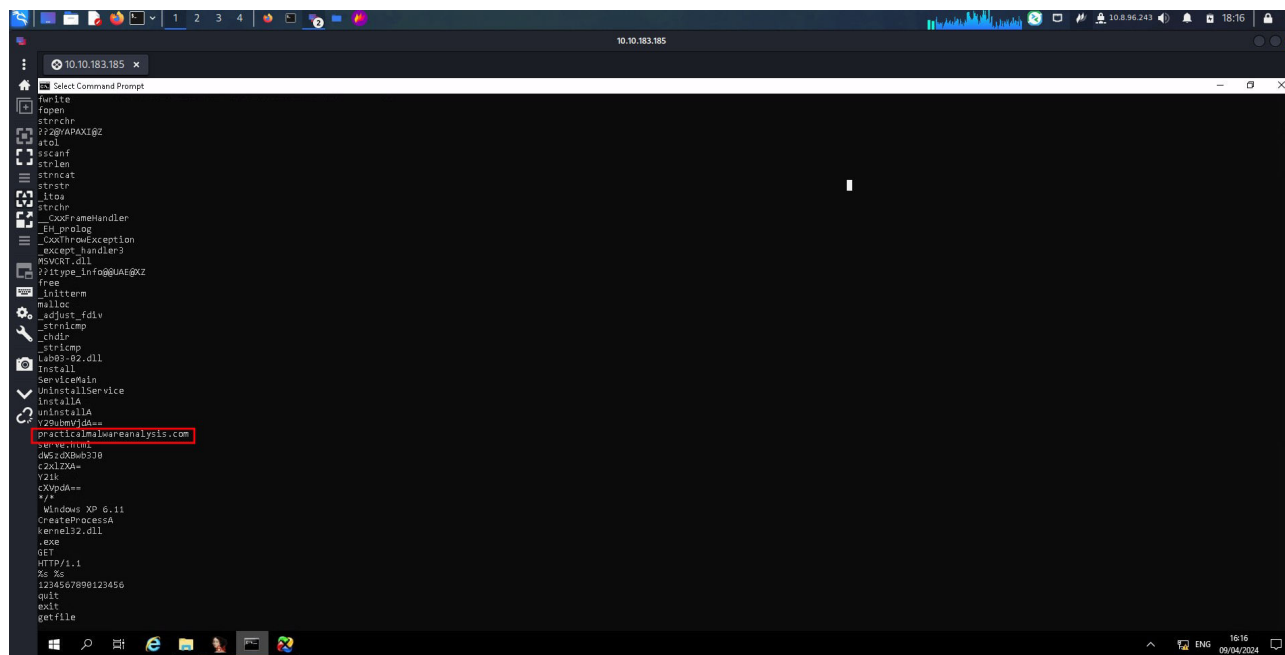


After import. Navigate to "View -> Imports"

Answer the questions below

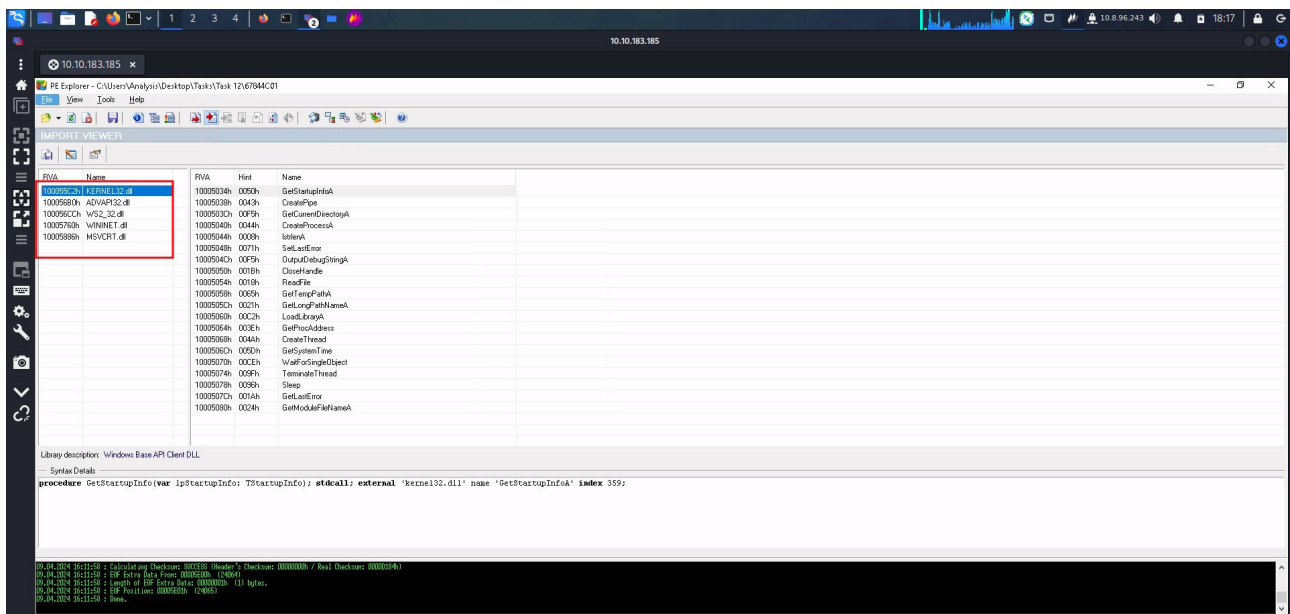
What is the URL that is outputted after using "strings"

practicalmalwareanalysis.com



How many unique "Imports" are there?

5



Introduction to Imports

The classification of IDA Freeware is arguable as the tool can be used for both static and dynamic analysis.

There are two classifications of tools like IDA Freeware:

1. Disassemblers
2. Debuggers

Disassemblers reverse the compiled code of a program from machine code to human-readable instructions (assembly). This is limited to how the program represents itself in its current state! I.e. If the contents of an executable changes during execution. Disassemblers will not reflect this.

Debuggers deploy the same techniques used by Disassemblers.

Debugger essentially facilitate execution of the program - where the analyser can view the changes made throughout each "step" of the program. These tools are great because a true picture of the program presents itself.

Practical:

For this room, we will be using IDA Freeware within the context of statistical analysis. I'll walkthrough how to import an executable into IDA Freeware below.

Navigate to the directory "Tasks/Task 13" and open "install.exe" with IDA Freeware, just like we did in the example above. Again, this may take a few seconds to a couple of minutes to compute dependant upon the size of the application. For this task expect roughly ~20 seconds.

1. Lets launch "IDA Freeware" and select the file to import, in this case we'll be using "uninstall.exe"
2. Since we know it is an executable file, we select "Portable executable for 80386 (PE) [pe64.dll]"
3. After pressing "OK" the application will load. Allow a few minutes for the executable to be decompiled.

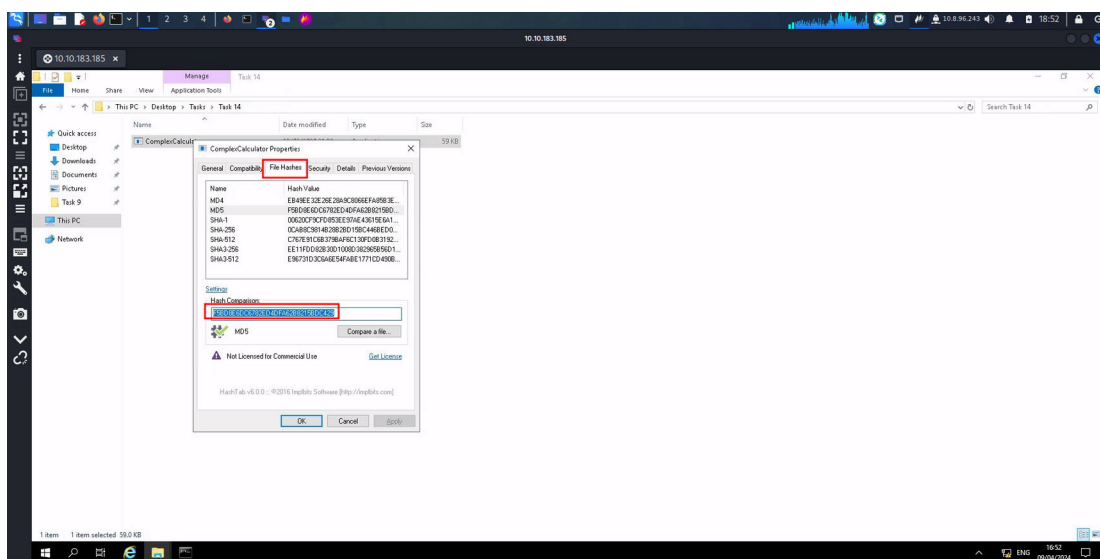
How many references are there to the library "msi" in the "Imports" tab of IDA Freeware for "install.exe" **9**



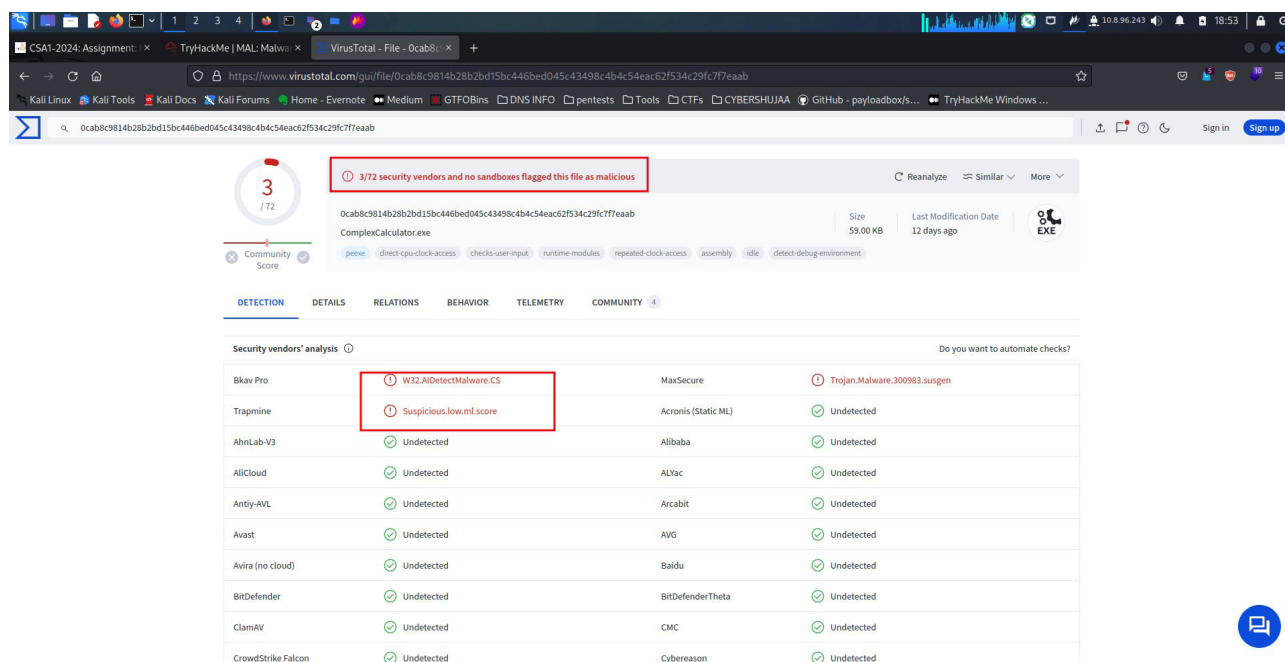
The file specified for analysis is "ComplexCalculator.exe" in the Directory "Tasks/Task 14". I'll leave it up to you to figure out what tool(s) out of what we've used above is best!

What is the MD5 Checksum of the file?

F5BD8E6DC6782ED4DFA62B8215BDC429



Does Virustotal report this file as malicious? **Yay**



The screenshot shows the VirusTotal web interface for a file named 'ComplexCalculator.exe' with SHA256 hash '0cab8c9814b28b2bd15bc446bed045c43498c4b4c54eac62f534c29fc77eaab'. The file is 59.00 KB and was last modified 12 days ago. A red box highlights the summary: '3/72 security vendors and no sandboxes flagged this file as malicious'. Below this, a table titled 'Security vendors' analysis' lists various antivirus engines and their results. Two red boxes highlight specific detections: 'W32.AIDetectMalware.CS' from MaxSecure and 'Suspicious.low.ml.score' from Trapsmine.

Security vendor	Result
Bkav Pro	W32.AIDetectMalware.CS
Trapsmine	Suspicious.low.ml.score
AhnLab-V3	Undetected
Allicloud	Undetected
Antiy-AVL	Undetected
Avast	Undetected
Avira (no cloud)	Undetected
BitDefender	Undetected
ClamAV	Undetected
CrowdStrike Falcon	Undetected
MaxSecure	Trojan.Malware.300983.sungen
Acronis (Static ML)	Undetected
Alibaba	Undetected
ALYac	Undetected
Arcabit	Undetected
AVG	Undetected
Baidu	Undetected
BitDefenderTheta	Undetected
CMC	Undetected
Cybereason	Undetected

Output the strings using Sysinternals "strings" tool.

What is the last string outputted?

d:h:

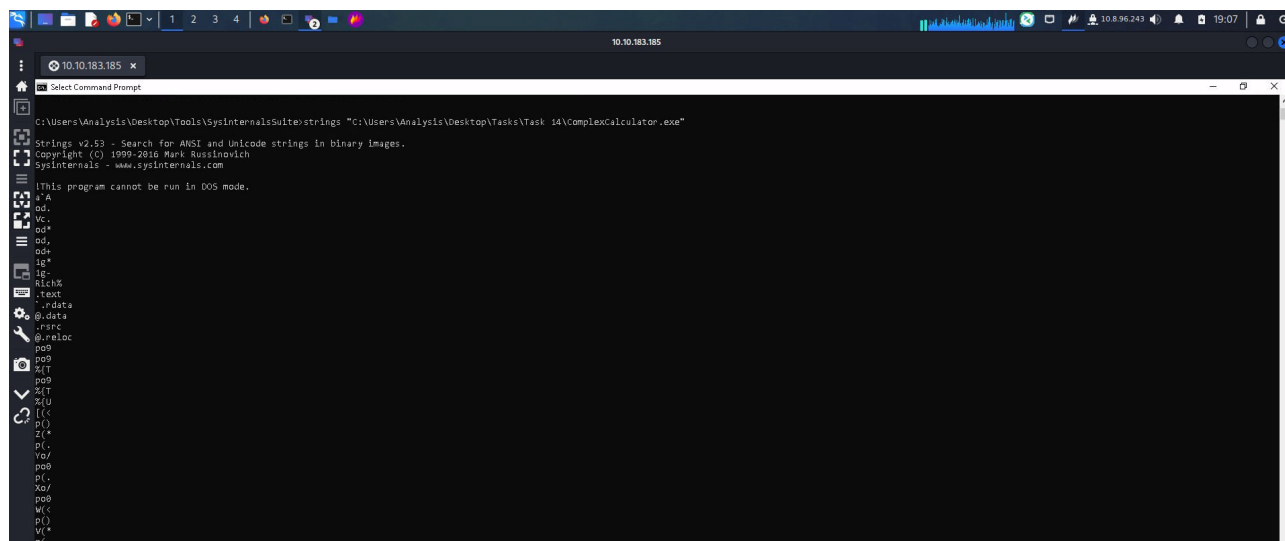
First open a Command prompt on the Windows Machine and navigate to the directory "Tools\SysinternalsSuite"

cd C:\Users\Analysis\Desktop\Tools\SysinternalsSuite

Use Microsoft's Sysinternals "Strings" program to output the retained strings within the specified file in "Task 14" (ComplexCalculator.exe).

Command Used:

strings "C:\Users\Analysis\Desktop\Tasks\Task 14\ComplexCalculator.exe"



The screenshot shows a Windows Command Prompt window with the command `strings "C:\Users\Analysis\Desktop\Tasks\Task 14\ComplexCalculator.exe"` executed. The output displays various strings found in the file, including file paths, version information, and system-related strings. The last string outputted is `d:h:`.

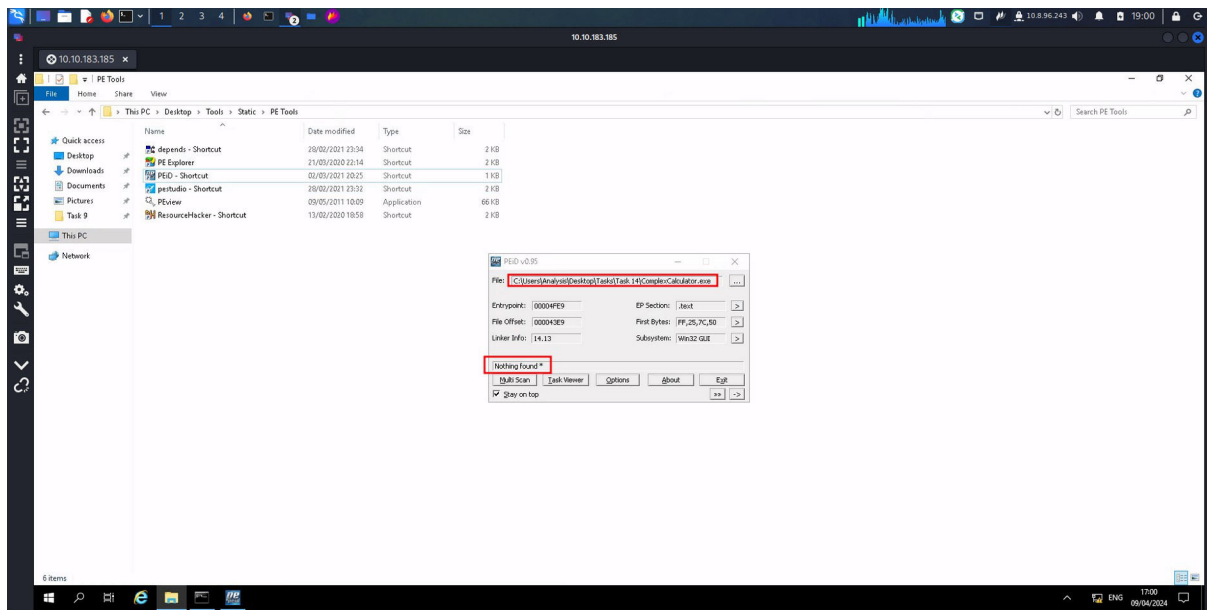
```
C:\Users\Analysis\Desktop\Tools\SysinternalsSuite>strings "C:\Users\Analysis\Desktop\Tasks\Task 14\ComplexCalculator.exe"

Strings v2.53 - Search for ANSI and Unicode strings in binary images.
Copyright (c) 1999-2018 Mark Russinovich
sysinternals - www.sysinternals.com

{This program cannot be run in DOS mode.
a'A
od.
Vc.
od*
od.
od*
ig*
ig.
RichX
.txt
.rdata
@.data
.rsrc
@.reloc
po9
po9
%T
po9
%T
%U
%I
Z(*
pI.
Yd/
po9
pI.
Xo/
po9
M(<
pI)
V(*
pI.
```

What is the output of PeID when trying to detect what packer is used by the file?

Nothing found *



Conclusion

In my conclusion, the Malware Introductory room has provided me with a comprehensive introduction to the world of malware, highlighting various malware types, their behaviors and how to detect and mitigate them. By exploring this room, I have gained valuable insights into the inner workings of malware, enhancing my understanding in Cybersecurity threats and defense. The practical exercises offered in this room enabled me to apply my knowledge in the simulated windows environment, fostering my hands-on learning and skill development. All in all this room has served as an excellent starting point into deepening my understanding in malware and how they function.

Thank You.