REPUBLIC OF ALBANIA
NATIONAL CYBER SECURITY AUTHORITY
DIRECTORATE OF CYBER SECURITY ANALYSIS

Technical Analysis of the Malicious File

Dokumenti përmban përmbajtje që shkel të drejtat e autorit

Version: 1.0
Date: 29/09/2025

TABLE OF CONTENTS

LIST OF FIGURES

This report has limitations and should be interpreted with caution!

Some of these limitations include:

Phase One:

*Information Sources:*This report relies on the data that was accessible at the time it was compiled. As a result, some elements might no longer reflect the current situation or may have changed since then.

Phase Two:

*Analysis Details:*Due to limited resources, certain aspects of the malicious file may not have been examined in depth. Any additional unknown information could lead to revisions or changes in the report.

Phase Three:

*Information Security:*To protect sources and confidential data, certain details may have been omitted or intentionally limited in this report. This decision was made to preserve the integrity and security of the information used.

AKSK reserves the right to modify, update, or change any part of this report without prior notice.

This report is not a final document.

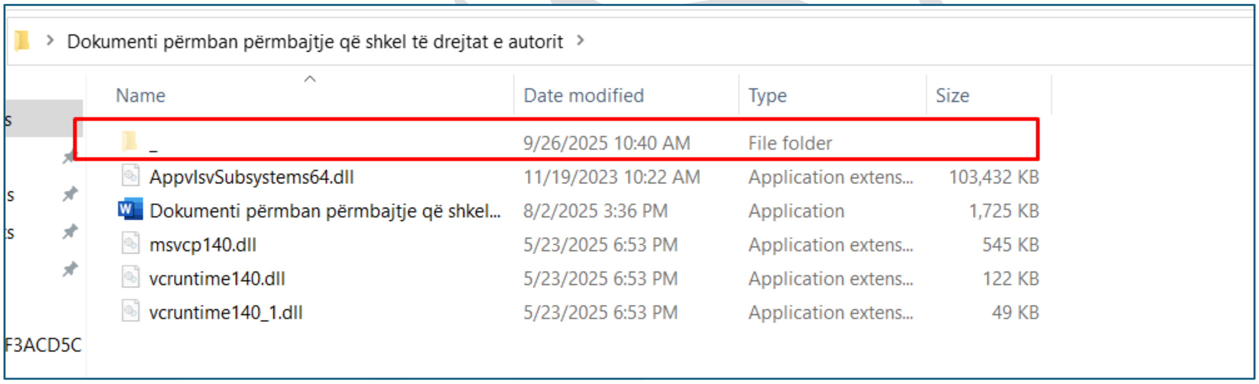
The findings are based on information available at the time of investigation and analysis. There is no guarantee regarding possible changes or updates to the reported information in the future. The authors of this report do not take responsibility for any misuse or consequences resulting from decisions made based on this report.

Technical Information

A phishing campaign has been identified targeting critical and key infrastructures of the Republic of Albania. The phishing emails contain an attachment named **" The document contains content that violates copyright.zip "**. This malicious file is designed to enable remote control by threat actors over the victims computers or systems, posing a serious cybersecurity risk.

Dokumenti përmban përmbajtje që shkel të drejtat e autorit

Analysis of this file begins with extraction from the .zip (archived) format. The first highlighted file is "The document contains content that violates copyright.exe," which is a ***PE (Portable Executable)*** type file, an executable file. Additionally, if the "View hidden items" option is enabled in Windows, several other hidden files are also detected

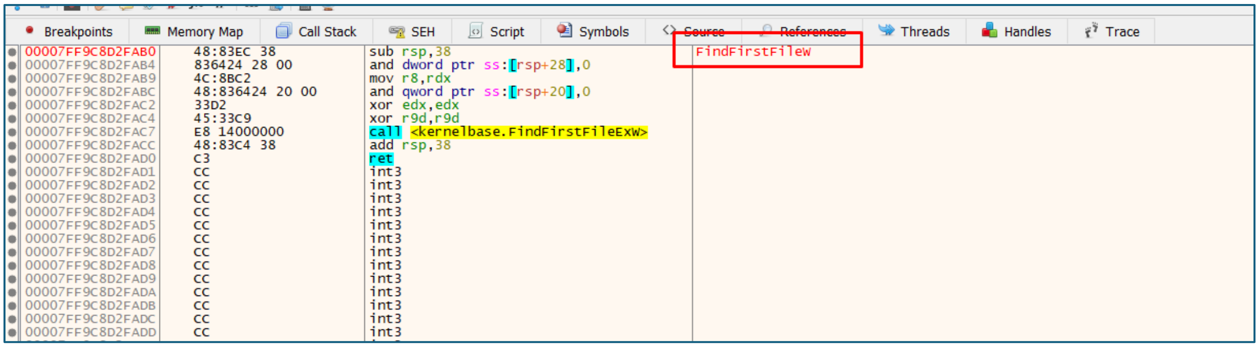


Dokumenti përmban përmbajtje që shkel të drejtat e autorit

Name	Date modified	Type	Size
_	9/26/2025 10:40 AM	File folder	
AppvIsvSubsystems64.dll	11/19/2023 10:22 AM	Application extens...	103,432 KB
Dokumenti përmban përmbajtje që shkel...	8/2/2025 3:36 PM	Application	1,725 KB
msvcp140.dll	5/23/2025 6:53 PM	Application extens...	545 KB
vcruntime140.dll	5/23/2025 6:53 PM	Application extens...	122 KB
vcruntime140_1.dll	5/23/2025 6:53 PM	Application extens...	49 KB

Figure 1. Hidden files

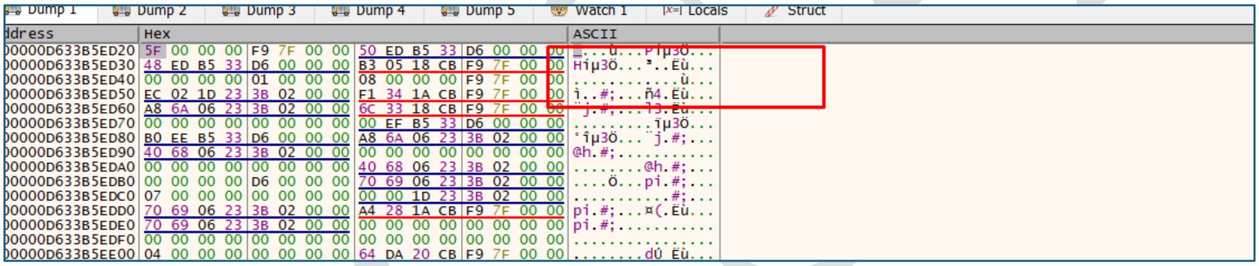
The most important file in the infection chain is also identified the dynamic link library **“AppvIsvSubsystems64.dll”,** which has a size of approximately 103 MB, an unusually large element. During static analysis, functions responsible for directory checks are identified, indicating that this **dll** file begins to search for the locations of various files. However, to determine exactly which function is being called, ***the debugging process*** is carried out.



```
00007FF9C8D2FAB0    48:83EC 38          sub rsp,38
00007FF9C8D2FAB4    836424 28 00        and dword ptr ss:[rsp+28],0
00007FF9C8D2FAB9    4C:8BC2             mov r8,rdx
00007FF9C8D2FABC    48:836424 20 00     and qword ptr ss:[rsp+20],0
00007FF9C8D2FAC2    33D2                xor edx,edx
00007FF9C8D2FAC4    45:33C9             xor r9d,r9d
00007FF9C8D2FAC7    E8 14000000         call <kernelbase.FindFirstFileExW>
00007FF9C8D2FACC    48:83C4 38          add rsp,38
00007FF9C8D2FAD0    C3                  ret
00007FF9C8D2FAD1    CC                  int3
00007FF9C8D2FAD2    CC                  int3
00007FF9C8D2FAD3    CC                  int3
00007FF9C8D2FAD4    CC                  int3
00007FF9C8D2FAD5    CC                  int3
00007FF9C8D2FAD6    CC                  int3
00007FF9C8D2FAD7    CC                  int3
00007FF9C8D2FAD8    CC                  int3
00007FF9C8D2FAD9    CC                  int3
00007FF9C8D2FADA    CC                  int3
00007FF9C8D2FADB    CC                  int3
00007FF9C8D2FADC    CC                  int3
00007FF9C8D2FADD    CC                  int3
```

Figure 2. FindFirstFileW Function

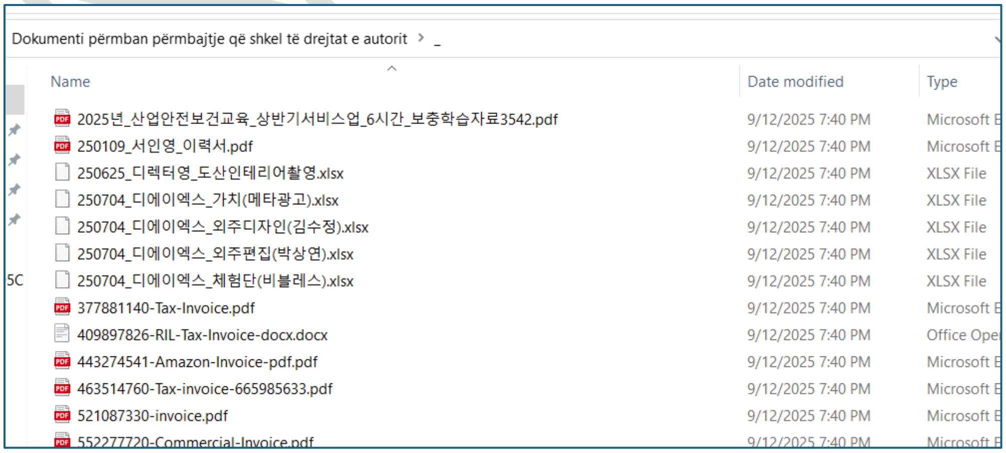
During debugging of the **dll file**, a check of the "_" folder is observed in the **RDX** register in the memory dump, from which it is evident that this directory contains a **payload** to continue the main purpose of the file



```
ddress            Hex                                              ASCII
00000D633B5ED20  5F 00 00 00 F9 7F 00 00 50 ED B5 33 D6 00 00 00  _...ù...Píµ3Ö...
00000D633B5ED30  48 ED B5 33 D6 00 00 00 B3 05 18 CB F9 7F 00 00  Híµ3Ö...³..Ëù...
00000D633B5ED40  00 00 00 00 01 00 00 00 08 00 00 00 F9 7F 00 00  ............ù...
00000D633B5ED50  EC 02 1D 23 3B 02 00 00 F1 34 1A CB F9 7F 00 00  ì..#;...ñ4.Ëù...
00000D633B5ED60  A8 6A 06 23 3B 02 00 00 6C 33 18 CB F9 7F 00 00  ¨j.#;...l3.Ëù...
00000D633B5ED70  00 00 00 00 00 00 00 00 00 EF B5 33 D6 00 00 00  .........ïµ3Ö...
00000D633B5ED80  B0 EE B5 33 D6 00 00 00 A8 6A 06 23 3B 02 00 00  °îµ3Ö...¨j.#;...
00000D633B5ED90  40 68 06 23 3B 02 00 00 00 00 00 00 00 00 00 00  @h.#;...........
00000D633B5EDA0  00 00 00 00 00 00 00 00 40 68 06 23 3B 02 00 00  ........@h.#;...
00000D633B5EDB0  00 00 00 00 D6 00 00 00 70 69 06 23 3B 02 00 00  ....Ö...pi.#;...
00000D633B5EDC0  07 00 00 00 00 00 00 00 00 00 1D 23 3B 02 00 00  ...........#;...
00000D633B5EDD0  70 69 06 23 3B 02 00 00 A4 28 1A CB F9 7F 00 00  pi.#;...¤(.Ëù...
00000D633B5EDE0  70 69 06 23 3B 02 00 00 00 00 00 00 00 00 00 00  pi.#;...........
00000D633B5EDF0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000D633B5EE00  04 00 00 00 00 00 00 00 64 DA 20 CB F9 7F 00 00  ........dÚ Ëù...
```

Figure 3. Inspection of the "_" directory

The "_" directory contains several files, some of which are legitimate PDFs, but there are other files that appear to be inaccessible yet may be decoded at a later stage and used for other purposes.



Dokumenti përmban përmbajtje që shkel të drejtat e autorit > _

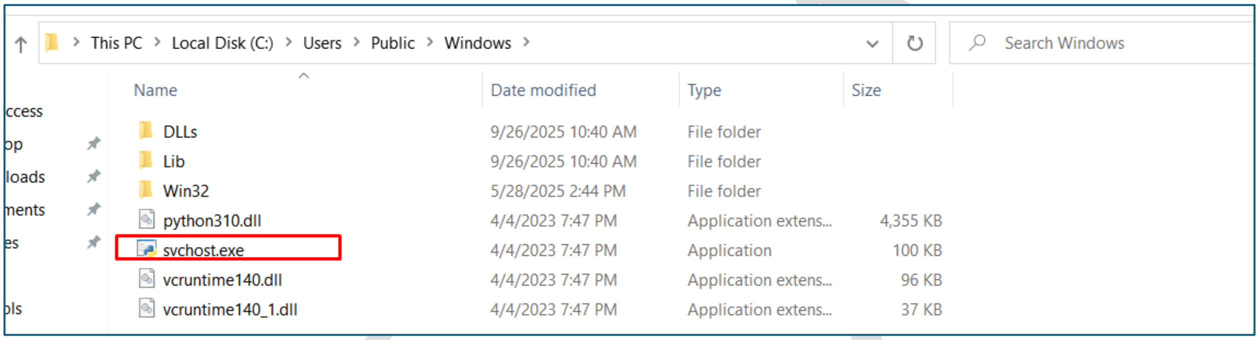
Name	Date modified	Type
2025년_산업안전보건교육_상반기서비스업_6시간_보충학습자료3542.pdf	9/12/2025 7:40 PM	Microsoft E
250109_서인영_이력서.pdf	9/12/2025 7:40 PM	Microsoft E
250625_디렉터영_도산인테리어촬영.xlsx	9/12/2025 7:40 PM	XLSX File
250704_디에이엑스_가치(메타광고).xlsx	9/12/2025 7:40 PM	XLSX File
250704_디에이엑스_외주디자인(김수정).xlsx	9/12/2025 7:40 PM	XLSX File
250704_디에이엑스_외주편집(박상연).xlsx	9/12/2025 7:40 PM	XLSX File
250704_디에이엑스_체험단(비블레스).xlsx	9/12/2025 7:40 PM	XLSX File
377881140-Tax-Invoice.pdf	9/12/2025 7:40 PM	Microsoft E
409897826-RIL-Tax-Invoice-docx.docx	9/12/2025 7:40 PM	Office Ope
443274541-Amazon-Invoice-pdf.pdf	9/12/2025 7:40 PM	Microsoft E
463514760-Tax-invoice-665985633.pdf	9/12/2025 7:40 PM	Microsoft E
521087330-invoice.pdf	9/12/2025 7:40 PM	Microsoft E
552277720-Commercial-Invoice.pdf	9/12/2025 7:40 PM	Microsoft E

Figure 4.Contents of the "_" directory

The main file in this directory is the file **Images.png**, which, despite its extension, is not an image but rather a **WinRAR archive application**. It is accessed via the command line using the command:

images.png x -ibck -y -paFr25vHl9vULPjJoV8rUcLS6YCzbMQ8k Invoice.pdf C:\\Users\\Public

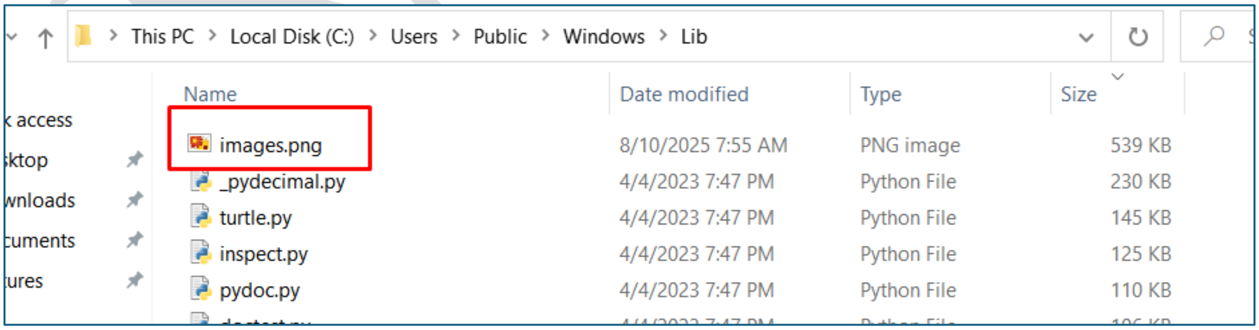
In this case, **WinRAR extracts** into the directory **C:\Users\Public** a folder which, in itself, contains the **Python library**.



Name	Date modified	Type	Size
DLLs	9/26/2025 10:40 AM	File folder	
Lib	9/26/2025 10:40 AM	File folder	
Win32	5/28/2025 2:44 PM	File folder	
python310.dll	4/4/2023 7:47 PM	Application extens...	4,355 KB
svchost.exe	4/4/2023 7:47 PM	Application	100 KB
vcruntime140.dll	4/4/2023 7:47 PM	Application extens...	96 KB
vcruntime140_1.dll	4/4/2023 7:47 PM	Application extens...	37 KB

Figure 5. Python.exe spoofed as svchost.exe

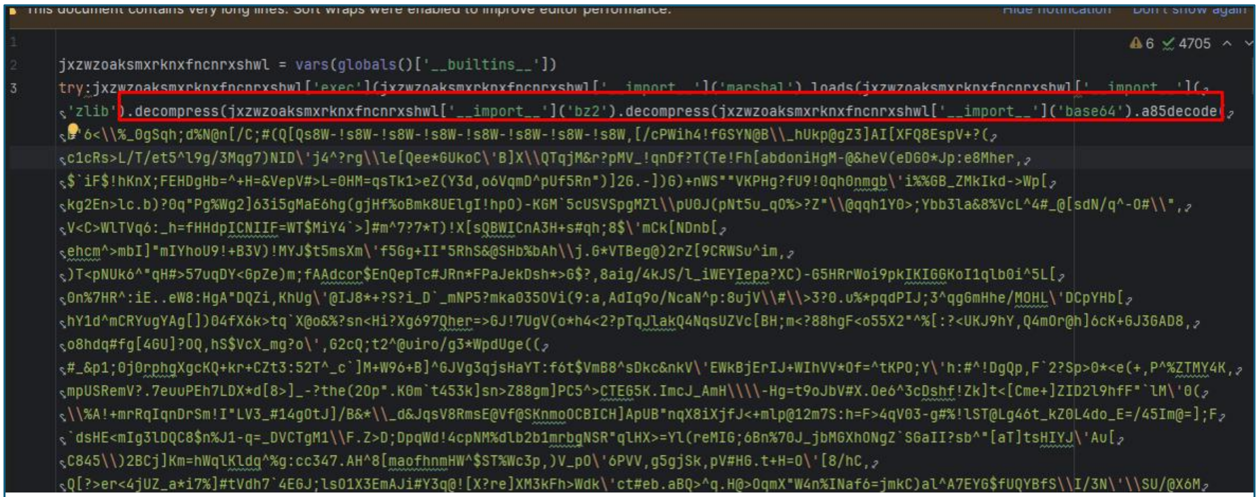
Svchost.exe or python.exe in this case is not malicious but considering the logic of how this malicious file has operated so far, python must be receiving some parameter in order to proceed to its final stage. During the investigation in ***C:\Users\Public\Windows\Lib,*** another suspicious file was identified named **images.png**, which appears to be repeated as in the previous case.



Name	Date modified	Type	Size
images.png	8/10/2025 7:55 AM	PNG image	539 KB
_pydecimal.py	4/4/2023 7:47 PM	Python File	230 KB
turtle.py	4/4/2023 7:47 PM	Python File	145 KB
inspect.py	4/4/2023 7:47 PM	Python File	125 KB
pydoc.py	4/4/2023 7:47 PM	Python File	110 KB

Figure 6. images.png phase 2.

This file is of type ***.py*** despite its extension. From the analysis in a regular text editor, its code is extracted as follows:



This document contains very long lines. Soft wraps were enabled to improve editor performance.

```
jxzwzoaksmxrknxfncnrxshwl = vars(globals()['__builtins__'])
try:jxzwzoaksmxrknxfncnrxshwl['exec'](jxzwzoaksmxrknxfncnrxshwl['__import__']('marshal').loads(jxzwzoaksmxrknxfncnrxshwl['__import__'](
'zlib').decompress(jxzwzoaksmxrknxfncnrxshwl['__import__']('bz2').decompress(jxzwzoaksmxrknxfncnrxshwl['__import__']('base64').a85decode(
b'6<\\%_0gSqh;d%N@n[/C;#(Q[Qs8W-!s8W-!s8W-!s8W-!s8W-!s8W-!s8W,[/cPWih4!fGSYN@B\\_hUkp@gZ3]AI[XFQ8EspV+?(
c1cRs>L/T/et5^l9g/3Mqg7)NID\'j4^?rg\\le[Qee*GUkoC\'B]X\\QTqjM&r?pMV_!qnDf?T(Te!Fh[abdoniHgM-@&heV(eDG0*Jp:e8Mher,
$`iF$!hKnX;FEHDgHb=^+H=&VepV#>L=0HM=qsTk1>eZ(Y3d,o6VqmD^pUf5Rn")]2G.-])G)+nWS""VKPHg?fU9!0qh0nmgb\'i%%GB_ZMkIkd->Wp[
kg2En>lc.b)?0q"Pg%Wg2]63i5gMaE6hg(gjHf%oBmk8UElgI!hp0)-KGM`5cUSVSpgMZl\\pU0J(pNt5u_q0%>?Z"\\@qqh1Y0>;Ybb3la&8%VcL^4#_@[sdN/q^-0#\\",
V<C>WLTVq6:_h=fHHdpICNIIF=WT$MiY4`>]#m^7?7*T)!X[sQBWICnA3H+s#qh;8$\'mCk[NDnb[
ehcm^>mbI]"mIYhoU9!+B3V)!MYJ$t5msXm\'f5Gg+II"5RhS&@SHb%bAh\\j.G*VTBeg@)2rZ[9CRWSu^im,
)T<pNUk6^"qH#>57uqDY<GpZe)m;fAAdcor$EnQepTc#JRn*FPaJekDsh*>G$?,8aig/4kJS/l_iWEYIepa?XC)-G5HRrWoi9pkIKIGGKoI1qlb0i^5L[
0n%7HR^:iE..eW8:HgA"DQZi,KhUg\'@IJ8*+?S?i_D`_mNP5?mka035OVi(9:a,AdIq9o/NcaN^p:8ujV\\#\\>3?0.u%*pqdPIJ;3^qgGmHhe/MOHL\'DCpYHb[
hY1d^mCRYugYAg[])04fX6k>tq`X@o&%?sn<Hi?Xg697Qher=>GJ!7UgV(o*h4<2?pTqJlakQ4NqsUZVc[BH;m<?88hgF<o55X2"^%[:?<UKJ9hY,Q4mOr@h]6cK+GJ3GAD8,
o8hdq#fg[4GU]?OQ,hS$VcX_mg?o\',G2cQ;t2^@uiro/g3*WpdUge((
#_&p1;0j0rphgXgcKQ+kr+CZt3:52T^_c`]M+W96+B]^GJVg3qjsHaYT:f6t$VmB8^sDkc&nkV\'EWkBjErIJ+WIhVV*Of=^tKPO;Y\'h:#^!DgQp,F`2?Sp>0*<e(+,P^%ZTMY4K,
mpUSRemV?.7euuPEh7LDX*d[8>]_-?the(2Op".K0m`t453k]sn>Z88gm]PC5^>CTEG5K.ImcJ_AmH\\\\-Hg=t9oJbV#X.0e6^3cDshf!Zk]t<[Cme+]ZID2l9hfF"`lM\'0(
\\%A!+mrRqIqnDrSm!I"LV3_#14g0tJ]/B&*\\_d&JqsV8RmsE@Vf@SKnmoOCBICH]ApUB"nqX8iXjfJ<+mlp@12m7S:h=F>4qV03-g#%!lST@Lg46t_kZ0L4do_E=/45Im@=];F
`dsHE<mIg3lDQC8$n%J1-q=_DVCTgM1\\F.Z>D;DpqWd!4cpNM%dlb2b1mrbgNSR"qlHX>=Yl(reMIG;6Bn%70J_jbMGXhONgZ`SGaII?sb^"[aT]tsHIYJ\'Au[
C845\\)2BCj]Km=hWqlKldq^%g:cc347.AH^8[maofhnmHW^$ST%Wc3p,)V_p0\'6PVV,g5gjSk,pV#HG.t+H=0\'[8/hC,
Q[?>er<4jUZ_a*i7%]#tVdh7`4EGJ;ls01X3EmAJi#Y3q@![X?re]XM3kFh>Wdk\'ct#eb.aBQ>^q.H@>0qmX"W4n%INaf6=jmkC)al^A7EYG$fUQYBfS\\I/3N\'\\SU/@X6M
```

Figure 7.Code of images.png

This portion of the code is hidden and used to perform the real purpose of the file. Next, a variable named xzwzoaksmxrknxfncnrxshwl is created and used to call Python's core functions.
Inside **the try** block the following main functions are observed:

__import__('base64').a85decode(...) where:

- It imports the base64 library and uses the a85decode function.
- a85decode is used to decode text encoded in Ascii85/Base85 format.

The result of this decoding is passed to **__import__('bz2').decompress(...):**

- So the decoded data is then processed through **BZ2** decompression **(Bzip2 algorithm).**
- Then the result is passed to **__import__('zlib').decompress(...):**

  Here it is decompressed again using the zlib algorithm.

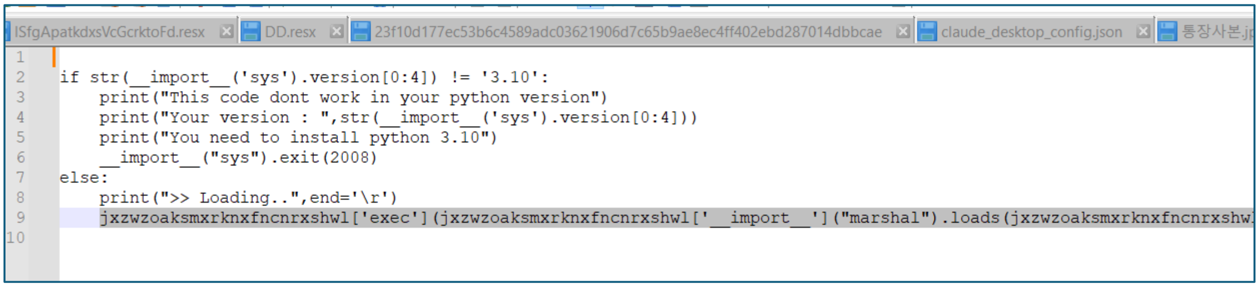
- **marshal.loads(...):**

marshal is used to deserialize Python objects. Here the result of the decompressions is expected to be a serialized code object (for example a .pyc-like object).

- **exec(...):**

Finally, exec runs the created object (or the marshaled content). Thus the hidden code will execute in the current environment.

If we modify the code step by step we can understand its behavior for each specific function


ISfgApatkdxsVcGrcktoFd.resx
DD.resx
23f10d177ec53b6c4589adc03621906d7c65b9ae8ec4ff402ebd287014dbbcae
claude_desktop_config.json
통장사본.jp

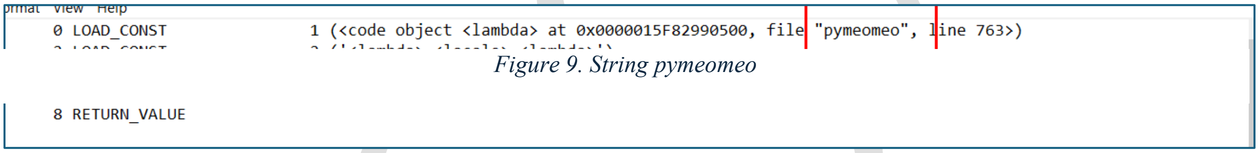
```

if str(__import__('sys').version[0:4]) != '3.10':
    print("This code dont work in your python version")
    print("Your version : ",str(__import__('sys').version[0:4]))
    print("You need to install python 3.10")
    __import__("sys").exit(2008)
else:
    print(">> Loading..",end='\r')
    jxzwzoaksmxrknxfncnrxshwl['exec'](jxzwzoaksmxrknxfncnrxshwl['__import__']("marshal").loads(jxzwzoaksmxrknxfncnrxshwl
```

Figure 8. Decoding the first phase of the script

What makes it interesting is the ***if*** logical condition which indicates that the version required for this file to run is **3.10**.

In the ***else*** branch the same logic continues as before, and again if the code is modified once it reaches the final phase it disassembles **the .pyc object and accesses the real code**.



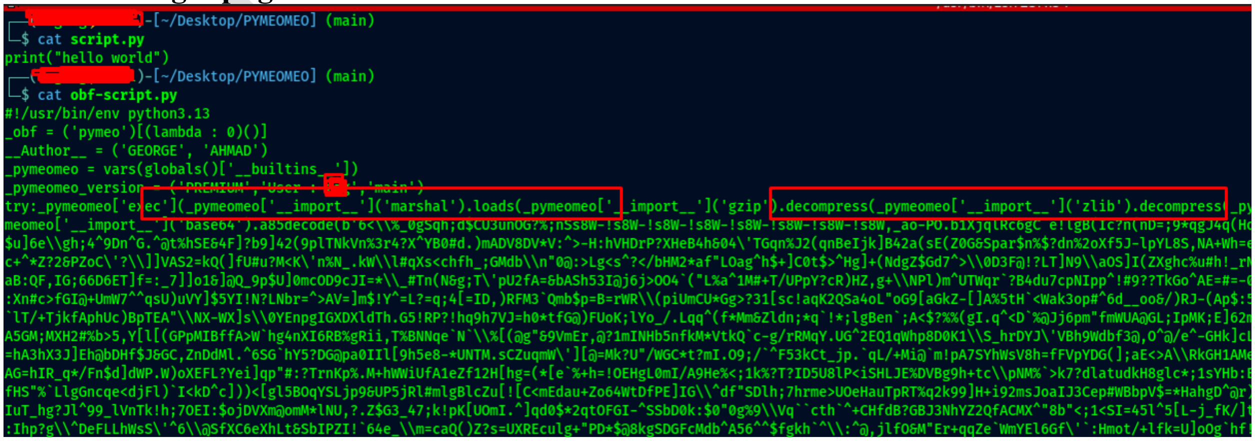
```
   0 LOAD_CONST               1 (<code object <lambda> at 0x0000015F82990500, file "pymeomeo", line 763>)

   8 RETURN_VALUE
```

Figure 9. String pymeomeo

During the disassembly of the .pyc objects, the character string **pymeomeo** is detected, which, if global information about it is searched, is understood to be used for hiding the code: **https://github.com/zrsx/PYMEOMEO**

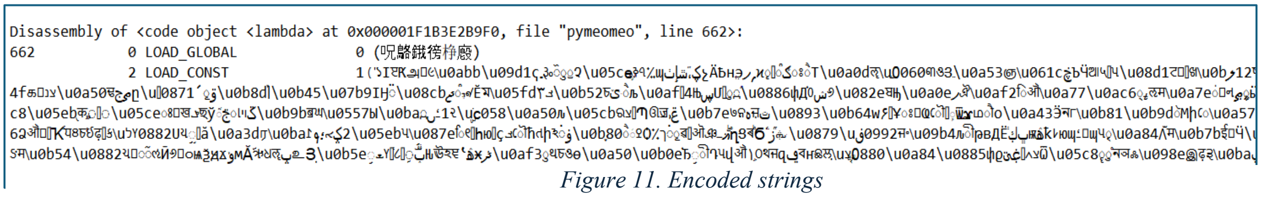
"**Advanced Python Obfuscation and Protection Suite".**This is also confirmed by the files found on GitHub, where the code obfuscation depends on the installed Python version—something that was previously observed during the analysis. If a simple Python file is created with the text **'Hello World'** and is obfuscated using this project, the resulting code will be identical to the Python code found in **images.png.**



```
┌──[redacted]-[~/Desktop/PYMEOMEO] (main)
└─$ cat script.py
print("hello world")
┌──([redacted])-[~/Desktop/PYMEOMEO] (main)
└─$ cat obf-script.py
#!/usr/bin/env python3.13
_obf = ('pymeo')[(lambda : 0)()]
__Author__ = ('GEORGE', 'AHMAD')
_pymeomeo = vars(globals()['__builtins__'])
_pymeomeo_version = ('PREMIUM', 'User : [redacted]', 'main')
try:_pymeomeo['exec'](_pymeomeo['__import__']('marshal').loads(_pymeomeo['__import__']('gzip').decompress(_pymeomeo['__import__']('zlib').decompress(_py
meomeo['__import__']('base64').a85decode(b'6<\\%_0gSqh;d$CU3unOG?%;nSs8W-!s8W-!s8W-!s8W-!s8W-!s8W-!s8W,_ao-PO.b1XjqlRc6gC`e!lgB(Ic?n(nD=;9*qgJ4q(Hc
$u]6e\\gh;4^9Dn^G.^@t%hSE&4F]?b9]42(9plTNkVn%3r4?X^YB0#d.)mADV8DV*V:^>-H:hVHDrP?XHeB4h&04\'TGqn%J2(qnBeIjk]B42a(sE(Z0G&Spar$n%$?dn%2oXf5J-lpYL8S,NA+Wh=e
c+^*Z?2&PZoC\'?\\]]VAS2=kQ(]fU#u?M<K\'n%N_.kW\\l#qXs<chfh_;GMdb\\n"0@:>Lg<s^?</bHM2*af"LOag^h$+]C0t$>^Hg]+(NdgZ$Gd7^>\\0D3F@!?LT]N9\\aOS]I(ZXghc%u#h!_rN
aB:QF,IG;66D6ET]f=:_7]]o1&]@Q_9p$U]0mcOD9cJI=*\\_#Tn(N&g;T\'pU2fA=&bASh53I@j6j>OO4`("L%a^1M#+T/UPpY?cR)HZ,g+\\NPl)m^UTWqr`?B4du7cpNIpp^!#9??TkGo^AE=#=-G
:Xn#c>fGI@+UmW7^^qsU)uVY]$5YI!N?LNbr=^>AV=]m$!Y^=L?=q;4[=ID,)RFM3`Qmb$p=B=rWR\\(piUmCU*Gg>?31[sc!aqK2QSa4oL"oG9[aGkZ-[]A%5tH`<Wak3op#^6d__oo&/)RJ-(Ap$:5
`lT/+TjkfAphUc)BpTEA"\\NX-WX]s\\0YEnpgIGXDXldTh.G5!RP?!hq9h7VJ=h0*tfG@)FUoK;lYo_/.Lqq^(f*Mm&Zldn;*q`!*;lgBen`;A<$?%%(gI.q^<D`%@Jj6pm"fmWUA@GL;IpMK;E]62m
A5GM;MXH2#%b>5,Y[l[(GPpMIBffA>W`hg4nXI6RB%gRii,T%BNNqe`N`\\%[(@g"&9VmEr,@?1mINHb5nfkM*VtkQ`c-g/rRMqY.UG^2EQ1qWhp8D0K1\\S_hrDYJ\'VBh9Wdbf3@,O^@/e^-GHk]cL
=hA3hX3J]Eh@bDHf$J&GC,ZnDdMl.^6SG`hY5?DG@pa0IIl[9h5e8-*UNTM.sCZuqmW\'][@=Mk?U"/WGC*t?mI.O9;/`^F53kCt_jp.`qL/+Mi@`m!pA7SYhWsV8h=fFVpYDG(];aE<>A\\RkGH1AMe
AG=hIR_q*/Fn$d]dWP.W)oXEFL?Yei]qp"#:?TrnKp%.M+hWWiUfA1eZf12H[hg=(*[e`%+h=!OEHgL0mI/A9He%<;1k%?T?ID5U8lP<iSHLJE%DVBg9h+tc\\pNM%`>k7?dlatudkH8glc*;1sYHb:E
fHS"%`LlgGncqe<djFl)`I<kD^c]))<[gl5BOqYSLjp9&UP5jRl#mlgBlcZu[![C<mEdau+Zo64WtDfPE]IG\\^df"SDlh;7hrme>UOeHauTpRT%q2k99]H+i92msJoaIJ3Cep#WBbpV$=*HahgD^@r)
IuT_hg?Jl^99_lVnTk!h;7OEI:$ojDVXm@omM*lNU,?.Z$G3_47;k!pK[UOmI.^]qd0$*2qtOFGI-^SSbD0k:$0"0g%9\\Vq``cth`^+CHfdB?GBJ3NhYZ2QfACMX^"8b"<;1<SI=45l^5[L-j_fK/]t
:Ihp?g\\^DeFLLhWsS\'^6\\@SfXC6eXhLt&SbIPZI!`64e_\\m=caQ()Z?s=UXREculg+"PD*$@8kgSDGFcMdb^A56^^$fgkh`^\\:^@,jlfO&M"Er+qqZe`WmYEl6Gf\'`:Hmot/+lfk=U]oOg`hf!
```

Figure 10. PYMEOMEO obfuscator

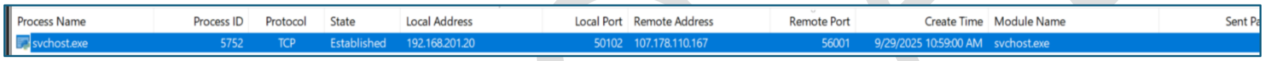
During the disassembly process, another layer of obfuscation is revealed, displaying characters in encoded strings.



```
Disassembly of <code object <lambda> at 0x000001F1B3E2B9F0, file "pymeomeo", line 662>:
662           0 LOAD_GLOBAL              0 (呪駱鋨徬梣廏)
              2 LOAD_CONST               1 ('ʖIঘKஅ□৫\u0abb\u09d1ςঃ০ুջ\u05ceূ৭%щشارت؛ڮ؏ÄѢН϶ر,ҡৄ্ঁѮুঁT\u0a0dন্\u0060ঞঙ3\u0a53௵\u061cڿѢӵআԿԿ\u08d1ट□ঙ\u0b,12
4fக□נצ\u0a50ছ্মج‌ը\u0871´ৃق\u0b8dl\u0b45\u07b9IӇ়\u08cbع়ৗ/ĔΆ\u05fdϒ܋\u0b52ঢ়ی়ՊՊ\u0af□4Ѣپ‌Ս□ৃ\u0886ф̇Д0ش\u082eঘӇ\u0a0eر̃\u0af2ীআ\u0a77\u0ac6ৃলম\u0a7eি□ণي
c8\u05ebক়□়\u05ce০৪□ঝ‌ৢচ্ছŷ়ঁ০।।ڭ\u0b9bষ্য\u0557ы\u0ba‌ِ؛1২\uc058\u0a50ՊՊ\u05cbৠষ□ٵৢঌজ়غ\u0b7e৩ঞৃল\u0893\u0b64wϟ□¥০ঃ□ɑ়া□ঢ়ঋ়o\u0a43ऀनΓ\u0b81\u0b9d়Мђ৫\u0a57
6Ωঔ□ΓKঘ৪চঔ্□§\u5Y0882Uप৾□ă\u0a3dդ\u0ba‌ٰوٕڪ2\u05ebԿ\u087eি৫□hю□ç܋়ীħժђ২়ؤ\u0b80়୨0%ղ়ৢ□ঔঞᄆ়ঢ়ঔ৪ঠσ̃‌ش\u0879\uف0992ন৹\u09b4ՊՊ়pвДЁڸپضk৶юԿ□ԿԿ্\u0a84ΛΆ\u0b7bঈ□Ӵ\u
ઙম\u0b54\u0882ধ□়়ঋӢ৩□০мӡԿxوмĂৠঘল্پ‌ڪـ3\u0b5e়ـY□ڭ□়ٿԿউ২২‘ڠxخرد\u0af3ঙঠঙ৪\u0a50\u0b0eҕ়ীԿԿԿঔ)০ঠমqیف্ব্মঞ্ল\u0880\u0a84\u0885փϱغ□ʌצω\u05c8ৃ্নঝ্ক\u098e@ড়ঽ\u0ba
```

Figure 11. Encoded strings

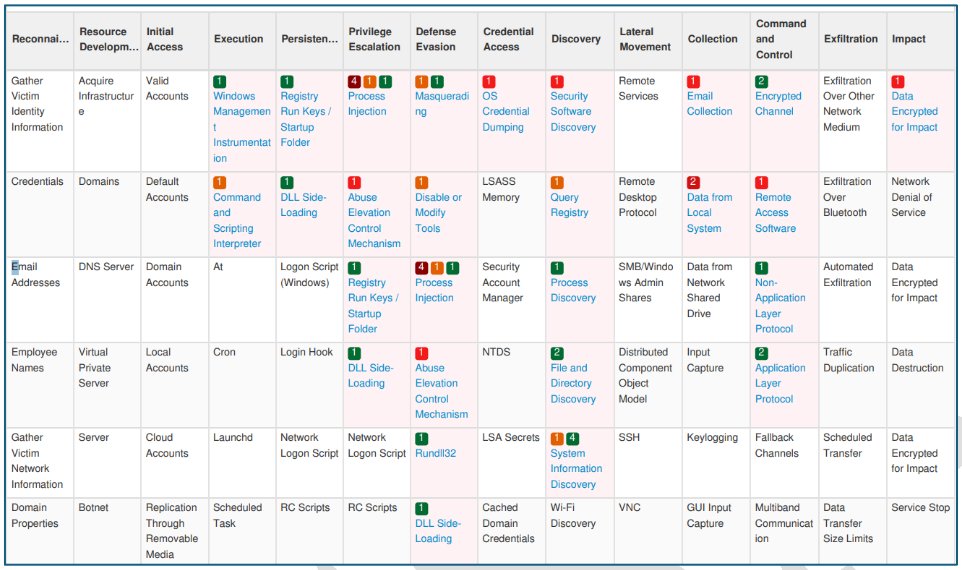
Given the very high level of obfuscation, an analysis is conducted to observe the dynamic activity performed by the malicious file when executed with the images.png parameter. This reveals that the malicious actors are able to perform remote command execution and, depending on their interests, carry out other illegitimate actions. At the end of the chain, this executable file establishes communication with a C2 server at IP: **107[.]178[.]110[.]167.**



Process Name	Process ID	Protocol	State	Local Address	Local Port	Remote Address	Remote Port	Create Time	Module Name	Sent Pa
svchost.exe	5752	TCP	Established	192.168.201.20	50102	107.178.110.167	56001	9/29/2025 10:59:00 AM	svchost.exe	

Figure 12. IP Command And Control

MITRE ATT&CK



Reconnai...	Resource Developm...	Initial Access	Execution	Persisten...	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Command and Control	Exfiltration	Impact
Gather Victim Identity Information	Acquire Infrastructure	Valid Accounts	1 Windows Management Instrumentation	1 Registry Run Keys / Startup Folder	4 1 1 Process Injection	1 1 Masquerading	1 OS Credential Dumping	1 Security Software Discovery	Remote Services	1 Email Collection	2 Encrypted Channel	Exfiltration Over Other Network Medium	1 Data Encrypted for Impact
Credentials	Domains	Default Accounts	1 Command and Scripting Interpreter	1 DLL Side-Loading	1 Abuse Elevation Control Mechanism	1 Disable or Modify Tools	LSASS Memory	1 Query Registry	Remote Desktop Protocol	2 Data from Local System	1 Remote Access Software	Exfiltration Over Bluetooth	Network Denial of Service
Email Addresses	DNS Server	Domain Accounts	At	Logon Script (Windows)	1 Registry Run Keys / Startup Folder	4 1 1 Process Injection	Security Account Manager	1 Process Discovery	SMB/Windows Admin Shares	Data from Network Shared Drive	1 Non-Application Layer Protocol	Automated Exfiltration	Data Encrypted for Impact
Employee Names	Virtual Private Server	Local Accounts	Cron	Login Hook	1 DLL Side-Loading	1 Abuse Elevation Control Mechanism	NTDS	2 File and Directory Discovery	Distributed Component Object Model	Input Capture	2 Application Layer Protocol	Traffic Duplication	Data Destruction
Gather Victim Network Information	Server	Cloud Accounts	Launchd	Network Logon Script	Network Logon Script	1 Rundll32	LSA Secrets	1 4 System Information Discovery	SSH	Keylogging	Fallback Channels	Scheduled Transfer	Data Encrypted for Impact
Domain Properties	Botnet	Replication Through Removable Media	Scheduled Task	RC Scripts	RC Scripts	1 DLL Side-Loading	Cached Domain Credentials	Wi-Fi Discovery	VNC	GUI Input Capture	Multiband Communication	Data Transfer Size Limits	Service Stop

Indicators of Compromise IoCs

Dokumenti përmban përmbajtje që shkel të drejtat e autorit.zip	**341BA8A556F4AC503AB23D9E5D2114261AFD24AED332F2E404705B522AFD5998**
AppvIsvSubsystems64.dll	**653F1B0F2B4C711B46016C268FB985D82528BB4240E202BE9640F31A0E6217B8**
Images.png	**A5B19195F61925EDE76254AAAD942E978464E93C7922ED6F064FAB5AAD901EFC**
C2	**107[.]178[.]110[.]167**

Recommendations

National Cyber Security Authority recommends:

- Immediate blocking of the Indicators of Compromise (IoCs) mentioned above on your protective devices.
- Continuous analysis of logs coming from the SIEM (Security Information and Event Management) system.
- Training of non-technical staff on phishing attacks and how to avoid infection from them.
- Installation of network perimeter devices that perform deep traffic analysis, relying not only on access control lists but also on traffic behavior (Next-Generation Firewalls).
- Segmentation of critical systems into different VLANs, applying access control lists across the entire network perimeter. Web services should be separated from their databases, and Active Directory should be placed in a separate VLAN.
- Implementation and use of LAPS (Local Administrator Password Solution) for Microsoft systems to manage local administrator passwords.
- Application of traffic filters in cases of remote access to hosts (employees/third parties/clients).
- Implementation of solutions that filter, monitor, and block malicious traffic between web applications and the internet, such as a Web Application Firewall (WAF).
- Behavior-based traffic analysis for endpoint devices, through the use of EDR/XDR solutions. This enables detection of malicious files not only by signature but also by behavior.
- Design and implementation of an Identity Access Management (IAM) solution to control user identities and privileges in real time, based on the “zero-trust” principle.