nullday About Blog Contact German PGP

DuBrute - a rare view into chinese blackhat techniques

Sep 5, 2015

As member of the MalwareMustDie ELF-Team, I was assigned to make report to dissect the DuBrute bruteforcing tool. This report is published on the effort as the first official report that hopefully can help good people in InfoSec.

We all know the ELF malware that comes from China. Linux/MrBlack, Linux/Xor.DDoS, Linux/BillGates are just a few of them. But what we don't know is the way how our suspects are working. What is their common way to spread so much malware in only a few days? Which tools are they using? How do they work?

Our team has disclosed the video of the way how ELF malware can be spread via exploiting SSH in this malwaremustdie blog post:

http://blog.malwaremustdie.org/2014/11/china-elf-botnet-malware-infection.html

But new tools just come up to be used, this post explains one of the new used tools called "DUBrute".

Due to the team's research we have found a .zip archive in one of the chinese HFS panels. In this .zip archive we have found a picture and a directory called "DUBrute".

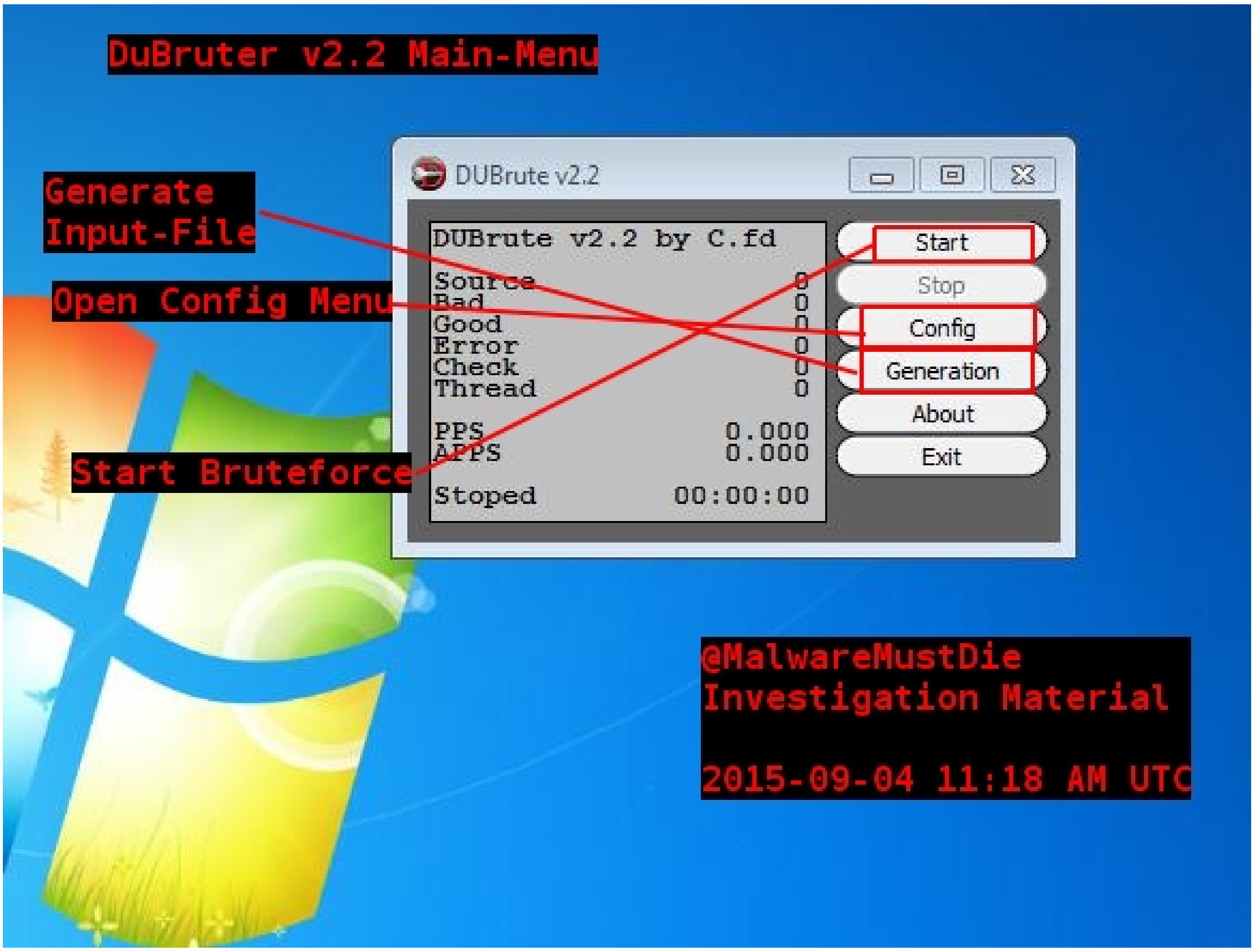
Let us have a look on this directory:

```
14b2374efbd0d6ec435eb10d053db1c3  1.bat
8f4ec24aeda03afaae1cac0d7030dcf2  2.bat
abf1f45b4fa467b830f091cf96d82586  QtCore4.dll
21e38b4a1f28bdef99d1c7dac5dce164  QtGui4.dll
f971323ecd66fdd66468c6551c8eaa79  config.ini
bac0e8e8fa5d330bc0a58e440eab5812  dubrute.exe
cdbdef73515997355e81a99421c1d721  libeay32.dll
86f1895ae8c5e8b17d99ece768a70732  msvcr71.dll
6c06fb9e1d818fde8d142ee180a65646  ssleay32.dll
16f9d4cbd70328c0370d6587b8e48c05  xf.TXT
```

```
956958b308193d9f064d49f13a4d1ee1  xiaofe.dll
```

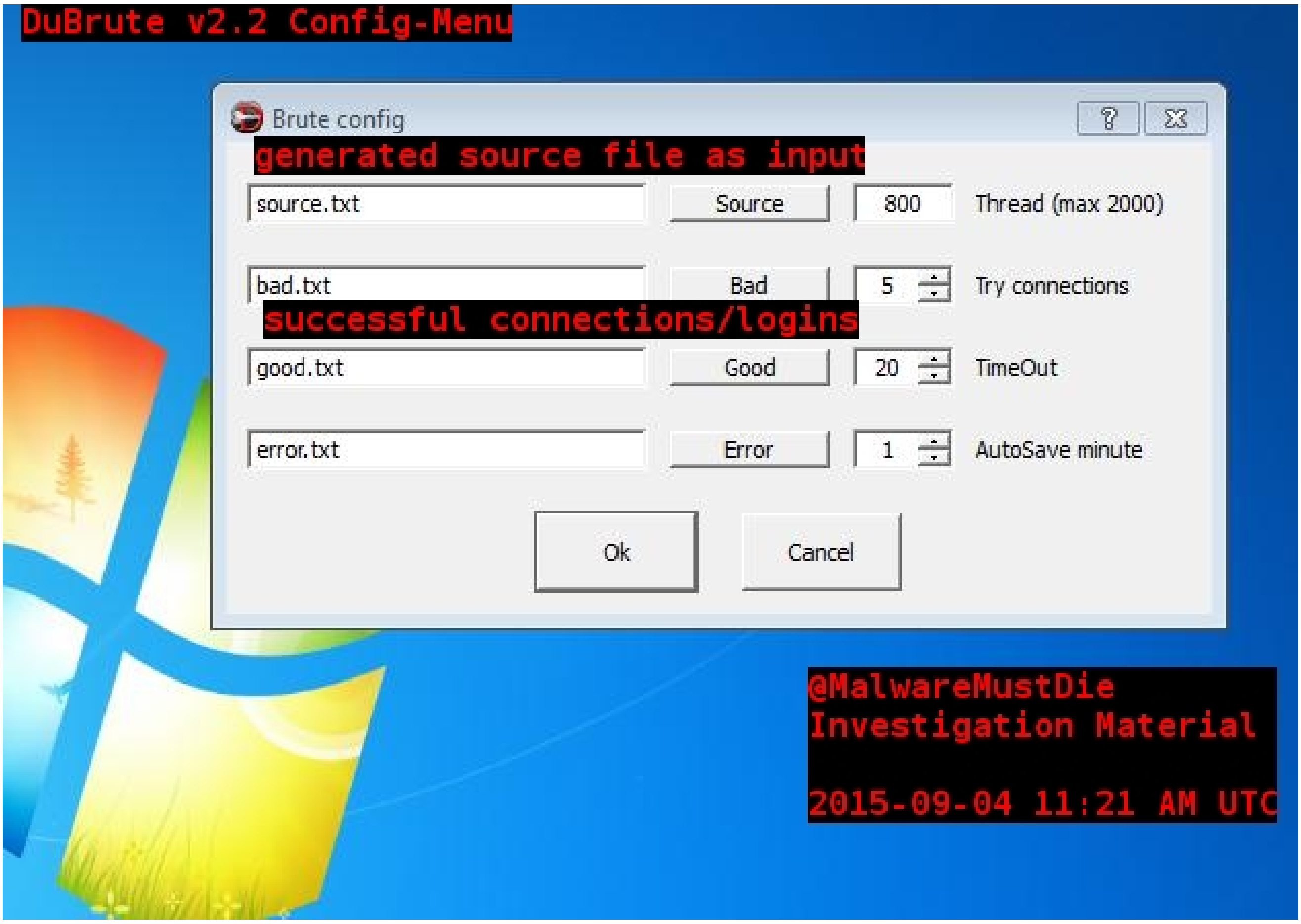
So what do we have? We have several different .dll (dynamic linked libraries) one .exe (Windows Executable), 2 batch files and one simple text file. So let us check out the dubrute.exe first:

```
dubrute.exe: PE32 executable (GUI) Intel 80386, for MS Windows
Filesize: 286720
MD5: bac0e8e8fa5d330bc0a58e440eab5812
```



```
DUBrute v2.2 by C.fd

Source             0
Bad                0
Good               0
Error              0
Check              0
Thread             0

PPS            0.000
APPS           0.000

Stoped      00:00:00
```

This is the main-menu of the DuBrute v2.2. It is pretty simple. "Start" starts the bruteforce and "Stop" stops it. "Config" opens the config-menu, "Generation" generates an input file for the tool and "Exit" exits the programm. The input file is called "Source". All successful logins will go into a file called "good.txt", errors will go into "error.txt" and IP's without a result will go into "bad.txt". Let us see the dubrute config menu:



DuBrute v2.2 Config-Menu

Brute config

generated source file as input

source.txt — 800 Thread (max 2000)

bad.txt — 5 Try connections

successful connections/logins

good.txt — 20 TimeOut

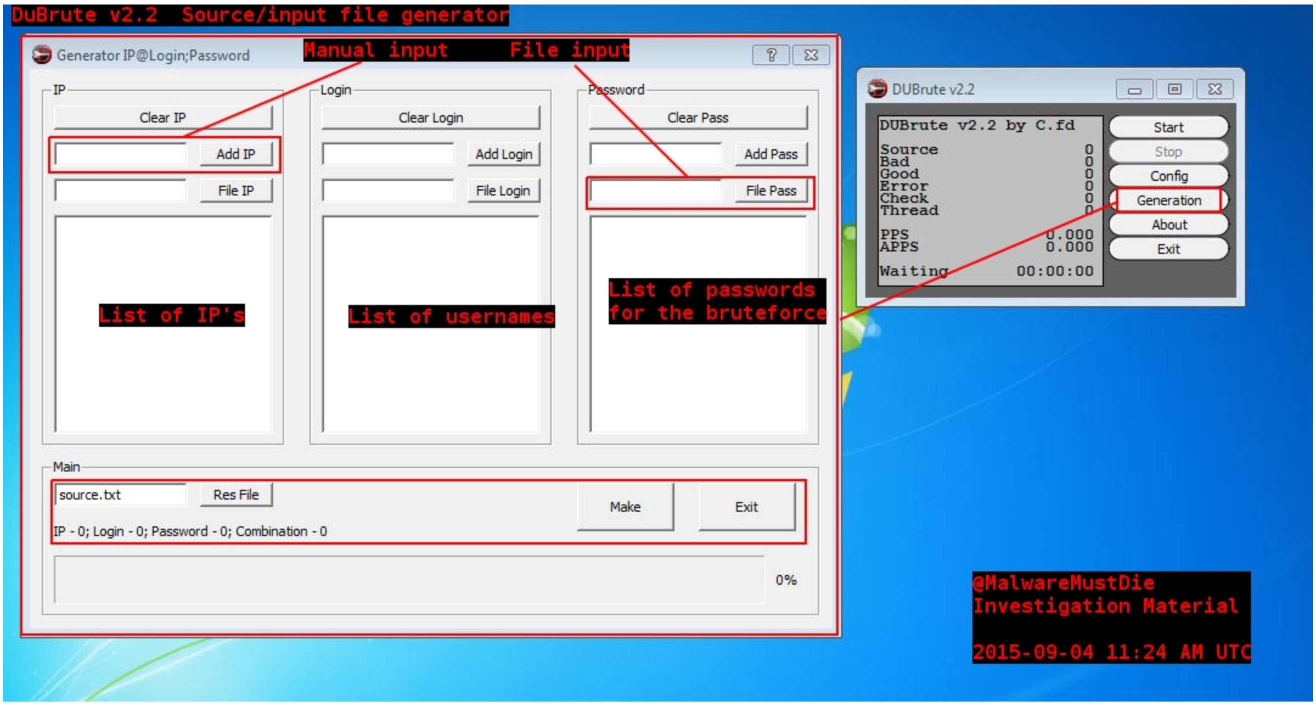
error.txt — 1 AutoSave minute

@MalwareMustDie Investigation Material

2015-09-04 11:21 AM UTC

Additional to the filepaths for the output files you can set different other options like "Thread rate", "Connections", "TimeOut" and "AutoSave".

The next picture is more interesting. This is the source.txt-Generator. With this tool they can create an input file. This input file contains the IP's, logins and passwords.



DuBrute v2.2 Source/input file generator

Generator IP@Login;Password

Manual input

File input

List of IP's

List of usernames

List of passwords for the bruteforce

source.txt

IP - 0; Login - 0; Password - 0; Combination - 0

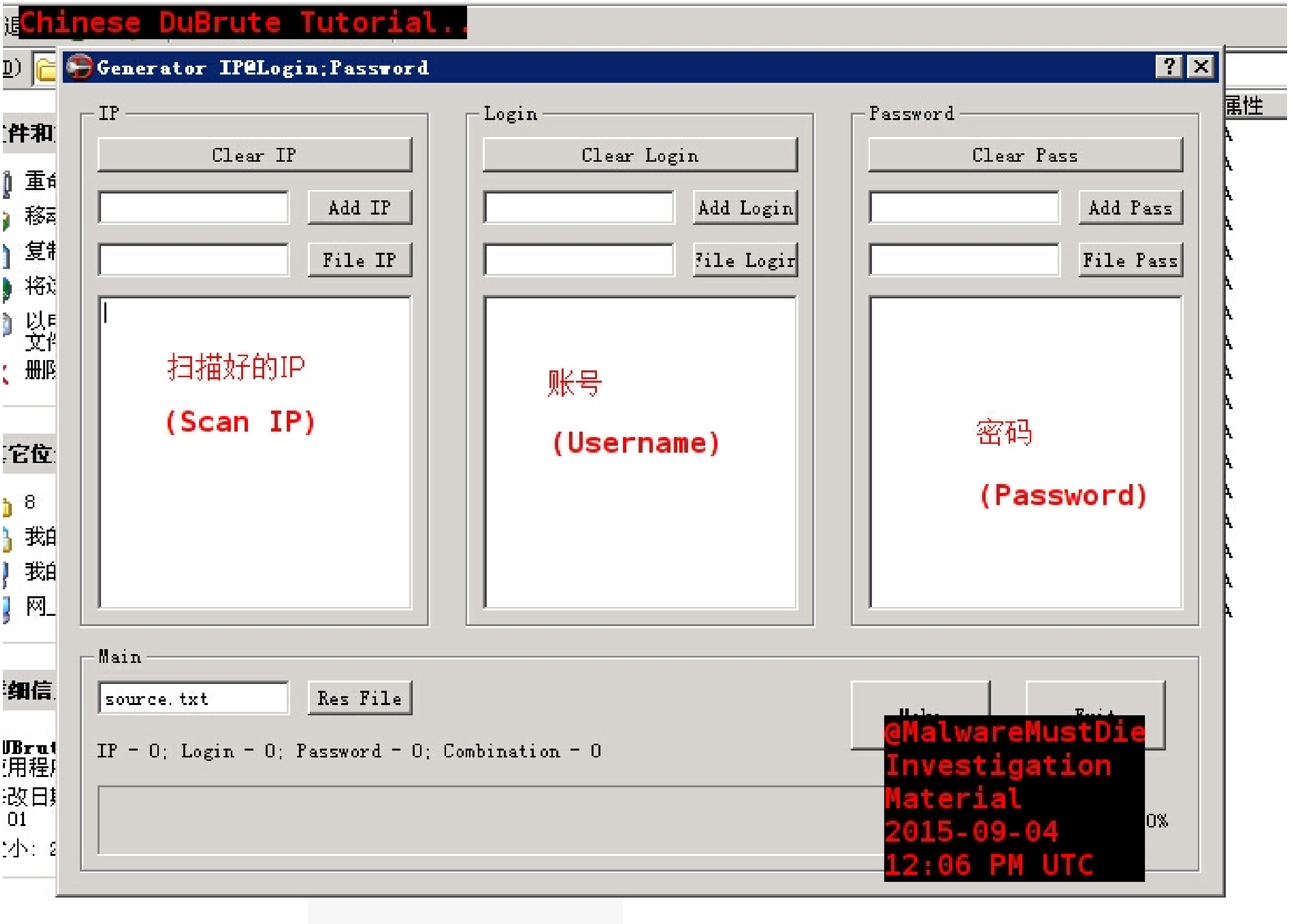
0%

```
DUBrute v2.2 by C.fd

Source          0
Bad             0
Good            0
Error           0
Check           0
Thread          0

PPS         0.000
APPS        0.000

Waiting  00:00:00
```

@MalwareMustDie Investigation Material

2015-09-04 11:24 AM UTC

The chinese seem to have problems with this panel. So they made a little tutorial for each other:



source.txt

IP - 0; Login - 0; Password - 0; Combination - 0

@MalwareMustDie Investigation Material 2015-09-04 12:06 PM UTC

This is the picture that I have found in the panel. I translated the chinese and marked it as Investigation Material. The other .dll's seems to be dependencies for the bruteforce-tool except one:

```
The dependencies for dubrute.exe:

cdbdef73515997355e81a99421c1d721  libeay32.dll
86f1895ae8c5e8b17d99ece768a70732  msvcr71.dll
6c06fb9e1d818fde8d142ee180a65646  ssleay32.dll
abf1f45b4fa467b830f091cf96d82586  QtCore4.dll
21e38b4a1f28bdef99d1c7dac5dce164  QtGui4.dll
```

If you want to examine your system for this kind of hacking-tool feel free to use my yara-rule for it:

```
rule dubrute : bruteforcer
{
    meta:
        author = "Christian Rebischke (@sh1bumi)"
        date = "2015-09-05"
        description = "Rules for DuBrute Bruteforcer"
        in_the_wild = true
        family = "Hackingtool/Bruteforcer"

    strings:
        $a = "WBrute"
        $b = "error.txt"
        $c = "good.txt"
        $d = "source.txt"
        $e = "bad.txt"
        $f = "Generator IP@Login;Password"

    condition:
        //check for MZ Signature at offset 0
        uint16(0) == 0x5A4D

        and

        //check for dubrute specific strings
        $a and $b and $c and $d and $e and $f
}
```

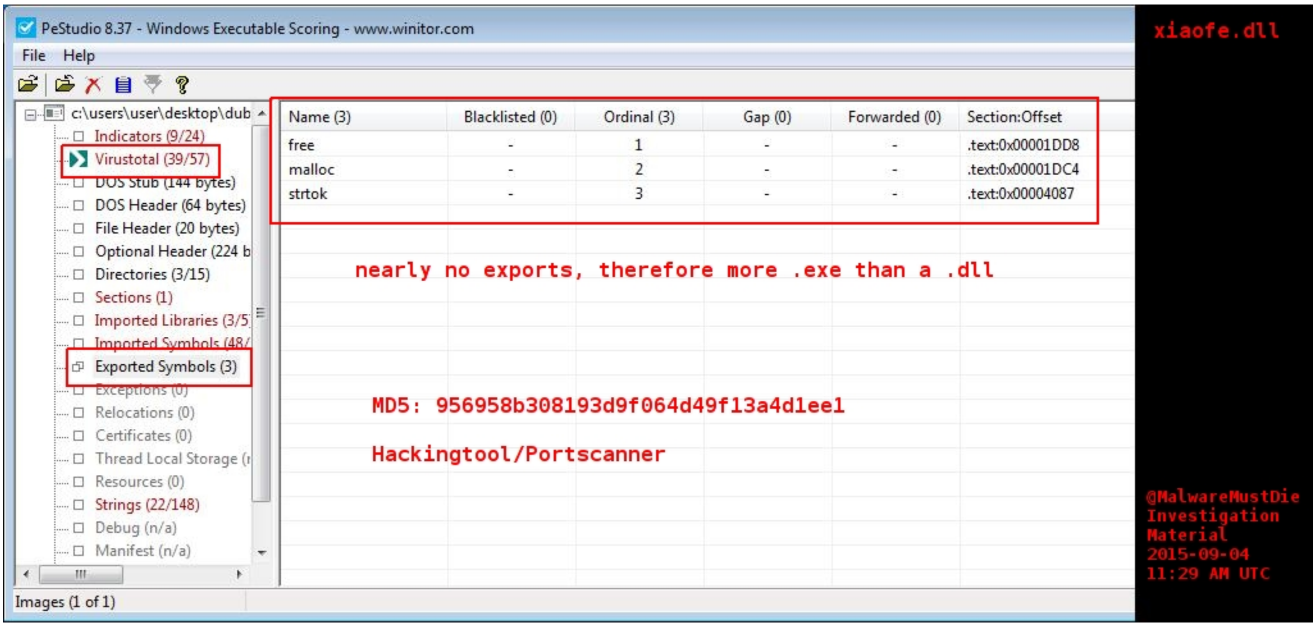
Who else is using DuBrute?

DuBrute is one of the mostly used bruteforcer for windows in the internet. Here are just a few examples who else is using DuBrute:

Russian Carders
Maroccoean Hackercrew
Albanian Hackercrew
some skiddies
AnonSquad
Black Team
Cyberguerilla

and much more..

Let us go to the next binary. This is xiaofe.dll. I analyzed it via PeStudio (thx to Marc Ochsenmeier for this awesome tool).



Name (3)	Blacklisted (0)	Ordinal (3)	Gap (0)	Forwarded (0)	Section:Offset
free	-	1	-	-	.text:0x00001DD8
malloc	-	2	-	-	.text:0x00001DC4
strtok	-	3	-	-	.text:0x00004087

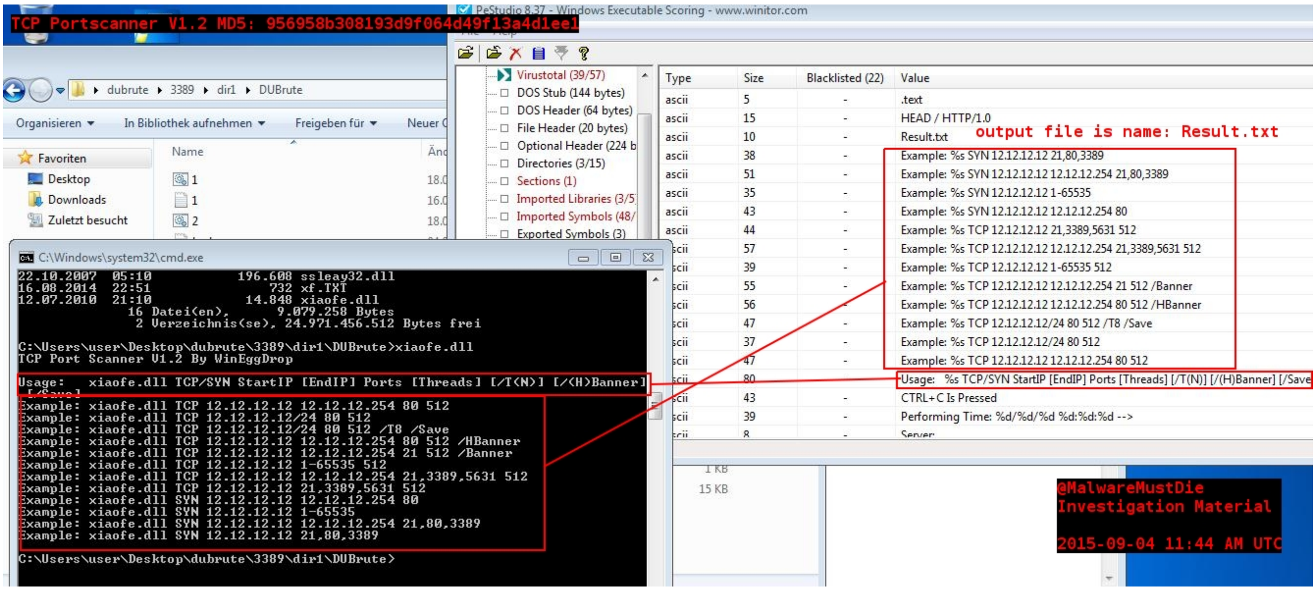
xiaofe.dll

nearly no exports, therefore more .exe than a .dll

MD5: 956958b308193d9f064d49f13a4d1ee1

Hackingtool/Portscanner

@MalwareMustDie
Investigation
Material
2015-09-04
11:29 AM UTC



TCP Portscanner V1.2 MD5: 956958b308193d9f064d49f13a4d1ee1

```
22.10.2007  05:10           196.608 ssleay32.dll
16.08.2014  22:51               732 xf.TXT
12.07.2010  21:10            14.848 xiaofe.dll
              16 Datei(en),      9.079.258 Bytes
               2 Verzeichnis(se), 24.971.456.512 Bytes frei

C:\Users\user\Desktop\dubrute\3389\dir1\DUBrute>xiaofe.dll
TCP Port Scanner V1.2 By WinEggDrop

Usage:   xiaofe.dll TCP/SYN StartIP [EndIP] Ports [Threads] [/T(N)] [/(H)Banner]
 [/Save]
Example: xiaofe.dll TCP 12.12.12.12 12.12.12.254 80 512
Example: xiaofe.dll TCP 12.12.12.12/24 80 512
Example: xiaofe.dll TCP 12.12.12.12/24 80 512 /T8 /Save
Example: xiaofe.dll TCP 12.12.12.12 12.12.12.254 80 512 /HBanner
Example: xiaofe.dll TCP 12.12.12.12 12.12.12.254 21 512 /Banner
Example: xiaofe.dll TCP 12.12.12.12 1-65535 512
Example: xiaofe.dll TCP 12.12.12.12 12.12.12.254 21,3389,5631 512
Example: xiaofe.dll TCP 12.12.12.12 21,3389,5631 512
Example: xiaofe.dll SYN 12.12.12.12 12.12.12.254 80
Example: xiaofe.dll SYN 12.12.12.12 1-65535
Example: xiaofe.dll SYN 12.12.12.12 12.12.12.254 21,80,3389
Example: xiaofe.dll SYN 12.12.12.12 21,80,3389

C:\Users\user\Desktop\dubrute\3389\dir1\DUBrute>
```

Type	Size	Blacklisted (22)	Value
ascii	5	-	.text
ascii	15	-	HEAD / HTTP/1.0
ascii	10	-	Result.txt
ascii	38	-	Example: %s SYN 12.12.12.12 21,80,3389
ascii	51	-	Example: %s SYN 12.12.12.12 12.12.12.254 21,80,3389
ascii	35	-	Example: %s SYN 12.12.12.12 1-65535
ascii	43	-	Example: %s SYN 12.12.12.12 12.12.12.254 80
ascii	44	-	Example: %s TCP 12.12.12.12 21,3389,5631 512
scii	57	-	Example: %s TCP 12.12.12.12 12.12.12.254 21,3389,5631 512
scii	39	-	Example: %s TCP 12.12.12.12 1-65535 512
scii	55	-	Example: %s TCP 12.12.12.12 12.12.12.254 21 512 /Banner
scii	56	-	Example: %s TCP 12.12.12.12 12.12.12.254 80 512 /HBanner
scii	47	-	Example: %s TCP 12.12.12.12/24 80 512 /T8 /Save
scii	37	-	Example: %s TCP 12.12.12.12/24 80 512
scii	47	-	Example: %s TCP 12.12.12.12 12.12.12.254 80 512
scii	80		Usage: %s TCP/SYN StartIP [EndIP] Ports [Threads] [/T(N)] [/(H)Banner] [/Save
scii	43	-	CTRL+C Is Pressed
scii	39	-	Performing Time: %d/%d/%d %d:%d:%d -->
scii	8	-	Server:

output file is name: Result.txt

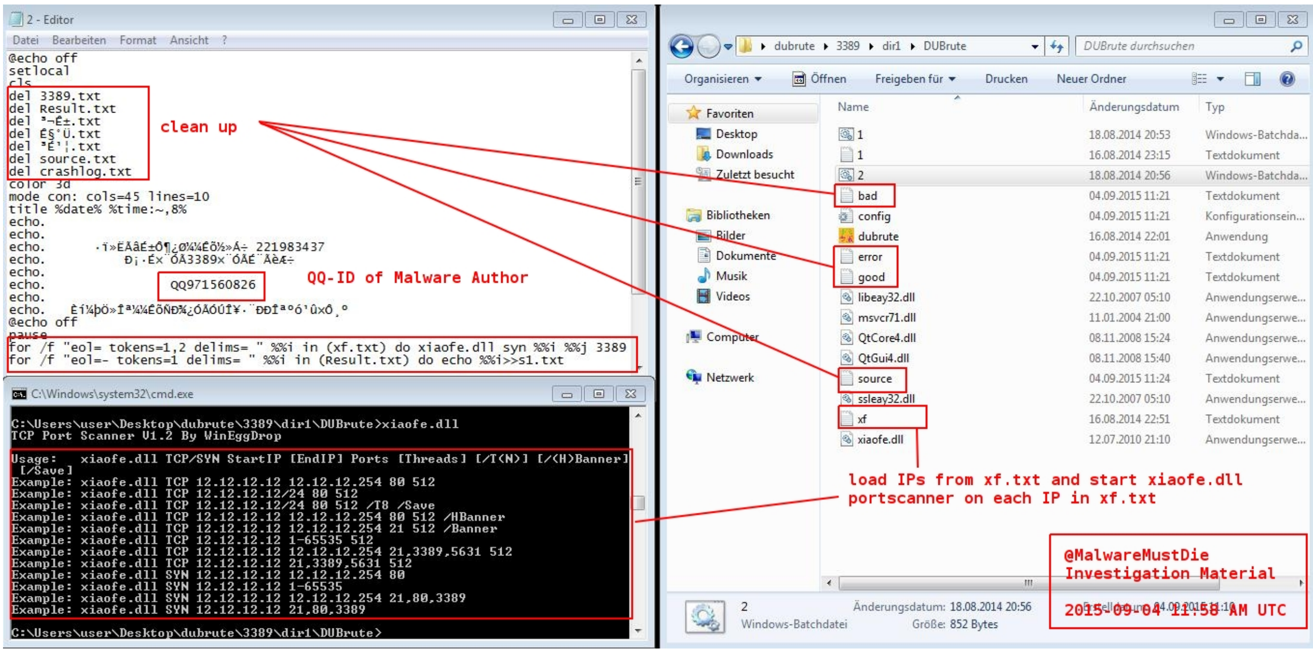
@MalwareMustDie
Investigation Material

2015-09-04 11:44 AM UTC

It's pretty clear a portscanner. I think this doesn't need more explaination. Here is a yara-rule for this tool, too:

```
rule wineggdrop : portscanner
{
    meta:
        author = "Christian Rebischke (@sh1bumi)"
        date = "2015-09-05"
        description = "Rules for TCP Portscanner VX.X by WinEggDrop
        in_the_wild = true
        family = "Hackingtool/Portscanner"

    strings:
        $a = { 54 43 50 20 50 6f 72 74 20 53 63 61 6e 6e 65 72
               20 56 3? 2e 3? 20 42 79 20 57 69 6e 45 67 67 44
               72 6f 70 0a }
        $b = "Result.txt"
        $c = "Usage:   %s TCP/SYN StartIP [EndIP] Ports [Threads] [

    condition:
        //check for MZ Signature at offset 0
        uint16(0) == 0x5A4D

        and

        //check for wineggdrop specific strings
        $a and $b and $c
}
```

After I analyzed the binaries I went further to the batch files. I didn't expect something special, but I was nicely surprised:



```
@echo off
setlocal
cls
del 3389.txt
del Result.txt
del ³¬Ê±.txt
del Ê§°Ü.txt
del ³É¹¦.txt
del source.txt
del crashlog.txt
color 3d
mode con: cols=45 lines=10
title %date% %time:~,8%
echo.
echo.
echo.      ·ï»ÈÄâÉ±Ô¶¿Ø¼¼ÊõÌ½»Á÷ 221983437
echo.          Ð¡·Ê×¨ÓÃ3389×¨ÓÃÉ¨ÃèÆ÷
echo.
echo.                   QQ971560826
echo.
echo.   Èí¼þÖ»Îª¼¼ÊõÑÐ¾¿ÓÃÓÚÎ¥·¨ÐÐÎªºó¹ûÐÔ¸º
@echo off
pause
for /f "eol= tokens=1,2 delims= " %%i in (xf.txt) do xiaofe.dll syn %%i %%j 3389
for /f "eol=- tokens=1 delims= " %%i in (Result.txt) do echo %%i>>s1.txt
```

clean up

QQ-ID of Malware Author

```
C:\Users\user\Desktop\dubrute\3389\dir1\DUBrute>xiaofe.dll
TCP Port Scanner V1.2 By WinEggDrop

Usage:   xiaofe.dll TCP/SYN StartIP [EndIP] Ports [Threads] [/T(N)] [/(H)Banner]
 [/Save]
Example: xiaofe.dll TCP 12.12.12.12 12.12.12.254 80 512
Example: xiaofe.dll TCP 12.12.12.12/24 80 512
Example: xiaofe.dll TCP 12.12.12.12/24 80 512 /T8 /Save
Example: xiaofe.dll TCP 12.12.12.12 12.12.12.254 80 512 /HBanner
Example: xiaofe.dll TCP 12.12.12.12 12.12.12.254 21 512 /Banner
Example: xiaofe.dll TCP 12.12.12.12 1-65535 512
Example: xiaofe.dll TCP 12.12.12.12 12.12.12.254 21,3389,5631 512
Example: xiaofe.dll TCP 12.12.12.12 21,3389,5631 512
Example: xiaofe.dll SYN 12.12.12.12 12.12.12.254 80
Example: xiaofe.dll SYN 12.12.12.12 1-65535
Example: xiaofe.dll SYN 12.12.12.12 12.12.12.254 21,80,3389
Example: xiaofe.dll SYN 12.12.12.12 21,80,3389

C:\Users\user\Desktop\dubrute\3389\dir1\DUBrute>
```

Name	Änderungsdatum	Typ
1	18.08.2014 20:53	Windows-Batchda...
1	16.08.2014 23:15	Textdokument
2	18.08.2014 20:56	Windows-Batchda...
bad	04.09.2015 11:21	Textdokument
config	04.09.2015 11:21	Konfigurationsein...
dubrute	16.08.2014 22:01	Anwendung
error	04.09.2015 11:21	Textdokument
good	04.09.2015 11:21	Textdokument
libeay32.dll	22.10.2007 05:10	Anwendungserwe...
msvcr71.dll	11.01.2004 21:00	Anwendungserwe...
QtCore4.dll	08.11.2008 15:24	Anwendungserwe...
QtGui4.dll	08.11.2008 15:40	Anwendungserwe...
source	04.09.2015 11:24	Textdokument
ssleay32.dll	22.10.2007 05:10	Anwendungserwe...
xf	16.08.2014 22:51	Textdokument
xiaofe.dll	12.07.2010 21:10	Anwendungserwe...

2 Windows-Batchdatei Änderungsdatum: 18.08.2014 20:56 Größe: 852 Bytes

load IPs from xf.txt and start xiaofe.dll portscanner on each IP in xf.txt

@MalwareMustDie
Investigation Material
2015-09-04 11:58 AM UTC

2.bat seems to be a clean-up script. It cleans up the working directory outputs a chinese header (that I can't read due to encoding problems. When you have an idea how to fix this feel free to email me) and initiates the portscan on all IP's in the xf.txt file. The interesting part is the QQ ID. (QQ is a famous chinese messenger and mailhoster). This QQ ID leads us to an old known suspect, who is pretty famous for the Linux/Xor.DDoS story:

Here are the virustotal-links to the two binaries:

dubrute.exe
xiaofe.dll

thanks to my fellow team mates and our supporters for the great assistance.

Visit http://blog.malwaremustdie.org for more information.

shibumi
sh1bumi

nullday is the webspace of Christian Rebischke and powered by jekyll without Javascript, PHP, Cookies and MySQL. Last change: 2015-09-06