

Deep Malware and Phishing Analysis

By Joe Security LLC

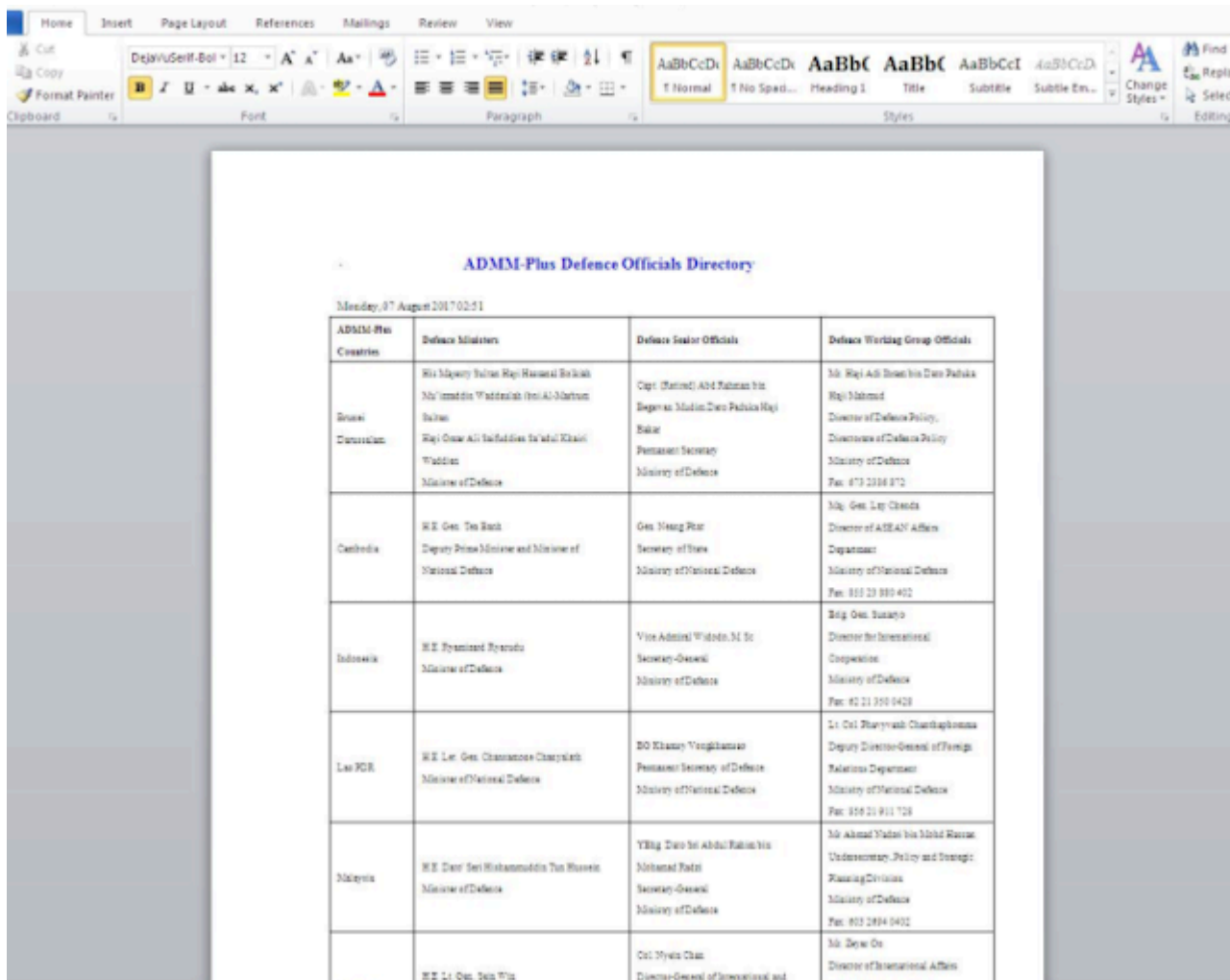
Archived: 2026-04-05 13:21:56 UTC



Recently we came across an interesting sample which seems to be related to Elise Malware. Elise is tight to the Dragon Fish and Lotus Blossom APT groups which primary targets governments and defense contractors. Elise is known to infect victims by using the latest exploits available and is often packed with interesting Sandbox evasion techniques.

In this blog post, we will dissect the latest version of Elise.

The sample under investigation is distributed as an Office document lure. To be more precise in Rich Text Format.



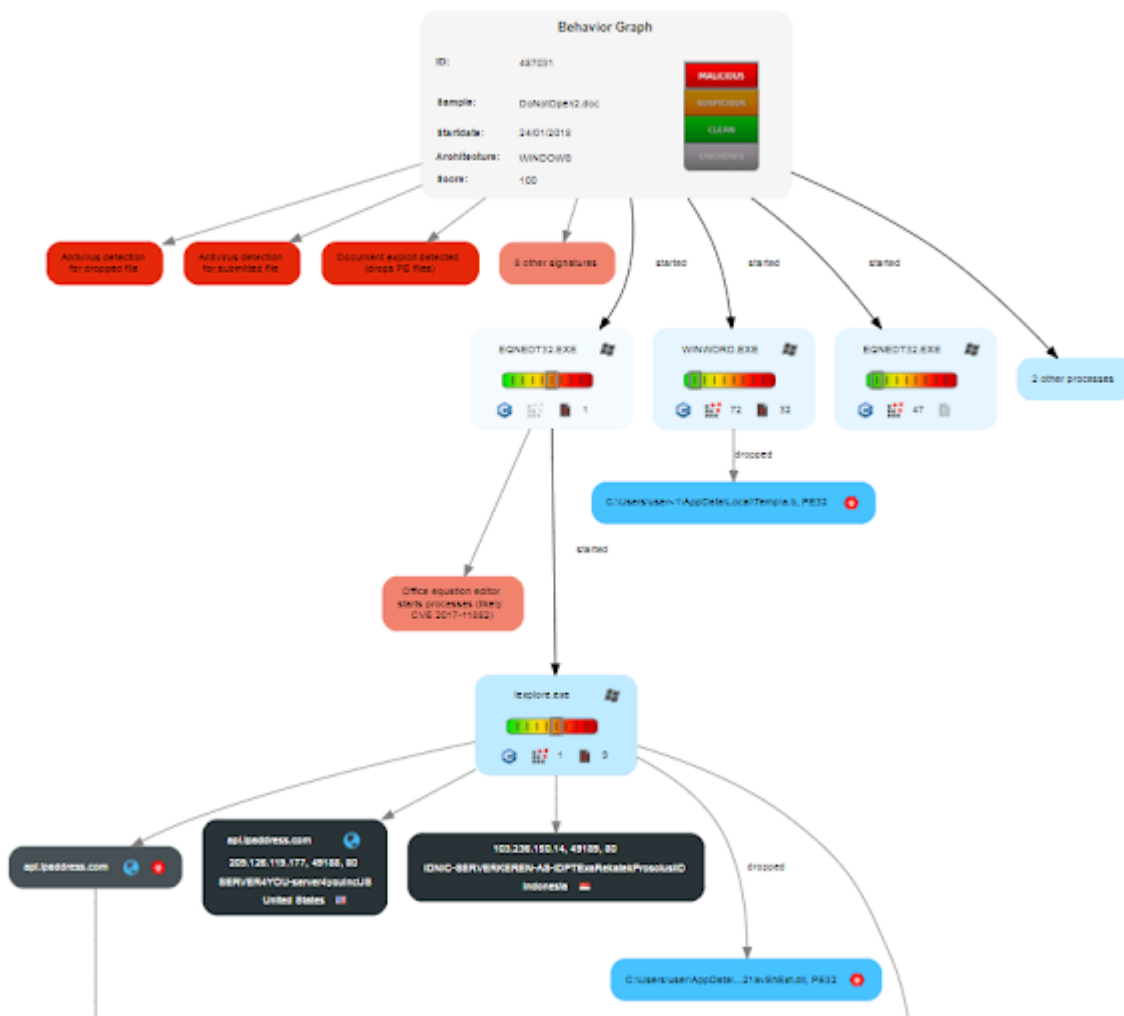
Static File Info

General

File type:	Rich Text Format data, version 1, unknown character set
Entropy (8bit):	4.652722413777991
TrID:	<ul style="list-style-type: none"> Rich Text Format (5005/1) 55.56% Rich Text Format (4004/1) 44.44%
File name:	DoNotOpen2.doc
File size:	261090
MD5:	f12fc711529b48bcef52c5ca0a52335a
SHA1:	5f89a6b2f1f38b581c65e9a1117c43a3060bfcd1
SHA256:	d3fc69a9f2ae2c446434abbfbc1693ef0f81a5da0a7f39d27c80d85f4a49c411
SHA512:	dcec5673653561354867fa1586a60899e4fd952fd693922aaba86c765710cd32186ca7c1c
File Content Preview:	{\rtf1\adeflang1025\ansi\ansicpg936\uc2\adeff0\deff0\stshfdbch13\stshfloch0\stshfhich0\stshflangc0\fonttbl{\f0\bidi \froman\charset0\prq2{*\panose 02020603050405

CVE-2018-0802

We start the analysis by having a look at the behavior graph and acknowledge that the process EQNEDT32.EXE was started among Winword.exe:



This process is the Microsoft Office Equation Editor. In November 2017 the security company [Embedi](#) detected an exploit in EQNEDT32.EXE which later got the identification [CVE-2017-11882](#). Microsoft patched the flaw in November.

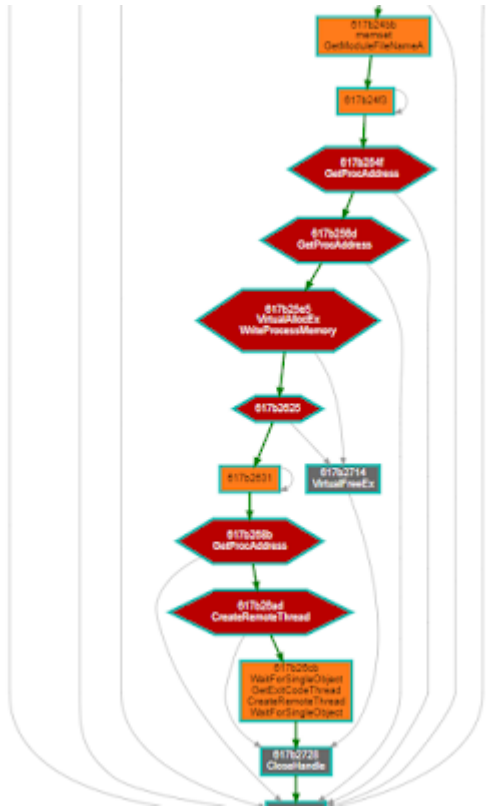
So, is Elise using this exploit? To answer this question we had a detailed look at the exploit itself. The outcome: no it is not CVE-2017-11882 but rather [CVE-2018-0802](#). CVE-2018-0802? This a second exploit also included in EQNEDT32.EXE which was detected in later December.

We extracted the trampoline and shellcode:

```

19 0029741 | 86 00 00 | | mov esi, dword ptr ds:[eax+1] | | st1_LDR_DATA_TABLE_ENTRY.dllName
20 0029744 | 80 70 20 | | mov esi, dword ptr ds:[eax+20] | | st1_LDR_DATA_TABLE_ENTRY.BaseDllName.Buffer
21 0029747 | 86 00 | | mov esi, dword ptr ds:[eax] | | next_obj
22 0029749 | 80 08 08 | | cmp byte ptr ds:[eax], 08 | | st1_LDR_DATA_TABLE_ENTRY.BaseDllName.Buffer[0] == 'k'
23 002974c | 75 P9 | | jnz 2D9741 | |
24 002974e | 80 78 02 65 | | cmp byte ptr ds:[eax+2], 65 | | st1_LDR_DATA_TABLE_ENTRY.BaseDllName.Buffer[2] == 'e'
25 0029752 | 75 80 | | jnz 2D9741 | |
26 0029754 | 80 78 0E 32 | | cmp byte ptr ds:[eax+8], 32 | | st1_LDR_DATA_TABLE_ENTRY.BaseDllName.Buffer[10] == '2'
27 0029758 | 75 87 | | jnz 2D9741 | |
28 0029765 | 48 7C | | mov esi, esi | |
29 0029767 | E8 80 00 00 80 | | call 2D9624 | | // resolve GetProcAddress, LoadLibraryA
30 002976c | E2 P9 | | loop 2D9747 | |
31 0029778 | 54 | | push esi | | // WideString
32 0029779 | 55 | | push esi | | // GetProcAddress(hKernel32, WideString)
33 002977a | FF 14 | | call dword ptr ds:[eax] | |
34 002977c | 49 44 08 | | mov dword ptr ds:[eax+4], eax | |
35 0029780 | 54 | | push esi | | // GetTempPathA
36 0029781 | 55 | | push esi | | // GetProcAddress(hKernel32, GetTempPathA)
37 0029782 | FF 16 | | call dword ptr ds:[eax] | |
38 0029784 | 89 44 0C | | mov dword ptr ds:[eax+4], eax | |
39 0029786 | 54 | | push esi | | // MoveFileExA
40 0029787 | 55 | | push esi | | // GetProcAddress(hKernel32, MoveFileExA)
41 0029788 | FF 16 | | call dword ptr ds:[eax] | |
42 002978a | E5 44 10 | | mov dword ptr ds:[eax+10], eax | |
43 002978c | 54 | | push esi | | // ExitProcess
44 002978d | 55 | | push esi | | // GetProcAddress(hKernel32, ExitProcess)
45 002978e | FF 16 | | call dword ptr ds:[eax] | |
46 0029793 | 89 44 14 | | mov dword ptr ds:[eax+14], eax | |
47 0029796 | 6A 50 | | push 50 | |
48 002979c | FF 0C | | call dword ptr ds:[eax+0C] | | // GetTempPathA
49 0029797 | 75 P9 | | jnz 2D9742 | |
50 0029799 | 89 04 00 00 80 | | mov eax, 4 | | // a.3
51 002979e | 46 4C 2C 33 80 | | push 2D9742 | |
52 002979f | 80 P4 | | mov esi, esi | |
53 | | | | | |
54 0029815 | FF 04 10 | | call dword ptr ds:[eax+10] | | MoveFileExA("C:\\Users\\user\\AppData\\Local\\Temp\\a.b", "C:\\Users\\user\\AppData\\Local\\a-3", MOVEFILE_REPLACE_EXISTING)
55 0029814 | 54 | | push esi | |
56 0029815 | FF 04 04 | | call dword ptr ds:[eax+4] | | LoadLibraryA("C:\\Users\\user\\AppData\\Local\\a-3")
57 0029818 | 6A 01 | | push 1 | |
58 002981a | 50 | | push esi | |
59 002981b | FF 14 | | call dword ptr ds:[eax] | | GetProcAddress(1)
60 002981c | FF 00 | | call esi | | (1)
61 002981f | 6A 00 | | push 0 | |
62 0029821 | FF 04 14 | | call dword ptr ds:[eax+14] | | ExitProcess(0)
63 | | | | | |
64 0029824 | 51 | | push esi | |
65 0029825 | 54 | | push esi | |
66 0029826 | 52 | | push esi | |
    
```

The code renames and loads the PE file (named a.b) previously dropped by Word. The newly loaded code is then injected into IExplorer.exe where the main payload is executed:



617B2700	push eax	
617B2701	push dword ptr [ebp-0000024Ch]	
617B2707	call dword ptr [ebp-00000234h]	CreateRemoteThread@KERNEL32.DLL (Import, Hidden, 7 Params) executed
617B270D	push FFFFFFFFh	executed
617B270F	push eax	
617B2710	call edi	WaitForSingleObject@KERNEL32.DLL (Import, 2 Params)
617B2712	jmp 617B2728h	target: 617B2728
617B2714	push edi	xref: 617B261F 617B262B
617B2715	push esi	
617B2716	push dword ptr [ebp-00000230h]	
617B271C	push dword ptr [ebp-0000024Ch]	
617B2722	call dword ptr [617B4048h]	VirtualFreeEx@KERNEL32.DLL (Import, Unknown Params)
617B2728	push dword ptr [ebp-0000024Ch]	xref: 617B26C9 617B2712
617B272E	call dword ptr [617B406Ch]	CloseHandle@KERNEL32.DLL (Import, 1 Params)
617B2734	pop edi	xref: 617B248E 617B24B5 617B2567 617B25DF 617B26A7

Sandbox Evasions

Elise performs a variety of sandbox checks in In IEplorer:

Function 004F278F, Relevance: 2.5, Strings: 2, Instructions: 23

Strings

- hXMV, xrefs: 004F27A6
- hXMV, xrefs: 004F27BB

Memory Dump Source

- Source File: 00000003.00000002.773005659.004F0000.00000040.sdmp, Offset: 004F0000, based on PE: true

Joe Sandbox IDA Plugin

- Snapshot File: hcareult_3_2_4f0000_iexplore.jbxid

Address	Instruction	Meta Information
004F278F	push 0000000Ch	
004F2791	push 004F6CC0h	
004F2796	call 004F5964h	target: 004F5964
004F279B	mov byte ptr [ebp-19h], 00000001h	
004F279F	and dword ptr [ebp-04h], 00000000h	
004F27A3	push edx	
004F27A4	push ecx	
004F27A5	push ebx	
004F27A6	mov eax, 564D5868h	ASCII "hXMV" (Chunk)
004F27AB	mov ebx, 00000000h	
004F27B0	mov ecx, 0000000Ah	
004F27B5	mov edx, 00005658h	
004F27BA	in eax, dx	
004F27BB	cmp ebx, 564D5868h	ASCII "hXMV" (Chunk)
004F27C1	sete byte ptr [ebp-19h]	

VMware backdoor check

Function 004F27E5, Relevance: 28.1, APIs: 10, Strings: 6, Instructions: 94 REGISTRY

APIs

- **RegOpenKeyExW**.ADVAPI32(80000002,SYSTEM\ControlSet001\services\Disk\Enum,00000000,00020019,?,
- **GetLastError**.KERNEL32 ref: [004F2824](#)
- **printf**.MSVCRT ref: [004F2830](#)
- **memset**.MSVCRT ref: [004F2852](#)
- **RegQueryValueExW**.ADVAPI32(?,004F646C,00000000,00000000,?,?), ref: [004F286C](#)
- **wcsstr**.MSVCRT ref: [004F288B](#)
- **wcsstr**.MSVCRT ref: [004F28A2](#)
- **wcsstr**.MSVCRT ref: [004F28B8](#)
- **wcsstr**.MSVCRT ref: [004F28CE](#)
- **RegCloseKey**.ADVAPI32(?), ref: [004F28DE](#)
 - Part of subcall function 004F32A0: **SetUnhandledExceptionFilter**.KERNEL32(00000000), ref: [004F3CE2](#)
 - Part of subcall function 004F32A0: **UnhandledExceptionFilter**.KERNEL32(004F6190), ref: [004F3CED](#)
 - Part of subcall function 004F32A0: **GetCurrentProcess**.KERNEL32(C0000409), ref: [004F3CF8](#)
 - Part of subcall function 004F32A0: **TerminateProcess**.KERNEL32(00000000), ref: [004F3CFF](#)
 - Part of subcall function 004F3197: **_errno**.MSVCRT ref: [004F31A4](#)
 - Part of subcall function 004F3197: **_wcslwr**.MSVCRT ref: [004F31D6](#)

Strings

- virtualhd, xrefs: [004F28C8](#)
- vmware, xrefs: [004F2885](#)
- 0x3A RegOpenKeyExW Disk Failed-%d, xrefs: [004F282B](#)
- qemu, xrefs: [004F289C](#)
- SYSTEM\ControlSet001\services\Disk\Enum, xrefs: [004F280D](#)
- vbox, xrefs: [004F28B2](#)

Disk Name Check

Address	Instruction	Comment
004F2912	stosd	Count: 5
004F2917	mov edi, dword ptr [004F6000h]	RegOpenKeyExW@ADVAPI32.DLL (Import, Unknown Params)
004F291D	xor esi, esi	
004F291F	mov dword ptr [ebp-40h], 004F64ACh	UTF-16 "Software\CommView"
004F2926	mov dword ptr [ebp-3Ch], 004F64D0h	UTF-16 "Software\Eye Digital Security"
004F292D	mov dword ptr [ebp-38h], 004F6510h	UTF-16 "Software\Win Sniffer"
004F2934	mov dword ptr [ebp-34h], 004F6540h	UTF-16 "Software\Microsoft\Windows\CurrentVersion\Explorer\MenuOrder\Start Menu2\Programs\APIS32"
004F293B	mov dword ptr [ebp-30h], 004F65F4h	UTF-16 "Software\Syser Soft"
004F2942	mov dword ptr [ebp-2Ch], 004F6620h	UTF-16 "Software\Classes\Folder\shell\sandbox"
004F2949	mov dword ptr [ebp-28h], 004F6670h	UTF-16 "Software\Classes\shell\sandbox"
004F2950	mov dword ptr [ebp-24h], 004F66B8h	UTF-16 "SYSTEM\CurrentControlSet\Services\IRIS5"
004F2957	mov dword ptr [ebp-20h], 004F6708h	UTF-16 "SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\Wireshark"
004F295E	mov dword ptr [ebp-1Ch], 004F6784h	UTF-16 "SOFTWARE\ZxSniffer"
004F2965	mov dword ptr [ebp-18h], 004F67B0h	UTF-16 "SYSTEM\CurrentControlSet\Services\VBxGuest"
004F296C	mov dword ptr [ebp-14h], 004F6808h	UTF-16 "SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\Oracle VM VirtualBox Guest Additions"
004F2973	mov dword ptr [ebp-10h], 004F68CDh	UTF-16 "SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\Sandboxie"

Check for various Analysis Tools

Function 004F2A05, Relevance: 46.0, APIs: 7, Strings: 19, Instructions: 482

APIs

- **CreateToolhelp32Snapshot**.KERNEL32(00000002,00000000), ref: [004F2A2A](#)
- **memset**.MSVCRT ref: [004F2A51](#)
- **memset**.MSVCRT ref: [004F2A6C](#)
- **Process32FirstW**.KERNEL32(?,?), ref: [004F2A7E](#)
- **__swprintf_I**.LIBCMT ref: [004F2AA4](#)
 - Part of subcall function 004F3197: **_errno**.MSVCRT ref: [004F31A4](#)
 - Part of subcall function 004F3197: **_wcslwr**.MSVCRT ref: [004F31D6](#)
- **Process32NextW**.KERNEL32(?,?), ref: [004F2F6A](#)
- **CloseHandle**.KERNEL32(?), ref: [004F2F81](#)
 - Part of subcall function 004F32A0: **SetUnhandledExceptionFilter**.KERNEL32(00000000), ref: [004F3CE2](#)
 - Part of subcall function 004F32A0: **UnhandledExceptionFilter**.KERNEL32(004F6190), ref: [004F3CED](#)
 - Part of subcall function 004F32A0: **GetCurrentProcess**.KERNEL32(C0000409), ref: [004F3CF8](#)
 - Part of subcall function 004F32A0: **TerminateProcess**.KERNEL32(00000000), ref: [004F3CFF](#)

Strings

- **irise.exe**, xrefs: [004F2C95](#)
- **vmupgradehelper.exe**, xrefs: [004F2BA9](#)
- **windbg.exe**, xrefs: [004F2E20](#)
- **Syser.exe**, xrefs: [004F2EA8](#)
- **Regshot.exe**, xrefs: [004F2D98](#)
- **SandboxieDcomLaunch.exe**, xrefs: [004F2F23](#)
- **vmtools.exe**, xrefs: [004F2C5A](#)
- **IrisSvc.exe**, xrefs: [004F2CCC](#)
- **vmwaretray.exe**, xrefs: [004F2B6E](#)
- **SandboxieRpcSs.exe**, xrefs: [004F2EEC](#)
- **vmwareuser.exe**, xrefs: [004F2B33](#)
- **vboxtray.exe**, xrefs: [004F2ABD](#)
- **vboxservice.exe**, xrefs: [004F2AF8](#)
- **wireshark.exe**, xrefs: [004F2D10](#)
- **vmacthlp.exe**, xrefs: [004F2C1F](#)
- **ollydbg.exe**, xrefs: [004F2DDC](#)
- **ZxSniffer.exe**, xrefs: [004F2D54](#)
- **vmtoolsd.exe**, xrefs: [004F2BE4](#)
- **PEBrowseDbg.exe**, xrefs: [004F2E64](#)

Process Check

Function 004F2FA1, Relevance: 42.2, APIs: 16, Strings: 8, Instructions: 155 **STRING**

APIs

- `??2@YAPAXI@Z.MSVCRT` ref: 004F2FC4
- `memset.MSVCRT` ref: 004F2FD4
- `GetAdaptersInfo.IPHLPAPI(00000000,?)`, ref: 004F2FE8
- `??3@YAXPAX@Z.MSVCRT` ref: 004F2FF5
- `??2@YAPAXI@Z.MSVCRT` ref: 004F2FFD
- `GetAdaptersInfo.IPHLPAPI(00000000,?)`, ref: 004F300B
- `memset.MSVCRT` ref: 004F3021
- `__swprintf_LLIBCMT` ref: 004F3078
- `??3@YAXPAX@Z.MSVCRT` ref: 004F309C
- `strstr.MSVCRT` ref: 004F30AB
- `strstr.MSVCRT` ref: 004F30C1
- `strstr.MSVCRT` ref: 004F30D5
- `strstr.MSVCRT` ref: 004F30E9
- `strstr.MSVCRT` ref: 004F30FD
- `strstr.MSVCRT` ref: 004F3111
- `strstr.MSVCRT` ref: 004F3125
 - Part of subcall function 004F32A0: `SetUnhandledExceptionFilter.KERNEL32(00000000)`, ref: 004F3125
 - Part of subcall function 004F32A0: `UnhandledExceptionFilter.KERNEL32(004F6190)`, ref: 004F3125
 - Part of subcall function 004F32A0: `GetCurrentProcess.KERNEL32(C0000409)`, ref: 004F3CF8
 - Part of subcall function 004F32A0: `TerminateProcess.KERNEL32(00000000)`, ref: 004F3CFF

Strings

- 00163E, xrefs: 004F310B
- 000569, xrefs: 004F30A5
- %02X%02X%02X%02X%02X%02X, xrefs: 004F306A
- 001C14, xrefs: 004F30CF
- 000C29, xrefs: 004F30BB
- 00155D, xrefs: 004F30F7
- 080027, xrefs: 004F311F
- 005056, xrefs: 004F30E3

Mac Address Check

Payloads

After passing all the sandbox checks Elise creates an autostart key:

Key Value Modified

Key Path	Name	Type	Old Data	New Data	Completion	Count	Source Address	Symbol
HKEY_USERS\Software\Microsoft\Windows\CurrentVersion\Run	IASlorD	unicode	C:\Windows\system32\rundll32.exe C:\Users\user\AppData\Roaming\Microsoft\Windows\Caches\NavShExt.dll,Setting	C:\Windows\system32\rundll32.exe C:\Users\user-1\AppData\Roaming\MICROS-1\Windows\Caches\NavShExt.dll,Setting	success or wait	1	4F11F4	RegSetValueExA

Function 0051129D, Relevance: 26.4, APIs: 10, Strings: 5, Instructions: 191 **REGISTRY**

APIs

- **memset.MSVCRT** ref: 00511301
 - Part of subcall function 00511124: **memset.MSVCRT** ref: 0051115A
 - Part of subcall function 00511124: **RegEnumKeyW**.ADVAPI32(80000003,00000000,?,00000104), ref: 0051117D
 - Part of subcall function 00511124: **wcsncmp.MSVCRT**(S-1-5-21,?,00000008), ref: 00511199
 - Part of subcall function 00511124: **wcsstr.MSVCRT** ref: 005111B1
 - Part of subcall function 00511124: **memset.MSVCRT** ref: 005111CF
 - Part of subcall function 00511124: **RegOpenKeyExW**.ADVAPI32(80000003,?,00000000,00020019,?), ref: 00511220
- **RegQueryValueExW**.ADVAPI32(00000000,ProxyEnable,00000000,00000000,?,?,?,00000000), ref: 00511345
- **RegQueryValueExW**.ADVAPI32(?,ProxyServer,00000000,00000000,?,?), ref: 00511370
- **RegCloseKey**.ADVAPI32(?), ref: 00511378
- **__swprintf_L**.LIBCMT ref: 00511398
 - Part of subcall function 00519317: **_errno**.MSVCRT ref: 00519324
 - Part of subcall function 00519317: **_wcurlwr**.MSVCRT ref: 00519356
- **wcsstr.MSVCRT** ref: 005113B9
- **wcsstr.MSVCRT** ref: 005113D3
- **wcsstr.MSVCRT** ref: 005113EE
- **memset.MSVCRT** ref: 00511412
 - Part of subcall function 0051A9BB: **_errno**.MSVCRT ref: 0051A9D3
 - Part of subcall function 0051A9BB: **_errno**.MSVCRT ref: 0051AA1C
- **__swprintf_L**.LIBCMT ref: 005114C2
 - Part of subcall function 00518133: **memset**.MSVCRT ref: 00518179
 - Part of subcall function 00518133: **GetLocalTime**.KERNEL32(?,?,?,?), ref: 00518188
 - Part of subcall function 00518133: **__swprintf_L**.LIBCMT ref: 005181D0
 - Part of subcall function 00518133: **memset**.MSVCRT ref: 0051821C
 - Part of subcall function 00518133: **WideCharToMultiByte**.KERNEL32(00000000,00000000,?,000000FF,?,0000C8), ref: 00518258
 - Part of subcall function 00518133: **CreateFileA**.KERNEL32(C:\Users\user~1\AppData\Local\Temp\FXSAPIDebugL,ref: 0051826B
 - Part of subcall function 00518133: **GetFileSize**.KERNEL32(00000000,00000000), ref: 0051826B
 - Part of subcall function 00518133: **SetEndOfFile**.KERNEL32(00000000), ref: 00518280
 - Part of subcall function 00518133: **SetFilePointer**.KERNEL32(00000000,00000000,00000000,00000002), ref: 00518280
 - Part of subcall function 00518133: **WriteFile**.KERNEL32(00000000,?,?,?,00000000), ref: 005182CB
 - Part of subcall function 00518133: **CloseHandle**.KERNEL32(00000000), ref: 005182D2
 - Part of subcall function 0051A460: **SetUnhandledExceptionFilter**.KERNEL32(00000000), ref: 0051B495
 - Part of subcall function 0051A460: **UnhandledExceptionFilter**.KERNEL32(0051E3C8), ref: 0051B4A0
 - Part of subcall function 0051A460: **GetCurrentProcess**.KERNEL32(C0000409), ref: 0051B4AB
 - Part of subcall function 0051A460: **TerminateProcess**.KERNEL32(00000000), ref: 0051B4B2

Strings

- GetIEProxy %s., xrefs: 005114CE
- ProxyServer, xrefs: 00511365
- %s=%s://%s, xrefs: 005114B3
- ProxyEnable, xrefs: 00511335
- %s=, xrefs: 00511390

Add a Proxy to Internet Explorer

- Part of subcall function 0051A99B: `_errno.MSVCRT` ref: 0051A9AD
- Part of subcall function 0051A9BB: `_errno.MSVCRT` ref: 0051A9D3
- Part of subcall function 0051A9BB: `_errno.MSVCRT` ref: 0051AA1C
- `fgetws.MSVCRT` ref: 005117B1
- `wcsstr.MSVCRT` ref: 005117C6
- `wcsstr.MSVCRT` ref: 0051185A
- `_wtoi.MSVCRT` ref: 00511881
- `feof.MSVCRT` ref: 00511893
- `fclose.MSVCRT` ref: 005118A7
- `__swprintf_I.LIBCMT` ref: 005118E3
 - Part of subcall function 00518133: `memset.MSVCRT` ref: 00518179
 - Part of subcall function 00518133: `GetLocalTime.KERNEL32(?,?,?,?)`, ref: 00518188
 - Part of subcall function 00518133: `__swprintf_I.LIBCMT` ref: 005181D0
 - Part of subcall function 00518133: `memset.MSVCRT` ref: 0051821C
 - Part of subcall function 00518133: `WideCharToMultiByte.KERNEL32(00000000,000C`
 - Part of subcall function 00518133: `CreateFileA.KERNEL32(C:\Users\user~1\AppData'`
 - Part of subcall function 00518133: `GetFileSize.KERNEL32(00000000,00000000)`, ref:
 - Part of subcall function 00518133: `SetEndOfFile.KERNEL32(00000000)`, ref: 0051828
 - Part of subcall function 00518133: `SetFilePointer.KERNEL32(00000000,00000000,0C`
 - Part of subcall function 00518133: `WriteFile.KERNEL32(00000000,?,?,?,00000000)`, r
 - Part of subcall function 00518133: `CloseHandle.KERNEL32(00000000)`, ref: 005182D

Strings

- `user_pref("network.proxy.http", , xrefs: 005116A2`
- `Get FireFoxProxy %s, xrefs: 005118E9`
- `user_pref("network.proxy.ssl_port", , xrefs: 005116D5`
- `HTTPS, xrefs: 0051172D`
- `user_pref("network.proxy.ssl", , xrefs: 005116C0`
- `profiles.ini, xrefs: 005115C6`
- `Profile%d, xrefs: 005115D9`
- `HTTP, xrefs: 005116E6`
- `\Mozilla\Firefox\, xrefs: 0051159B`
- `\prefs.js, xrefs: 0051161F`
- `%s=%s://%s:%d, xrefs: 005118DB`
- `Path, xrefs: 005115FC`
- `user_pref("network.proxy.http_port", , xrefs: 005116B2`

Add a Proxy to Firefox

Finally, in function 514D05, 5159AF and 515486 we find the download, upload and command execution handlers. Elise can collect and upload the following data:

- CPU Usage
- Ram (size/free)
- Disk space (size/free)
- Windows Version
- Username
- Locale
- Timezone
- SID

- List of tasks
- List of network adapters
- List of files on Desktop

Final Words

Elise is a very advanced piece of malware using for its distribution only the latest exploits. Before the main payload is executed many different Sandbox evasions are performed. The payload and the communication code is injected into IExplorer likely bypassing PFW and HIPS.

Source: <https://www.joesecurity.org/blog/8409877569366580427>