Lab 6: Tripwire, Log and Packet Analysis

Details

Tripwire is a Host IDS which can monitor a host system for signs of intrusion. It is classified as a File Integrity Monitor, and works by comparing the current state of a system to a baseline or snapshot of the system.

If you have not already done so, setup your Splunk infrastructure from the previous lab.

Tripwire setup

On-line video demo of the tripwire lab:
https://youtu.be/MvMnwDeXvZo

1 Run UBUNTU (User name: napier, Password: napier123). Within the virtual image, run a Terminal and determine its IP address using ifconfig.

2 Create Tripwire Filesystem Snapshot

Go to the **/etc/tripwire** folder, and view the **twpol.txt** file.

Next run the following commands to create the tripwire policy file and the tripwire database (use **napier123** as the site and local encryption keys):

```
sudo twadmin --create-polfile --cfgfile ./tw.cfg --site-keyfile  ./site.key
./twpol.txt

sudo tripwire --init --cfgfile /etc/tripwire/tw.cfg --polfile
/etc/tripwire/tw.pol --site-keyfile /etc/tripwire/site.key --local-keyfile
/etc/tripwire/ubuntu-local.key
```

This has created the tripwire integrity database, which contains the filesystem snapshot or baseline. This will then be used as a reference point for all file integrity verifications.

3 Run a Tripwire Filesystem Check

Go to the **/etc/passwd** file and change the owner to **fred** with a command such as:

```
/etc$  sudo chown fred:fred passwd
```

Next go to the **/tmp** folder and change the ownership of one of the files.
Next run a filesystem integrity check with Tripwire, sending it to a file and edit the file, with:

```
sudo tripwire --check > check.txt
```

```
nano check.txt
```

- ☞ What do you observe from the results?

- ☞ Why does Tripwire not report changes to the file in the temp directory? (check the twpol.txt)

4 Create A New Tripwire Rule

Go to your home directory and create a new folder, and a file within, using commands such as:

```
cd ~
mkdir pizza
cd pizza
touch bbq.pizza
```

Run another filesystem integrity check with Tripwire, with:

```
sudo tripwire --check > check.txt
nano check.txt
```

- ☞ Does Tripwire report the creation of the new directory?

- ☞ Why?

Go back to the tripwire directory, and add a rule to the policy file for Tripwire (**`twpol.txt`**), to monitor the pizza directory, using SEC_CRIT.

- ☞ Tripwire Rule:

Next run the following commands to recreate the tripwire policy file and the tripwire database (use **napier123** as the site and local encryption keys):

```
sudo twadmin --create-polfile --cfgfile ./tw.cfg --site-keyfile  ./site.key
./twpol.txt

sudo tripwire --init --cfgfile /etc/tripwire/tw.cfg --polfile
/etc/tripwire/tw.pol --site-keyfile /etc/tripwire/site.key --local-keyfile
/etc/tripwire/ubuntu-local.key
```

Run another filesystem integrity check with Tripwire, with:

```
sudo tripwire --check > check.txt ; vi check.txt
```

☞ Does Tripwire report the creation of the new directory?
☞ Why?

Tripwire Syslog

For this next step, you will need to install the Syslog client on Ubuntu with:

```
sudo apt-get install syslogd
```

Then, edit the syslog config file (/etc/syslog.conf) and add the following line:

```
*.*          @192.168.y.8
```

(Replacing 192.168.y.8 with the IP of your Windows 2008 Server.)

In addition to that, set the variable SYSLOGREPORTING to "true" in /etc/tripwire/twcfg.txt.

Next, restart the syslog daemon with:

```
sudo /etc/init.d/syslog restart
```

Your Splunk installation should already be listening on UDP port 514, reflecting on the previous lab. If it is not, make sure you enable that through the web interface.

From Ubuntu, send a test syslog message with:

```
logger "test message from syslog client"
```

Splunk should be receiving this message. Similarly, running:

```
sudo tripwire --check
```

… should send a syslog entry to Splunk.

Now enable VNC to be forwarded through your firewall from DMZ to Private, and connect from the Windows 2003 computer to Ubuntu using the VNC Viewer. Next make a modification to a file that Tripwire is monitoring.

Does Trip detect the change? Yes/No

Red v Blue

Now forward the VNC connection from outside the firewall, so that someone on the public network can log into your Ubuntu machine. Next run vncviewer from Kali and make a connection to the Ubuntu machine.

Once you have tested this, get a neighbour to access your Ubuntu machine and get them to make a modification on one of the files that Trip monitors.

Can you detect their action?

Enumeration – Windows WMIC

The Windows Management Instrumentation Command-line (WMIC) is used to gather information about a computer, and is used by Splunk to grab information.

To assist with this part of the lab and the following demo can be used:

http://bit.ly/fyyF0B

On your the target hosts details with the following:

wmic.exe /node:w.x.y.z CPU list brief
wmic.exe /node:w.x.y.z NIC list brief
wmic.exe /node:w.x.y.z OS list brief
wmic.exe /node:w.x.y.z SHARE list brief

The output should be similar to Figure 1.

What is the MAC address of the windows host? Which Shares are found on the host? Outline some other details: What other options are available in WMIC?

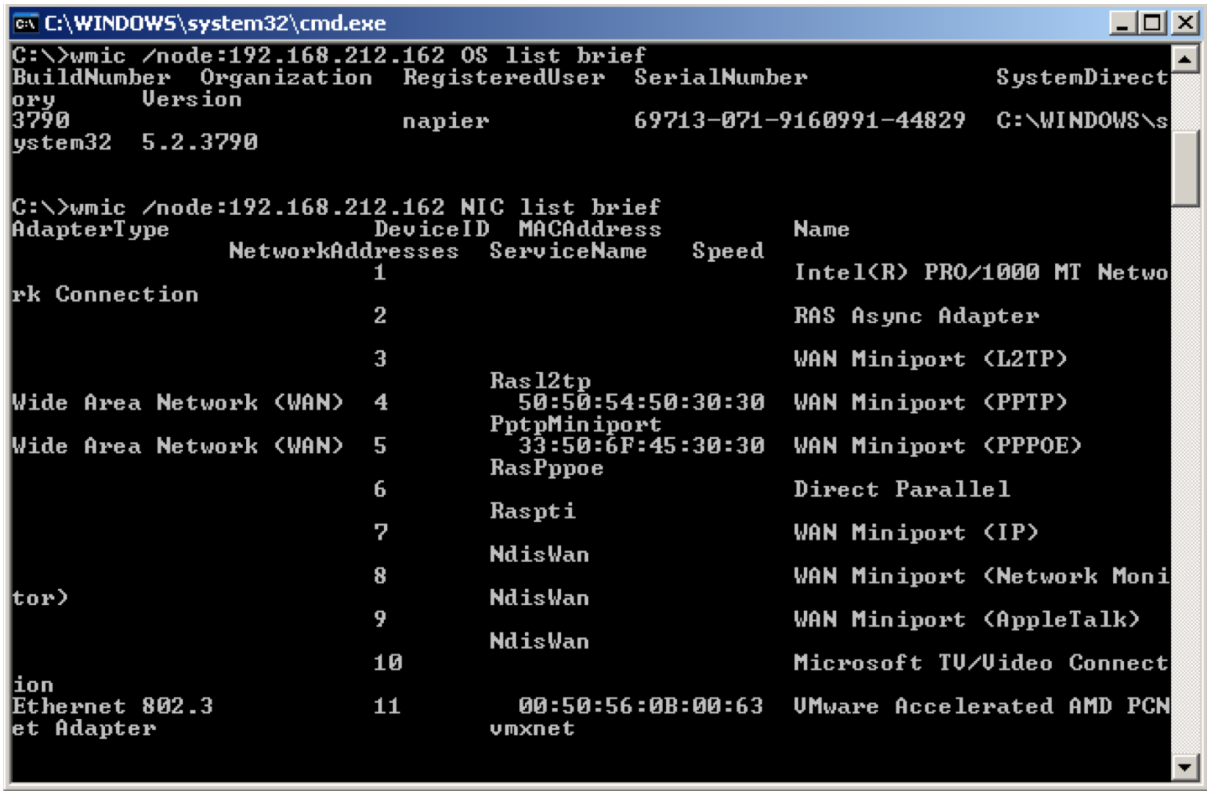


Figure 1

Next complete the following table for your Windows instances:

Description	Windows 2003	Windows 2008
CPU		
Bios Manufacturer		
Disk interface type		
Operating system type		
List two Environment variables		
Workgroup (or Domain)		

Analysing Web logs

Within Ubuntu install Logstalgia from:

```
sudo apt-get install  logstalgia
```

You can then analyse one of your Apache log files. Go into:
`/var/log/apache2`

and then run:

```
/var/log/apache2$ logstalgia access.log
```

What do you observe from the visualisation?

Download the following file and analyse it in logstalgia:

http://asecuritysite.com/log/log01.zip

Log parsing

In this example we will create a Python script to analyse the log file downloaded in the previous section. On Ubuntu (or Kali) create a Python file which will use a regular expression to parse an Apache Web log:

```
regex = '([(\d\.)]+) - - \[(.*?)\] "(.*?)" (\d+)'

import sys
import re

if (len(sys.argv)>1):
   file=sys.argv[1]

count200=0

for line in open(file):
   try:
     inp=re.match(regex, line).groups()
     status=inp[3]
     if (status=="200"): count200=count200+1
   except:
     continue
print count200
```

Run the Python program with:

```
python log.py log01.log
```

where log01.log is the name of your log file, and log.py is the name of your Python program.

Run the program, and observe the output:

Now modify the program so that it picks-off the other HTTP response codes that are contained in the file.

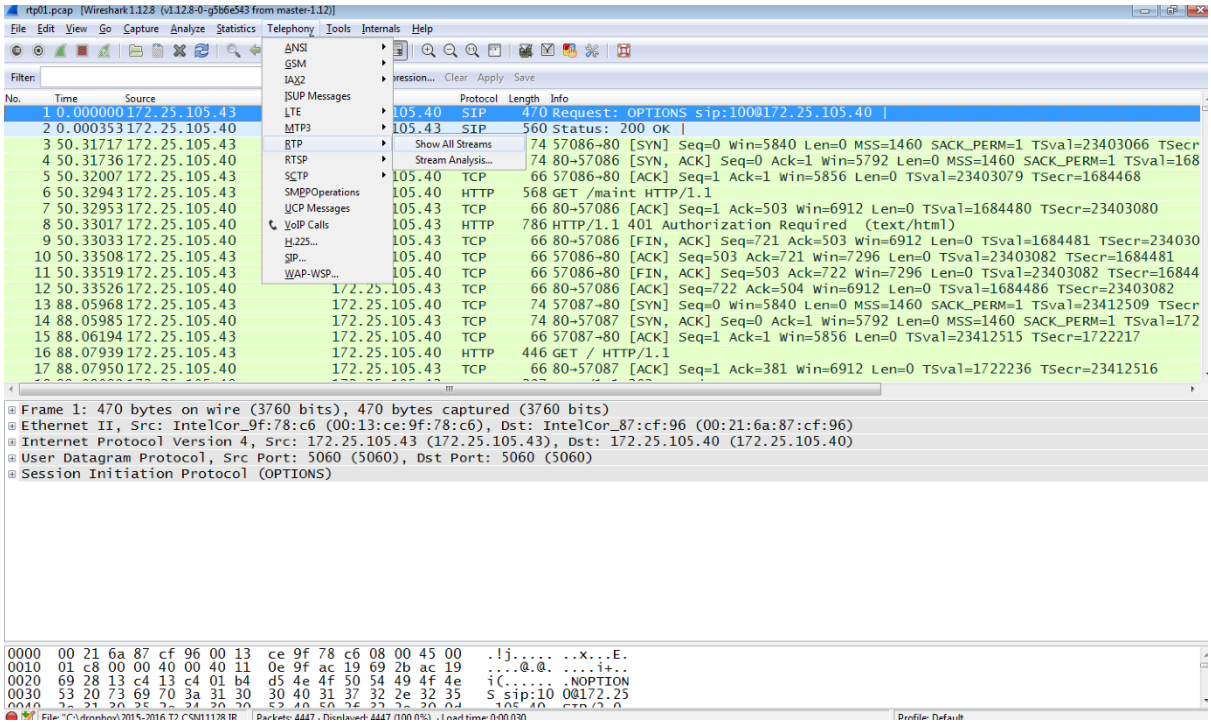
Voice analysis

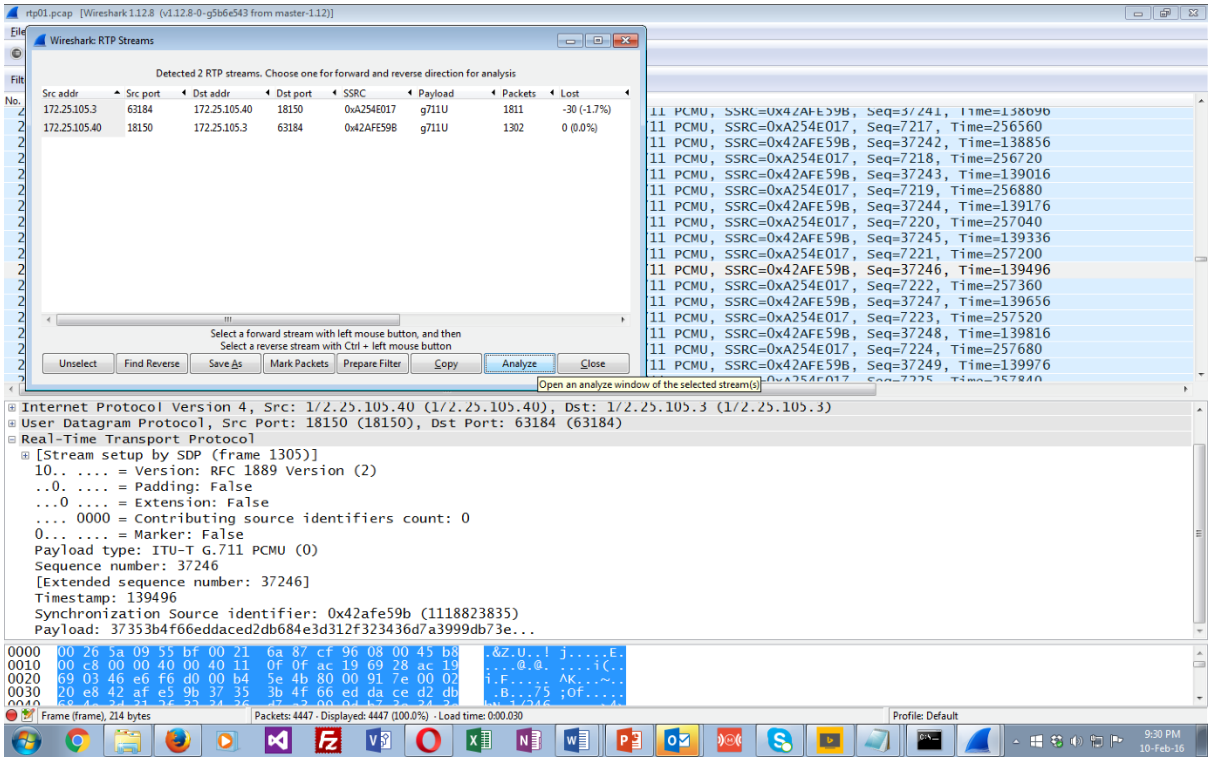
Download the file:

http://asecuritysite.com/log/rtp01.zip

What is the SIP address that it being contacted?

Go to Telephony->RTP-Show All Streams. Pick a stream and Analyse. Next output the stream to an AU file (audio file). Analyse the voice file, and determine the password:




rtp01.pcap [Wireshark 1.12.8 (v1.12.8-0-g5b6e543 from master-1.12)]
Wireshark: RTP Streams
Detected 2 RTP streams. Choose one for forward and reverse direction for analysis
Src addr | Src port | Dst addr | Dst port | SSRC | Payload | Packets | Lost
172.25.105.3 | 63184 | 172.25.105.40 | 18150 | 0xA254E017 | g711U | 1811 | -30 (-1.7%)
172.25.105.40 | 18150 | 172.25.105.3 | 63184 | 0x42AFE59B | g711U | 1302 | 0 (0.0%)
Select a forward stream with left mouse button, and then
Select a reverse stream with Ctrl + left mouse button
Unselect
Find Reverse
Save As
Mark Packets
Prepare Filter
Copy
Analyze
Close
Open an analyze window of the selected stream(s)
11 PCMU, SSRC=0x42AFE59B, Seq=37241, Time=138696
11 PCMU, SSRC=0xA254E017, Seq=7217, Time=256560
11 PCMU, SSRC=0x42AFE59B, Seq=37242, Time=138856
11 PCMU, SSRC=0xA254E017, Seq=7218, Time=256720
11 PCMU, SSRC=0x42AFE59B, Seq=37243, Time=139016
11 PCMU, SSRC=0xA254E017, Seq=7219, Time=256880
11 PCMU, SSRC=0x42AFE59B, Seq=37244, Time=139176
11 PCMU, SSRC=0xA254E017, Seq=7220, Time=257040
11 PCMU, SSRC=0x42AFE59B, Seq=37245, Time=139336
11 PCMU, SSRC=0xA254E017, Seq=7221, Time=257200
11 PCMU, SSRC=0x42AFE59B, Seq=37246, Time=139496
11 PCMU, SSRC=0xA254E017, Seq=7222, Time=257360
11 PCMU, SSRC=0x42AFE59B, Seq=37247, Time=139656
11 PCMU, SSRC=0xA254E017, Seq=7223, Time=257520
11 PCMU, SSRC=0x42AFE59B, Seq=37248, Time=139816
11 PCMU, SSRC=0xA254E017, Seq=7224, Time=257680
11 PCMU, SSRC=0x42AFE59B, Seq=37249, Time=139976
Internet Protocol Version 4, Src: 172.25.105.40 (172.25.105.40), Dst: 172.25.105.3 (172.25.105.3)
User Datagram Protocol, Src Port: 18150 (18150), Dst Port: 63184 (63184)
Real-Time Transport Protocol
[Stream setup by SDP (frame 1305)]
10.. .... = Version: RFC 1889 Version (2)
..0. .... = Padding: False
...0 .... = Extension: False
.... 0000 = Contributing source identifiers count: 0
0... .... = Marker: False
Payload type: ITU-T G.711 PCMU (0)
Sequence number: 37246
[Extended sequence number: 37246]
Timestamp: 139496
Synchronization Source identifier: 0x42afe59b (1118823835)
Payload: 37353b4f66eddaced2db684e3d312f323436d7a3999db73e...
0000 00 26 5a 09 55 bf 00 21 6a 87 cf 96 08 00 45 b8 .&Z.U..! j.....E.
0010 00 c8 00 00 40 00 40 11 0f 0f ac 19 69 28 ac 19 ....@.@. ....i(..
0020 69 03 46 e6 f6 d0 00 b4 5e 4b 80 00 91 7e 00 02 i.F..... ^K...~..
0030 20 e8 42 af e5 9b 37 35 3b 4f 66 ed da ce d2 db .B...75 ;Of.....
Frame (frame), 214 bytes
Packets: 4447 · Displayed: 4447 (100.0%) · Load time: 0:00.030
Profile: Default
9:30 PM
10-Feb-16