NetWire

Technical Analysis Report

ZAYOTEM
ZARARLI YAZILIM ÖNLEME VE TERSİNE MÜHENDİSLİK

Contents

Overview

The software, which belongs to the NetWire Family,is a RAT type malware that can track user movements such as secret credentials, keyboard strokes, and execute commands from a remote server. This threat spreads through MS Office Documents, download links in PDF content and compressed files containing payloads.

Some of the information obtained from infected devices include;

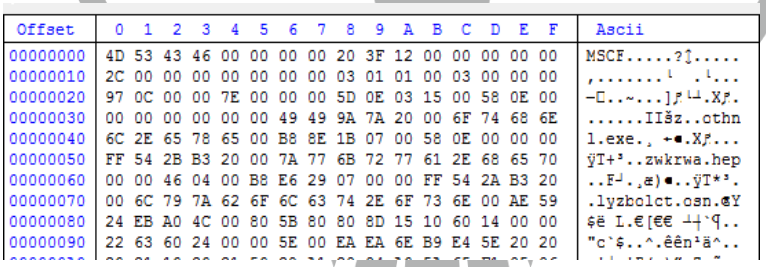
- Browser credentials,
- Keyboard strokes,
- Registery manipulation,
- Device properties and file information,
- Remote access

xox.exe Analysis

Name	xox.exe
MD5	5c9ad0440fefa31403bd944a1a10a3b8
SHA256	2b1245c4547eee5a4545431f1969ab4dd5ba8ac4d0d2dd758d3c77a250e6ddb8
File Type	PE32 / EXE

Static Analysis

The malicious file, creates its malicious activities by using files that come archived with **Microsoft Cabinet File** (MSCF). It performs operations such as permanence and info stealer with the operations it will perform when it is run.



Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	Ascii
00000000	4D	53	43	46	00	00	00	00	20	3F	12	00	00	00	00	00	MSCF.....?↕.....
00000010	2C	00	00	00	00	00	00	00	03	01	01	00	03	00	00	00	,.......ⁱ .ⁱ...
00000020	97	0C	00	00	7E	00	00	00	5D	0E	03	15	00	58	0E	00	–□..~...]♬┘┘.X♬.
00000030	00	00	00	00	00	00	49	49	9A	7A	20	00	6F	74	68	6E	......IIšz..othn
00000040	6C	2E	65	78	65	00	B8	8E	1B	07	00	58	0E	00	00	00	l.exe.¸ ←■.X♬...
00000050	FF	54	2B	B3	20	00	7A	77	6B	72	77	61	2E	68	65	70	ÿT+³..zwkrwa.hep
00000060	00	00	46	04	00	B8	E6	29	07	00	00	FF	54	2A	B3	20	..F┘.¸æ)■..ÿT*³.
00000070	00	6C	79	7A	62	6F	6C	63	74	2E	6F	73	6E	00	AE	59	.lyzbolct.osn.®Y
00000080	24	EB	A0	4C	00	80	5B	80	80	8D	15	10	60	14	00	00	$ë L.€[€€ ┴┼`¶..
00000090	22	63	60	24	00	00	5E	00	EA	EA	6E	B9	E4	5E	20	20	"c`$..^.êên¹ä^..

Image 1- MSCF and files to extract

MSCF, files with the ".cab" extension store data for various Windows installations. The applications to be extracted are clearly visible in the ".cab" (MSCF) file When run, it saves files to the targeted directory.

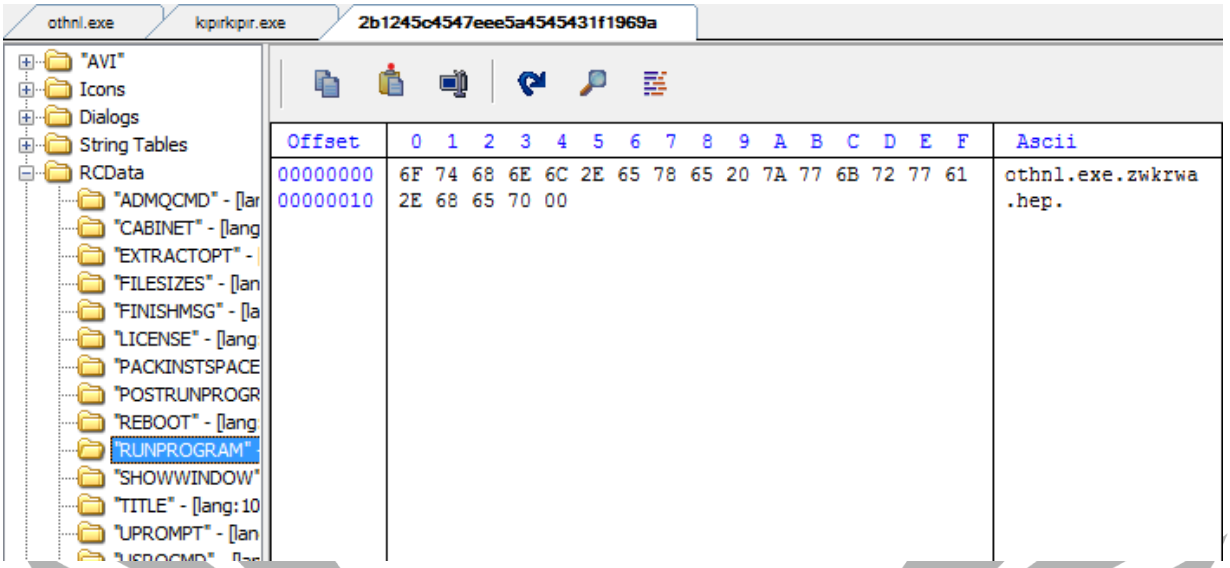


Image 2-Source view

With the “RUNPROGRAM” in its sources, information on how to run the file is reached. When “Othnl.exe” runs by taking the “zwkrwa.hep” file as parameter, malicious activities occur.

Dynamic Analysis

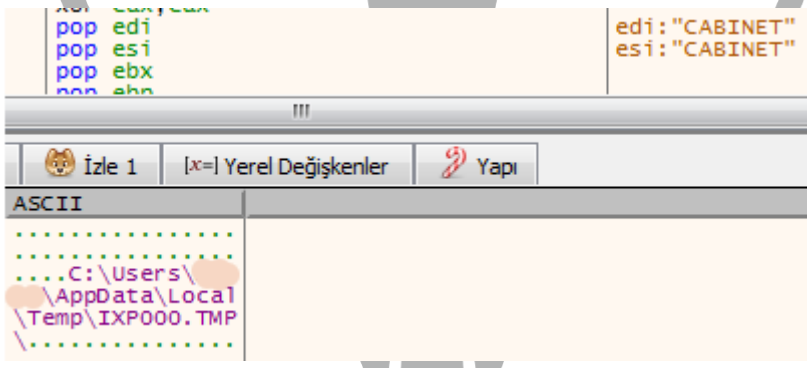


Image 3-Extracted file path

It extracts the files into the “IXP000.TMP” folder created in the “C:\Users\%USERNAME%\AppData\Local\Temp\” directory specified in the memory in Image-3, in order to keep the files temporarily.

Ad	Değiştirme tarihi	Tür	Boyut
lyzbolct.osn	31.07.2022 22:25	OSN Dosyası	274 KB
othnl.exe	09.10.2016 15:20	Uygulama	918 KB
zwkrwa.hep	31.07.2022 22:25	HEP Dosyası	116.452 KB

Image 4-Extracted files

The **Zwkrwa.hep** file must be given to **othnl.exe** as a parameter, otherwise **othnl.exe** comes as a software which is waiting for a command to execute only **AU3** files. The software with malicious activities is loaded with othnl.exe (AutoIt V3, 3, 10, 0) shown in Image-4. Also, **lyzbolct.osn** is an ecrypted file.

Othnl.exe Analysis

Name	Othnl.exe
MD5	ad5e6eb33f8b6b48fab6d9ab3e1212c1
SHA256	dd998d69304649d295691a188f8d0b04b4c2ca5dc7fb03494867bd7738200daa
File Type	PE32 / EXE

Dynamic Analysis

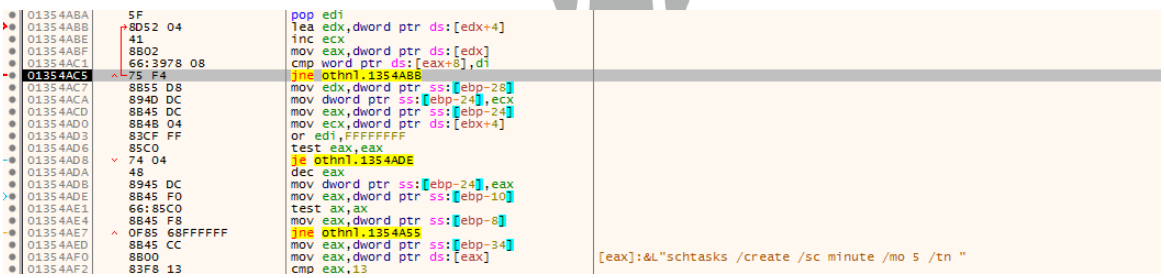


Image 5- CMD command used to persistence

In order to maintain its continuity on the device, it activates **schtask.exe,** which is a task time management application, with command information, and provides persistence by starting malicious execution every 5 minutes.
(schtasks /create /sc minute /mo 5 /tn %s)

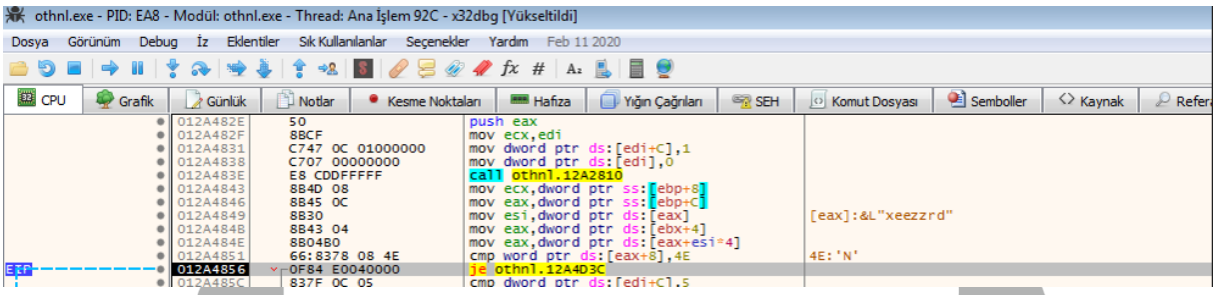


Image 6- Malicious file folder

Othnl.exe creates the **"C:\Users\%USERNAME%\xeezzrd**" directory and saves the malware in this directory. If this directory exists, **no harmful activity** is observed.

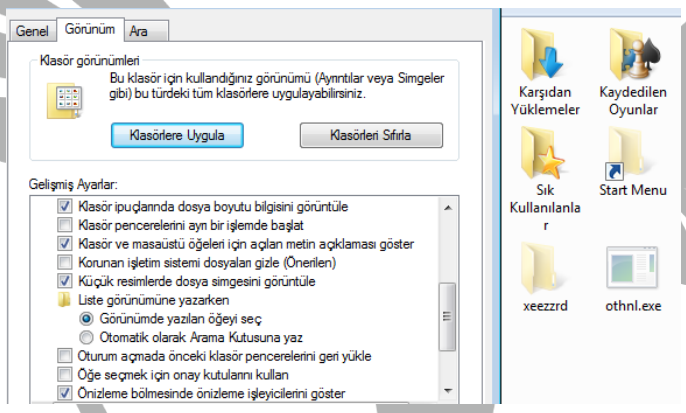


Image 7- Hiding for persistence

Files and folders appear when the "**Hide protected operating system files**" option of the view properties is turned off.With this method, the attacker aims to make detection difficult by showing his own applications as belonging to the system.

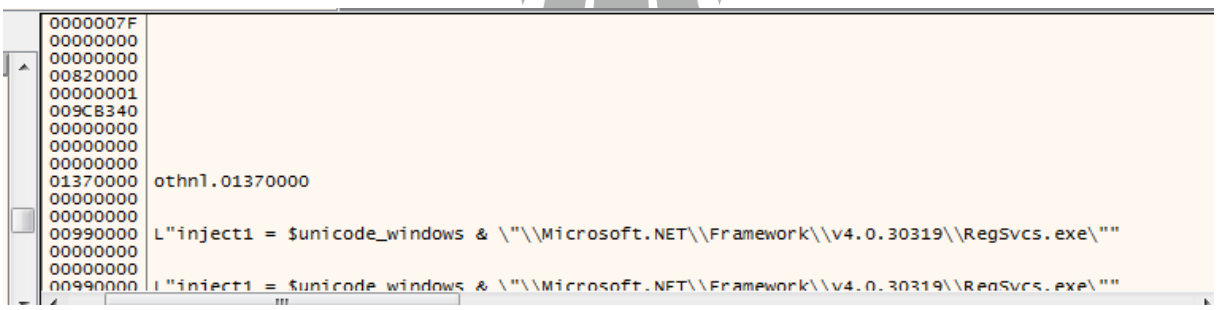


Image 8-Process injection

Process Injection is implemented on legal software "**RegSvcs.exe**". In this way, it aims not to be detected by security applications. It also complicates analysis.

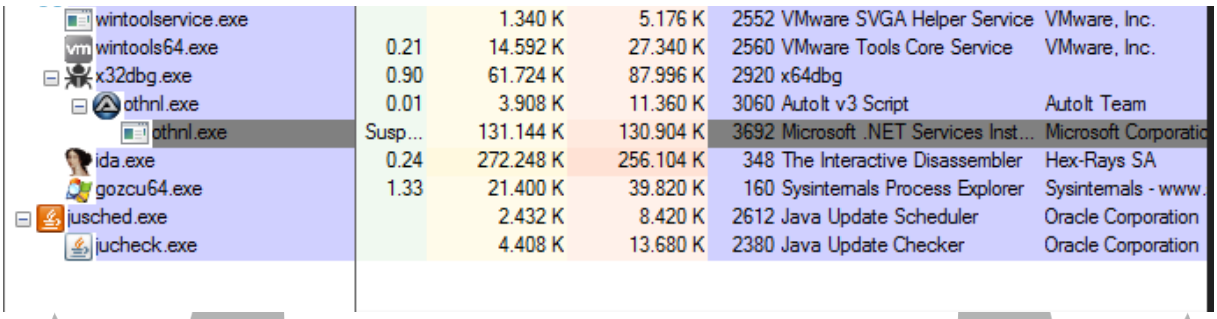


Image 9- Process Hollowing

The process is started as "**Suspend**" and it is clearly observed from its name and file sizes that it is not "**RegSvcs.exe**". It presents itself as a **legal** process with the **Process Hollowing** technique.

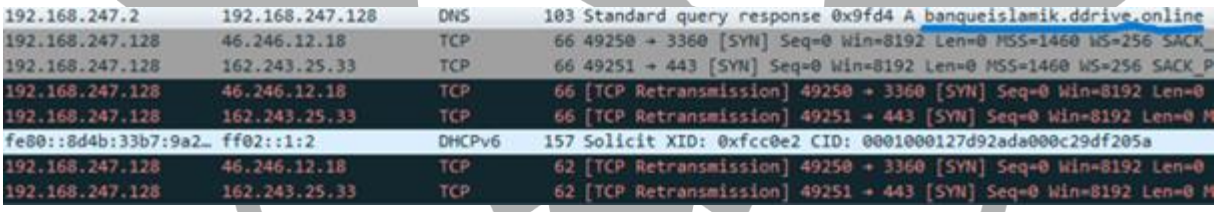


Image 10- IP and Domanin of the control server

The malware is **constantly** trying to connect to the command and control server. Because of the connection cannot be obtained here, the process repeats constantly.

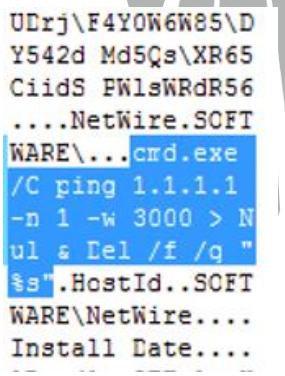


Image 11-CMD Script

With the CMD Script in the image, the network connection is checked and then it deletes itself. In addition, the text "**NetWire SOFTWARE**" and encrypted file directories are clearly observed.

cmd.exe /C ping 1.1.1.1 -n 1 -w 3000 > Nul & Del /f /q "s"

Kıpırkıpır.exe Analysis

Name	kıpırkıpır.exe
MD5	5b7e592b91d231807c75fd166e51e144
SHA256	45803a77c6a4211b8d7a342c9d9fc4625e90bbc9195675e01191638b8f05718a
File Type	PE32 / EXE

Static Analysis

The attacker aim to steal **browser credentials and passwords** from infected devices. There are some targeted browsers in the table below.

Google\Chrome\User Data\Default\LoginDataCopy	Google\Chrome\User Data\Default\Login Data
Google\Chrome\User Data\Local State	Chromium\User Data Default\LoginDataCopy
Chromium\User Data\Default\Login Data	Chromium\User Data\Local State
Comodo\Dragon\User Data\Default\LoginDataCopy	Comodo\Dragon\User Data\Local State
BraveSoftware\Brave-Browser\User Data\Default\LoginDataCopy	Yandex\YandexBrowser\User Data\Default\LoginDataCopy
Yandex\YandexBrowser User Data\Default\Login Data	Comodo\Dragon\User Data\Default\Login Data
Yandex\YandexBrowser\User Data\Local State	360Chrome\Chrome\User Data\Local State
BraveSoftware\Brave-Browser\User Data\Default\Login Data	Brave Software\Brave-Browser\User Data\Local State
360 Chrome\Chrome\User Data\Default\LoginDataCopy	Opera Software\Opera Stable\Login Data
Opera Software\Opera Stable\LoginDataCopy	Opera Software\Opera Stable\Local State

Some of the file paths of sensitive data on various targeted browsers are shown in the table.

The texts shown in the tables below are encrypted by the attacker using the **substitution** method and are decoded at runtime.

9HGGpEd5XR5dORCIHdZMIW5	9HGGpDQ5IdR54YC5d	9mpcC6doOadywSd	MjPXqjFpx80ddX5d1	9HGGMarpadYOZ55
9HGGp_OddMiw5	LMMPMIQ5SWER	67145dNpWsR	67i45dNpYiW6d	67i45dNpsOdsCodp2h
67i45dNp65ds	67145dNpYWIQIRp5df5	MT_qUDrj\FWk4ii	PQO0dR5zd064WR	IWkniQd.Sii
6didY5 * 80WI IWkpiWn4R6	XR65Cii a40dY5WOZ	IWKQ5416.Sii	IWk67i45dN.Sii	QYO5VC6d.Sii
MT_qUDrj\FWk4iiC\%6\%6\FC4R	%6\FWk4iiC_40d8Wf %6	2YOQR541dGOy.Sii	162YsGOy.Sii	MT_qUDrj\FWk4iiC\
%6\FWk4iiC_40d8Wf\sO W84id6.4R4	%6\qIQRSJOV40S\%6	PQ0OdR5zd064WR	R6500.Sii	siYO.Sii
%6\qIQRSDOV40S\s0W84id6.4R4	XR65Cii a40dY5WOZ	siS60.Sii	162YOGYy.Sii	R66054iN.Sii
MT_qUDrj\FWk4jC\%6\%6\FC4R	67145dN.Sii	6W85WwRN.Sii	162YOGhy.Sii	162YsGhy.Sii

A Python script was created using a dictionary fort he solution of strings.

```
QY05VC6d.Sii = ucrtbase.dll
2Y0QR541dGOy.Sii = vcruntime140.dll
l62YsGOy.Sii = msvcp140.dll
lWkQ54i6.Sii = mozutils.dll
lWkniQd.Sii = mozglue.dll
lWk67i45dN.Sii = mozsqlite3.dll
FWk4iiC\MdCFWRwdZ\s0W84id6.4R4 = Mozilla\SeaMonkey\profiles.ini
64nRWR6.67i45d = signons.sqlite
iWn4R6.e6WR = logins.json
sQ0sid\CYYWQR56.fli = purple\accounts.xml
qIQRSd0V40S\s0W84id6.4R4 = Thunderbird\profiles.ini
67i45dNpYiW6d = sqlite3.close
67i45dNpWsdR = sqlite3.open
67i45dNps0dsC0dp2h = sqlite3.prepare.v2
67i45dNp65ds = sqlite3.step
67i45dNpYWiQlRp5df5 = sqlite3.column.text
6didY5 *  80Wl lWkpiWn4R6 = select *  from moz.logins
IW65RCld = hostname
MFq9 9C66gW0S = SMTp password
jDM u6d0 = EAS User
jDM Md02d0 urm = EAS Server URm
jDM 9C66gW0S = EAS password
Y0Zs5Nh.Sii = crypt32.dll
P0Zs5uRs0W5dY5aC5C = CryptUnprotectData
4RSdf.SC5 = index.dat
2CQi5Yi4.Sii = vaultcli.dll
zCQi5TsdRzCQi5 = VaultOpenVault
zCQi5PiW6dzCQi5 = VaultCloseVault
zCQi5jRQld0C5dX5dl6 = VaultEnumerateitems
zCQi5Ed5X5dl = VaultGetitem
6Z65dlNh\YlS.dfd = system32\cmd.exe
Ed5LC542dMZ65dlXR8 = GetNativeSysteminf
wd0RdiNh.Sii = kernel32.dll
EiWVCiFdlW0ZM5C5Q6jf = GlobalMemoryStatusEx
-DraUDrj jMPrX9qXTL\MZ65dl\PdR50Ci90WYd66W0\y = HARDWARE ESCRipTiON\System\Centralprocessor\0
DiiWYC5dDRSXR454Ci4kdM4S = AllocateAndinitializeSid
PIdYwqkWdRFdlVd06I4s = CheckTokenMembership
jMPrX9qXTL = ESCRipTiON
```

Image 12-Deciphered texts

It seems that there are many operations related to **SQL query, browser information, passwords** among the texts.

Dynamic Analysis

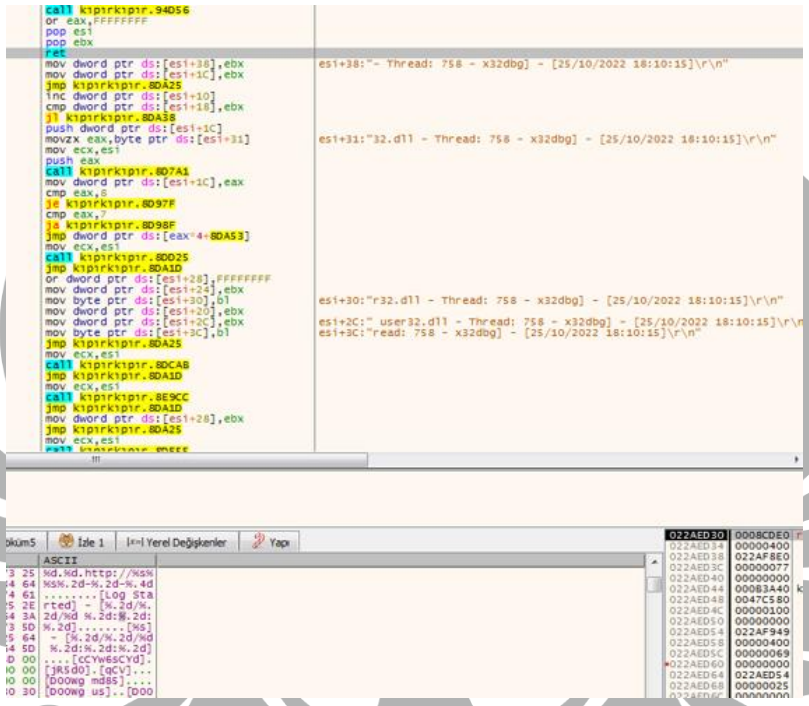


Image 13-Log Fileı (DD:MM:YYYY)

To save the log file, the state of the information obtained before encrypted is observed.

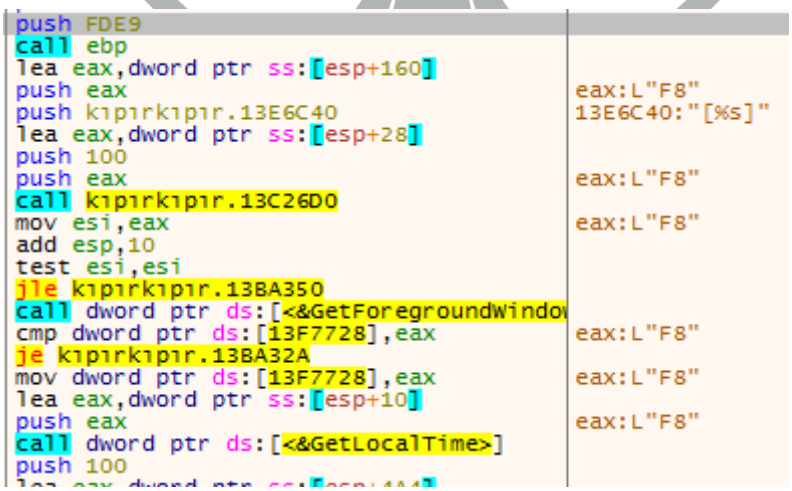


Image 14- Current process API and keystroke on it

It detects the instant application with the API used, then records the keystrokes made within the application and collects log data in this way.

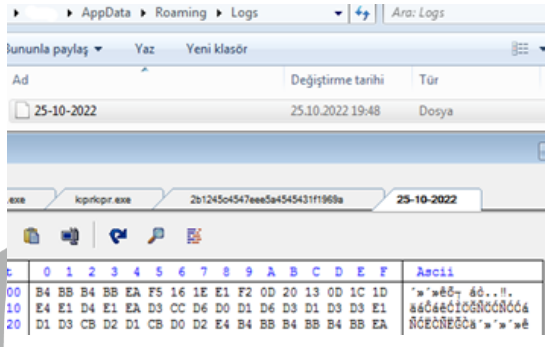


Image 15-File path &File & Encrypted logs

Keystrokes and window information created in the "**AppData\Roaming\Logs**" directory (using the date as a file name) and saved in encrypted form are kept as in the image to be sent to the command and control server.

```
mov al,byte ptr ds:[ecx+esi]
xor al,9D
add al,24
mov byte ptr ds:[ecx+esi],al
```

Image 16-Encryption

The recorded data is encrypted by applying the process in the image. By applying the reverse of this process, the raw form of the data is obtained.

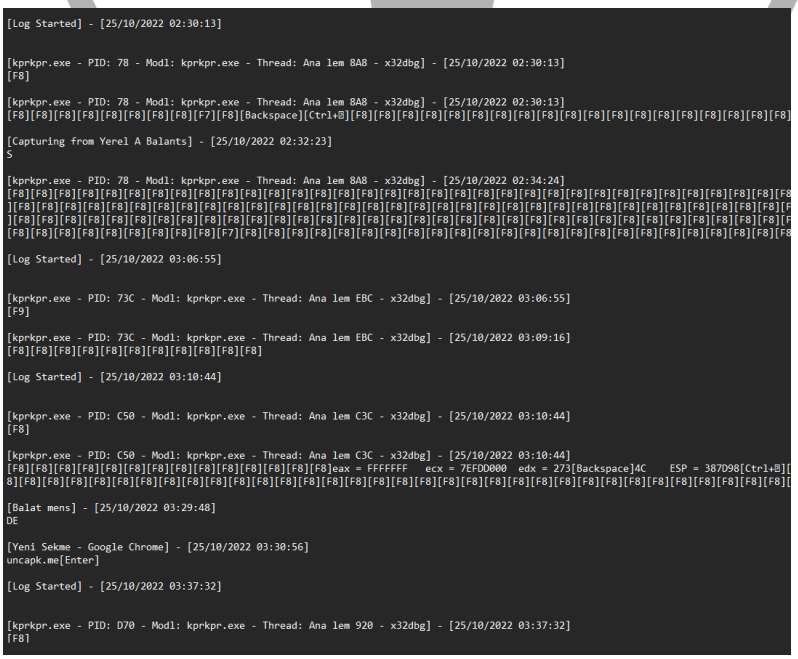


Image 17-Solved Logs

All keyboard and application activities are encrypted. Its resolved state is shown in image-17. With this method, the attacker can capture sensitive data such as **credit card, bank, account information** of the user.

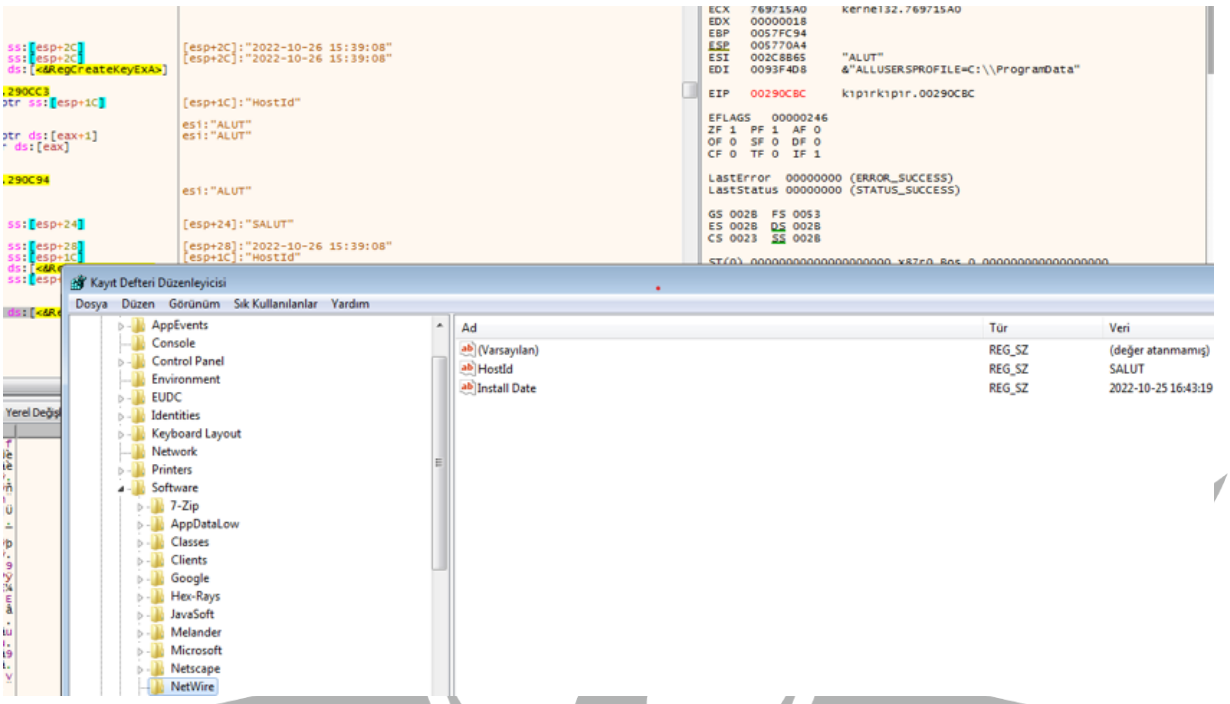


Image 18-Registry activity

By creating a new record, the **Hostid** value and the **Install Date** are added to the registry under the "**NetWire**" directory.

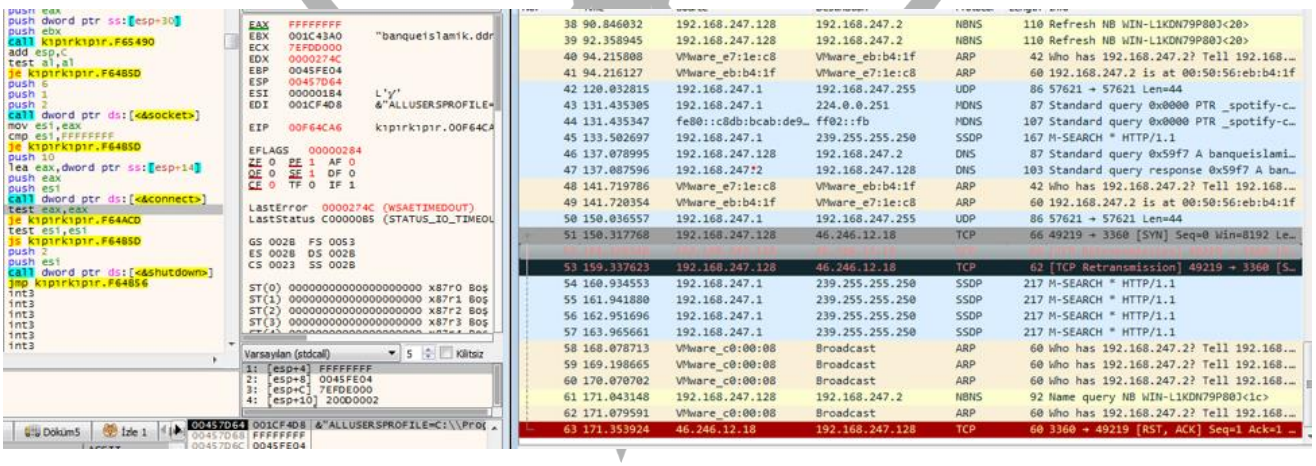


Image 19- TCP connection

It tries to connect to the socket, but receives the **RST packet** because the server is currently down. (banqueislamik[.]ddrive[.]online:3360)

YARA Rule

```
import “hash”

rule xox

{

    meta:

        author = "enessakircolak”

        date = “28.10.2022”

    strings:

        $a = “zwkrwa.hep”

        $b = “othnl.exe”

        $c = "lyzbolct.osn"

        $d = “ISOBURN.EXE.MUI”

        $e = “POSTRUNPROGRAM”

        $f = “IXP000.TMP”

    condition:

hash.m5(0,filesize) == “5c9ad0440fefa31403bd944a1a10a3b8” or all of them

}
```

```
import “hash”

rule kprkpr

{

 meta:

        author = "enessakircolak"

        date = "28.10.2022"

 strings:

        $a = "SOFTWARE\NetWire"

        $b = "Cs43l63g4R3YW0d34R5d0iWYwdS3iG3G3y.Sii"

        $c = “%%.2d/%.2d/%d %.2%.4d-%.2d-%.2d %GRN9sY1n3Ppc7g-ClJWhj0m5o2ErLt6vQASx4VuXdZibUIey_BqwHaF8TkKDMfOz%s”

        $d = “http://%s%%s%.2d-%.2d-%.4d”

        $e = “MT_qUDrj\F4Y0W6W85\U4RSWg6\PQ00dR5zd064WR\rQR\”

        $f=“Software\Microsoft\Office\15.0\Outlook\Profiles\Outlook\9375CFF0413111d3B88A00104B2A6676”

        $g = “banqueislamik.ddrive.online:3360”

 condition:

        hash.md5(0,filesize) == “5b7e592b91d231807c75fd166e51e144” or any of ($g, $a, $e) or all of ($b, $c, $d, $f)

}
```

MITRE ATTACK TABLE

Reconnaissance	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	C&C	Exfliration
Gather Victim Host Information (T1592)	Windows Command Shell (T1059)	Scheduled Tasks/Job (T1053.005)	Manipulate System Process (T1053)	Hidden Files and Directories (T1564.001)	OS Credential Dumping (T1003)	Remote Access Software (T1219)	Exfliration Over C2 Channel (T1041)
Hardware (T1592)	Scheduled Task (T1053)	Startup Folder (T1547.001)	Process Injection (T1055)	File / String Obfuscation (T1027)	Credentials From Web Browsers (T1606)	Application Layer Protocol (T1071)	
	Startup Folder (T1547.001)	Modify System Process (T1543)	Registery Run Keys (T1547)	Anti-Debugger (T1622)	Keylogging (T1056)	Encrypted Channel (T1573)	
				Software Packing (T1027)			

Solution Proposals

1. The system should be kept up to date.
2. The links in the PDF should not be clicked without looking at the target address.
3. E-mail documents, whether commercial, individual or community, should be inspected.
4. Every process must be inspected at runtime.
5. A reliable anti-virus software should be used.

Prepared By

Enes Şakir Çolak LinkedIn