Sample: 19a3dd

Sample

SHA256: `19a3dd8024bb4677261ecd8bb85e8a4c53d15870e4b9d2203e933a00b7eecb85`

The sample is a packed UPX PE, that tries connect to a URI.

Task

This sample has an embedded URL that is used as a C2, what is it?

- It tries to connect to `h[tt]p://test.hacker4hire.test.com/`

What is the mutex that is created for this sample?

- It creates a mutex via the `CreateMutexA` function with the following arguments `(0, 0, "Test4U!")`

Is the sample obfuscated or packed? How?

- The sample is packed with UPX

Static Analysis

Running floss and pestudio yielded the following results:

Notable Strings

```
@http:
//%s/cgi/online.asp?hostname=AB
o&$typ
text
command
&del=
file
~UKern
32.dll
etTickCount
d
<PProxySv
\Software\Ms
\WrF\
n[{Cur
```

```
=nUrlAMoz1la/4.0 (
ZpaS'; MSIE 8
JWM0
JWM1
JWM2
```

Imports

```
FreeSid
HttpEndRequestA
VirtualProtect
```

Libs

```
KERNEL32.DLL
ADVAPI32.dll
MSVCRT.dll
USER32.dll
WININET.dll
```

Dynamic Analysis

I used Fiddler to check which C2 Server it tries to connect to. It tries to connect to `http://test.hacker4hire.test.com/` and It seems that the Sample is being packed with UPX (determined that by using `floss $file` where I saw the UPX string. It seems that the program is being obfuscated since a simple `upx -d $file` won't work.

I assumed a `pocket` trick/method is being used to scramble the Section names named `JWM(1-3)`, and I used HxD to rename them back to `UPX1..3` and I successfully unpacked it with `upx -d $file`

I ran the program through IDA Freeware's disassembler and from there I saw that it creates a mutex via the `CreateMutexA` function with the following arguments `(0, 0, "Test4U!")`, I am not sure if that is sufficient enough for this sample, I ran it through Process Hacker and Process Monitor for further investigation.

The sample is using http tunneling to establish a connection to the C2 Server, through Process Monitor it seems to be accessing a lot registers and it tries to modify them (looks like editing the registers for HTTP certificates and so on). It also tries to get a handle of some specific directories (ex. `*Windows\History\*`).

Tools

- procmon
- Fiddler
- Inetsim

- HxD
- Wireshark
- IDA Freeware
- floss
- pestudio
- VirtualBox
    - FLARE VM (Win10 Enterprise for Malware Analysis)
    - REMnux