HARDENING
IDS/IPS Technologies

Salvatore Lentini
Davide Antonino Vincenzo Micale

Hardening Three

University of Catania

29 May 2017
Summary

• **Problems**
  • Arp spoofing
  • Video with Netcat Backdoor
  • Ping out of network
  • Attempt access and Http BruteForce

• **Tools:**
  • Snort IDS/IPS
  • Suricata IDS/IPS
  • PulledPork
  • Barnyard2
  • Base

• **Some theoretical concepts**
First Scenario
Ping out of network

We are an Network Administrator and don’t want that someone to ping from out of network our hosts into our network (HOME_NET)

So, we want know who execute the ping

```
root@kali:~# ping 192.168.248.2
PING 192.168.248.2 (192.168.248.2) 56(84) bytes of data.
64 bytes from 192.168.248.2: icmp_seq=1 ttl=63 time=1.14 ms
64 bytes from 192.168.248.2: icmp_seq=2 ttl=63 time=1.36 ms
64 bytes from 192.168.248.2: icmp_seq=3 ttl=63 time=1.27 ms
64 bytes from 192.168.248.2: icmp_seq=4 ttl=63 time=1.04 ms

--- 192.168.248.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 1.047/1.207/1.367/0.121 ms
```
Ping Detected

```
sudo nano snort -c /etc/snort/snort.conf -A console -i ens38
```
Ping out of Network Prevention

--- 192.168.248.2 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6009ms
rtt min/avg/max/mdev = 0.889/2.505/5.687/1.542 ms
Ping out of Network Prevention

```
sudo snort -c /etc/snort/snort.conf -i ens38:ens39 -Q -A console -q
```

University of Catania

Hardening Three
Possible problem

Did you have this error?

Did you have this error?

Don't worry. This is an IPS then it want more memory! You need add more memory your virtual machine.
Video with Netcat Backdoor

- Seems a video, but......

DEMO
“It’s dangerous, so dangerous”

- The file is a **Self Extracting WinRar (SFX)** archive containing:
  - Video
  - Netcat
  - Caller.vbs: on execute, run Netcat on a specific port opening the shell calling the attacker. Open the video
Creation of the archive

- “Compression method”: select “Store”
- “Archiving option”: check “Create SFX archive”
- On tab “Advanced” → “SFX options”
  - On tab “General” → “Path to Extract”: “C:\virus” and tick “Absolute path”
  - On tab “Setup” → “Setup program/Run after extraction”: “caller.vbs”
  - On tab “Modes” → “Silent mode”: tick “Hide all”
  - On tab “Update” → “Overwrite mode”: tick “Skip existing files”
  - On tab “Text and icon” → “Load SFX icon from the file”: select a .ico file containing the video icon.
Extension Spoofing

• Use the Unicode “U+202E: Right-To-Left Override” character.
Attacker side

- Run netcat on listen mode on the specific port
- Wait until the victim play the fake video
Detection and block rules

- Rules to detect prompt commands on Windows:
  - drop tcp any any -> any any (msg:"Dir Command - Possible Remote Shell"; content:"dir"; sid:10000001;)
  - drop tcp any any -> any any (msg:"Dir Command - Possible Remote Shell"; content:"DIR"; sid:10000002;)
  - drop tcp any any -> any any (msg:"Dir Command - Possible Remote Shell"; content:"<DIR>"; sid:10000003;)
  - alert tcp any any -> any any (msg:"Cd Command - Possible Remote Shell"; content:"cd "; sid:10000004;)
  - alert tcp any any -> any any (msg:"Cd Command - Possible Remote Shell"; content:"Cd "; sid:10000005;)
  - alert tcp any any -> any any (msg:"Cd Command - Possible Remote Shell"; content:"cD "; sid:10000006;)
  - alert tcp any any -> any any (msg:"Cd Command - Possible Remote Shell"; content:"CD "; sid:10000007;)
  - alert tcp any any -> any any (msg:"MKDIR Command - Possible Remote Shell"; content:"mkdir "; sid:10000008;)
  - alert tcp any any -> any any (msg:"Del Command - Possible Remote Shell"; content:"del "; sid:10000009;)
  - alert tcp any any -> any any (msg:"Erase Command - Possible Remote Shell"; content:"erase "; sid:10000010;)
  - alert tcp any any -> any any (msg:"Format Command - Possible Remote Shell"; content:"format "; sid:10000011;)

- Rule to block on Ubuntu in IPS mode: “drop tcp any any -> any any (msg:"Startup Shell - Possible Remote Shell"; content:"Microsoft Windows [Versione 6.1.7600]"; sid:10000000;)”
Second Scenario
Attempt of login

Now, we want to know who attempted login on our web server.
In our rule we can use this one to detect traffic.
Attempt access
Detected

Detected Attempt of Login
Http Bruteforce

```
root@kali:~# nmap --script http-brute -p 80 192.168.248.2
Starting Nmap 7.31 ( https://nmap.org ) at 2017-05-16 13:57 EDT
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled.
Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 192.168.248.2
Host is up (0.0012s latency).
PORT       STATE       SERVICE
80/tcp      open       http
| http-brute:
|   Accounts:
|   | web:123456 - Valid credentials
|   | Statistics: Performed 45011 guesses in 52 seconds, average tps: 890.6

Nmap done: 1 IP address (1 host up) scanned in 51.84 seconds
root@kali:~#
```

Our credentials

University of Catania

Hardening Three
Http Brute force Detection

File: /etc/snort/rules/custom.rules

Alert icmp !HOME_NET any -> $HOME_NET any (msg:"someone is pinging out of network"; sid:100001;)

Alert tcp $HOME_NET any -> 192.168.248.2 80 (msg:"attempt login"; content:"Authorization"; sid:100002;)

Hardening Three
Http Bruteforce Prevention

Starting Nmap 7.31 (https://nmap.org) at 2017-05-17 01:08 EDT
mass_dns: warning: Unable to open /etc/resolv.conf. Try using --system-dns or specify valid servers with --dns-servers
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled. Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 192.168.248.2
Host is up (0.0015s latency).
PORT   STATE SERVICE
80/tcp open  http
 | http-brute:
 |   Accounts: No valid accounts found
 | Statistics: Performed 10 guesses in 1 seconds, average tps: 10.0
 |   ERROR: Too many retries, aborted ...

Nmap done: 1 IP address (1 host up) scanned in 0.89 seconds
root@kali:~#
Http Bruteforce

Snort as IPS with nft

It's an inline modality on Linux using Netfilter

University of Catania

Hardening Three
ARP Spoofing

“ALL YOUR PACKETS ARE BELONG TO US”

DEMO
ARP (Address Resolution Protocol): bind the IP address to the MAC address of the devices inside the LAN.
Ping

- Goal: A ping B
- ARP, Request who-has 192.168.18.3 tell 192.168.18.1
- ARP, Reply 192.168.18.3 is-at 00:0C:29:6C:54:28
ARP Cache

- ARP Cache: table with the associations IP → MAC

<table>
<thead>
<tr>
<th>IP</th>
<th>MAC</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.168.18.3</td>
<td>00:0C:29:6C:54:28</td>
</tr>
</tbody>
</table>

- The associations will be deleted periodically
- The device will accept any ARP reply in ANY time!
- No authenticity control of the association
MITM with ARP Poisoning

• The attacker can sniff the victim packets
• The attacker send on the LAN ARP Reply for each device
  • “ARP, Reply X.X.X.X is-at Y:Y:Y:Y” where Y:Y:Y:Y is the attacker MAC address
  • Let’s $C$ the attacker and $B$ the victim
    • ARP, Reply 192.168.18.3 is-at 00:0C:29:B0:C9:71
    • ARP, Reply 192.168.18.1 is-at 00:0C:29:B0:C9:71
ARP Cache after the poisoning

- **A table**

<table>
<thead>
<tr>
<th>IP</th>
<th>MAC</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.168.18.3</td>
<td>00:0C:29:B0:C9:71</td>
</tr>
</tbody>
</table>

- **B table**

<table>
<thead>
<tr>
<th>IP</th>
<th>MAC</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.168.18.1</td>
<td>00:0C:29:B0:C9:71</td>
</tr>
</tbody>
</table>
ARP Cache after the poisoning - Continue

- The attacker need to ARP spoofing very often
- Every packet of the victim will be seen by the attacker
Ettercap

• “Ettercap is a comprehensive suite for man in the middle attacks.”

• Man in the middle could be accomplished using the -M arp mode

• The attacker must activate the promiscous mode

```
root@kali:~
File   Modifica   Visualizza   Cerca   Terminale   Auto
root@kali:~# sysctl -w net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1
```

• “ettercap -T -i IF -M arp /IP_VICTIM///”
Run Ettercap

1) Scan the devices of the LAN
2) ARP Poisoning
Sniffing web navigation

- The attacker can sniff the victim web navigation
Password sniffing

• The victim goes to a HTTP web page

• The attacker catches the Username and Password

```plaintext
POST /portalestudente/j_security_check HTTP/1.1

j_username=MDLDVN89T14F123F&j_password=1111111&Submit=Entra
```
1) Promiscuous mode: every packet will be read by the network interface (whatever it is the destinatary of the packet)

```
davide@ubuntu:~$ sudo sysctl -w net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1
```

2) Masquerade: every packet transmitted outside through the specific network interface will have the IP Sender set to the gateway IP

```
davide@ubuntu:~$ sudo iptables -t nat -A POSTROUTING -o ens33 -j MASQUERADE
```
IDS and the ARP spoofing

- Suricata can’t do anything to detect an ARP spoofing
- Snort can detect using the ARP Spoof Preprocessor
Snort detection ARP spoofing

- Modify `/etc/snort/snort.conf`
  - Uncomment “preprocessor arpspoof”
  - For each device of the LAN, add “preprocessor arpspoof_detect_host: IP MAC”
  - Uncomment “include $PREPROC_RULE_PATH/preprocessor.rules”
  - Download the preprocessor rules in the Snort website and save it on the file “/etc/snort/preproc_rules/preprocessor.rules”
Installation and Configuration
Virtual Network

- VMware Workstation
- No DHCP

192.168.232.0/24
192.168.232.1
192.168.232.2
192.168.232.0/24

INTERNET
(NAT)

192.168.248.0/24
192.168.248.1
192.168.248.2
Create Networks

Open Virtual Network Editor:

• **Add 2 networks**
  
  **vmnet11**
  
  • Host-Only
  
  • 192.168.232.0
  
  • 255.255.255.0

  **vmnet12**
  
  • Host-Only
  
  • 192.168.248.0
  
  • 255.255.255.0

**NO DHCP!**
Router (Ubuntu)

Add 2 Network Adapter
(total)
→ NAT
→ Custom: vmnet11
→ Custom: vmnet12

```
sudo nano /etc/network/interfaces
```

```
auto ens38
iface ens38 inet static
address 192.168.232.1
netmask 255.255.255.0
```

```
sudo nano /etc/sysctl.conf
```

```
net.ipv4.ip_forward=1
```

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

```
sudo nano /etc/sysctl.conf
```

```
(take off #)
net.ipv4.ip_forward=1
```

```
sudo reboot
```
Hosts

Victim (Ubuntu)
Add 1 Network Adapter
(total) → NAT → Custom: vmnet11

Attacker (Kali Linux)
Add 1 Network Adapter
(total) → NAT → Custom: vmnet12

sudo nano /etc/network/interfaces

auto eth0
iface eth0 inet static
address 192.168.232.2
netmask 255.255.255.0
gateway 192.168.232.1

sudo /etc/init.d/networking restart
In victim machine we must install Apache

```bash
sudo apt-get install apache2 apache2-utils
```

```bash
sudo su
```

```bash
cd /var/www/html
```

```bash
echo "This is victim webservice" > index.html
```

```bash
nano /etc/apache2/site-available/000-default.conf
```
Apache Web Server

VirtualHost *:80
  # The ServerName directive sets the request scheme, hostname and port that
  # the server uses to identify itself. This is used when creating
  # redirection URLs. In the context of virtual hosts, the ServerName
  # specifies what hostname must appear in the request's Host: header to
  # match this virtual host. For the default virtual host (this file) this
  # value is not decisive as it is used as a last resort host regardless.
  # However, you must set it for any further virtual host explicitly.
  # ServerName www.example.com

ServerAdmin webmaster@localhost
DocumentRoot /var/www/html

<Directory "/var/www/html">
  AuthType Basic
  AuthName "Restricted Content"
  AuthUserFile /etc/apache2/.htpasswd
  Require valid-user
</Directory>

# Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
# error, crit, alert, emerg.

Get Help: ^G
Write Out: ^O
Where Is: ^M
Cut Text: ^K
Justify: ^J
Cur Pos: ^C
Exit: ^X
Read File: ^R
Replace: ^A
Uncut Text: ^U
To Spell: ^T
Go To Line: ^G
Apache Web Server

htpasswd -c /etc/apache2/.htpasswd web
(insert new password for “web” user)

sudo /etc/init.d/networking restart
sudo /etc/init.d/apache2 restart

#After this configuration, disable the NAT interface
Install Snort

```bash
sudo apt-get install snort
```

* Put the network interface of the network that we want to defend (ens39)
* the network address (192.168.248.0/24)

Test Installation

```bash
sudo snort -V
```

Configuration

```bash
sudo nano /etc/snort/snort.conf
```

(with the automatic installation you have to modified only HOME_NET inserting the network that you want defend)

Test Configuration

```bash
sudo snort -c /etc/snort/snort.conf
```

(If start the sniffing mode, it’s a good thing!)
Create the custom rules file

```bash
sudo touch /etc/snort/rules/custom.rules
sudo nano /etc/snort/snort.conf
```

*include custom.rules in the snort.conf

**Write the rules**

```bash
sudo nano /etc/snort/rules/custom.rules
```

In rules folder you can put other rules that you can download from Snort site.
Some commands to execute snort IDS

**Verbose mode**

```
snort -v
```

**Using configuration file and show alert on the terminal**

```
snort -c /etc/snort/snort.conf -A console -i ens38
```

**Log mode**

```
sudo snort -c /etc/snort/snort.conf -A console -i ens38 -l /var/log/snort -k ascii
```
Configuration Snort IPS
manual installation

Install dependencies

```bash
sudo su


sudo apt-get update

sudo apt-get upgrade
```

Create Tmp directory

```bash
mkdir -p /tmp/snort-install/

cd /tmp/snort-install/

Download in this folder daq from the Snort Site

Download in this folder libdnet.1.11
Configuration Snort IPS
manual installation

Install libdnet

tar -xvf libdnet-1.11.tar.gz
./configure “CFLAGS=-fPIC”
make
sudo checkinstall -y
sudo dpkg -i libdnet-1.11-1.amd64.deb
sudo ln -s /usr/local/lib/libdnet.1.0.1 /usr/lib/libdnet.1

Install daq

(Download and Extract daq) in /tmp/snort-install/

./configure
make
sudo checkinstall -y
sudo dpkg -i daq_2.0.6-1_amd64.deb
Snort as IPS with AFPACKET

Enable DAQ in AFPACKET

```bash
sudo nano /etc/snort/snort.conf
```

It’s a inline modality on Linux using two bridged interfaces
Installing Suricata on Windows

1) Install WinPcap from https://www.winpcap.org/install/
2) Install Suricata from https://suricata-ids.org/download/
3) Place rules into C:\Program Files\Suricata\rules\ or C:\Program Files (x86)\Suricata\rules\ 
4) Modify the Suricata configuration file C:\Program Files\Suricata\suricata.yaml or C:\Program Files (x86)\Suricata\suricata.yaml settings
   • Home Network
   • External Network
   • Rule Files
   • (In Windows 32 bit) Suricata Folder
Run Suricata on Windows

1) “Run as administrator” *Prompt command*

2) Run the command “suricata.exe -c suricata.yaml -i IPHOST”

3) In `C:\Program Files\Suricata\log\fast.log` `C:\Program Files (x86)\Suricata\fast.log` contains the logs
Installing Suricata in IPS mode on Ubuntu

1) On terminal: “sudo apt-get install suricata”
2) Place rules into /etc/suricata/rules
3) Modify the Suricata configuration file /etc/suricata/suricata.yaml settings
   - Home Network
   - External Network
   - Rule Files
4) On terminal: “sudo iptables -I FORWARD -j NFQUEUE”
Run Suricata on Linux IPS mode

1) Run the command “sudo suricata -c suricata.yaml -q 0”
2) /var/log/suricata/fast.log contains the logs
Barnyard2

- Parsing and processing of snort log (unified2 format file) and output it in other output:
  - Storing on database (PostgreSQL or MySQL/MariaDB)
  - Talk to Sguil daemon
  - Talk to SnortSam
Installing Barnyard2 with MySql/MariaDB on Ubuntu 17.04

- sudo apt-get install mysql-server libmysqlclient-dev mysql-client autoconf libtool
- Set the MySql root password
- Be sure to have *libdumbnet-dev* *libpcap-dev* *bison* *flex* packages and to install the DAQ
- Be sure to have the sid-msg.map file on /etc/snort
- Be sure the custom rules have the “rev” field
- Modify the Snort configuration file adding: “output unified2: filename snort.u2, limit 128”
Installing Barnyard2 - Continue

- Install Barnyard2:
  - `wget https://github.com/firnsy/barnyard2/archive/master.tar.gz -O barnyard2-Master.tar.gz`
  - `tar zxvf barnyard2-Master.tar.gz`
  - `cd barnyard2-master`
  - `autoreconf -fvi -I ./m4`
  - `sudo ln -s /usr/include/dumbnet.h /usr/include/dnet.h`
  - `sudo ldconfig`
  - `./configure --with-mysql --with-mysql-libraries=/usr/lib/x86_64-linux-gnu`
  - `make`
  - `sudo make install`

- Put barnyard2 configuration file in Snort folder: `sudo cp etc/barnyard2.conf /etc/snort/`

- Create barnyard2 log folder and file. Make them belong to Snort user and Snort Group:
  - `sudo mkdir /var/log/barnyard2`
  - `sudo chown snort.snort /var/log/barnyard2`
  - `sudo touch /var/log/snort/barnyard2.waldo`
  - `sudo chown snort.snort /var/log/snort/barnyard2.waldo`
Installing Barnyard2 - Continue

- Create the Barnyard2 database using the commands preinstalled and create the Snort user of the database. Be sure to change “PASSWORD” with a secure password:
  - mysql -u root -p
  - mysql> create database snort;
  - mysql> use snort;
  - mysql> source schemas/create_mysql
  - mysql> CREATE USER 'snort'@'localhost' IDENTIFIED BY 'PASSWORD';
  - mysql> grant create, insert, select, delete, update on snort.* to 'snort'@'localhost';
  - mysql> exit
- Modify barnyard2 configuration file /etc/snort/barnyard2.conf adding (Be sure to change “PASSWORD” with the previous password): “output database: log, mysql, user=snort password=PASSWORD dbname=snort host=localhost sensor name=sensor01”
- Remove others read permission of the barnyard2 file: “sudo chmod o-r /etc/snort/barnyard2.conf”
Run Snort and Barnyard2

- Run Snort: “sudo snort -u snort -g snort -c /etc/snort/snort.conf -i INTERFACE”
- Run Barnyard2: “sudo barnyard2 -c /etc/snort/barnyard2.conf -d /var/log/snort -f snort.u2 -w /var/log/snort/barnyard2.waldo -g snort -u snort”
Installing BASE on Ubuntu 17.04

- BASE is a PHP frontend for Barnyard2 log database
- Permit to analyse the logs with a graphical interface
- Install the no more supported PHP 5.6:
  - sudo add-apt-repository ppa:ondrej/php
  - sudo apt-get update
  - sudo apt-get install -y apache2 libapache2-mod-php5.6 php5.6-mysql php5.6-cli php5.6 php5.6-common php5.6-gd php5.6-cli php-pear php5.6-xml
- Install Pear image Graph: “sudo pear install -f --alldeps Image_Graph”
- Install ADODB:
  - tar -xvzf adodb-5.20.8.tar.gz
  - sudo mv adodb5 /var/adodb
  - sudo chmod -R 755 /var/adodb
- Download BASE and copy on the Apache web folder:
  - wget http://sourceforge.net/projects/secureideas/files/BASE/base-1.4.5/base-1.4.5.tar.gz
  - tar xzvf base-1.4.5.tar.gz
  - sudo mv base-1.4.5 /var/www/html/base/
Configure BASE

- Copy the default configuration file:
  - cd /var/www/html/base
  - sudo cp base_conf.php.dist base_conf.php

- Modify it and set (Be sure to change “PASSWORD” with the mysql snort password):
  - $BASE_urlpath = '/base';
  - $DBlib_path = '/var/adodb/';
  - $alert_dbname = 'snort';
  - $alert_host = 'localhost';
  - $alert_port = '';
  - $alert_user = 'snort';
  - $alert_password = 'PASSWORD';

- Remove others read permission of the BASE configuration file and make the BASE folder, subfolders and files belong to “www-data” user and group:
  - sudo chmod o-r /var/www/html/base/base_conf.php

- Restart Apache: “sudo service apache2 restart”

- Open via Browser the URL: “http://localhost/base/index.php”
  - Click to “Setup page”
  - “Create BASE AG”
Installing PulledPork

- PulledPork is a tool that download automatically the Snort/Suricata rules to keep they up to date and unify every rule files in a single one
- Install the dependencies: “sudo apt-get install libcrypt-ssleay-perl liblwp-useragent-determined-perl”
- Download and install PulledPork:
  - wget https://github.com/shirkdog/pulledpork/archive/master.tar.gz -O pulledpork-master.tar.gz
  - tar xzvf pulledpork-master.tar.gz
  - cd pulledpork-master/
  - sudo cp pulledpork.pl /usr/local/bin
  - sudo chmod +x /usr/local/bin/pulledpork.pl
  - sudo cp etc/*.conf /etc/snort
- Modify the PulledPork configuration file /etc/snort/pulledpork.conf:
  - Comment or insert the Oinkcode in order to download the Oinkcode
  - Comment the rules you want download and uncomment the rules you don’t want
  - Set the version of rules “snort_version”
  - Set where to store the Snort rules (for example “snort.rules”)
  - Set where are stored the custom Snort rules
  - Set where is stored the sid-msg.map
  - Set where is stored the Snort configuration file
  - Set the distro version (in this case, Ubuntu 12.04)
- Modify the Snort configuration file. Comment every rules and add the rule “snort.rules”
- Run PulledPork: “sudo /usr/local/bin/pulledpork.pl -c /etc/snort/pulledpork.conf -l “
Snort rules

- From https://www.snort.org/downloads/#rule-downloads you can download the Snort rules (rules are organized in category in rules)
  - Snort Community (free)
  - Snort Registered (free previous registration)
  - Snort Subscription (starting at 29,99 €/year)
- From EmergingThreat https://rules.emergingthreats.net/open/ (free)
Some Theoretical concepts on IDS and IPS

Concept of IDS was developed by James P. Anderson on 15 April in 1980 in his technical rapport.

An IDS or an IPS is a system that was developed to detect typical signs of an intrusion. An IDS is similar to an anti-theft, it see the bad traffic but does not block it. An IPS is an IDS but with the behavior of a firewall.

Both see until the Application layer of the packets:

- NIDS/NIPS
- HIDS/HIPS
- Anomaly detection (“behavior based”)
- Misuse detection (“Signatures based”)

University of Catania
Thanks
IDS/IPS Technologies

Hardening three
University of Catania
29 May 2017