

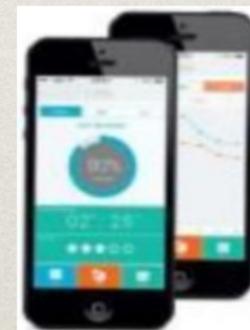
# SecurityPi

Secure your Raspberry Pi

@rabimba | Mozilla Tech Speaker | RICE University

OpenIoT Summit 2017

# Why

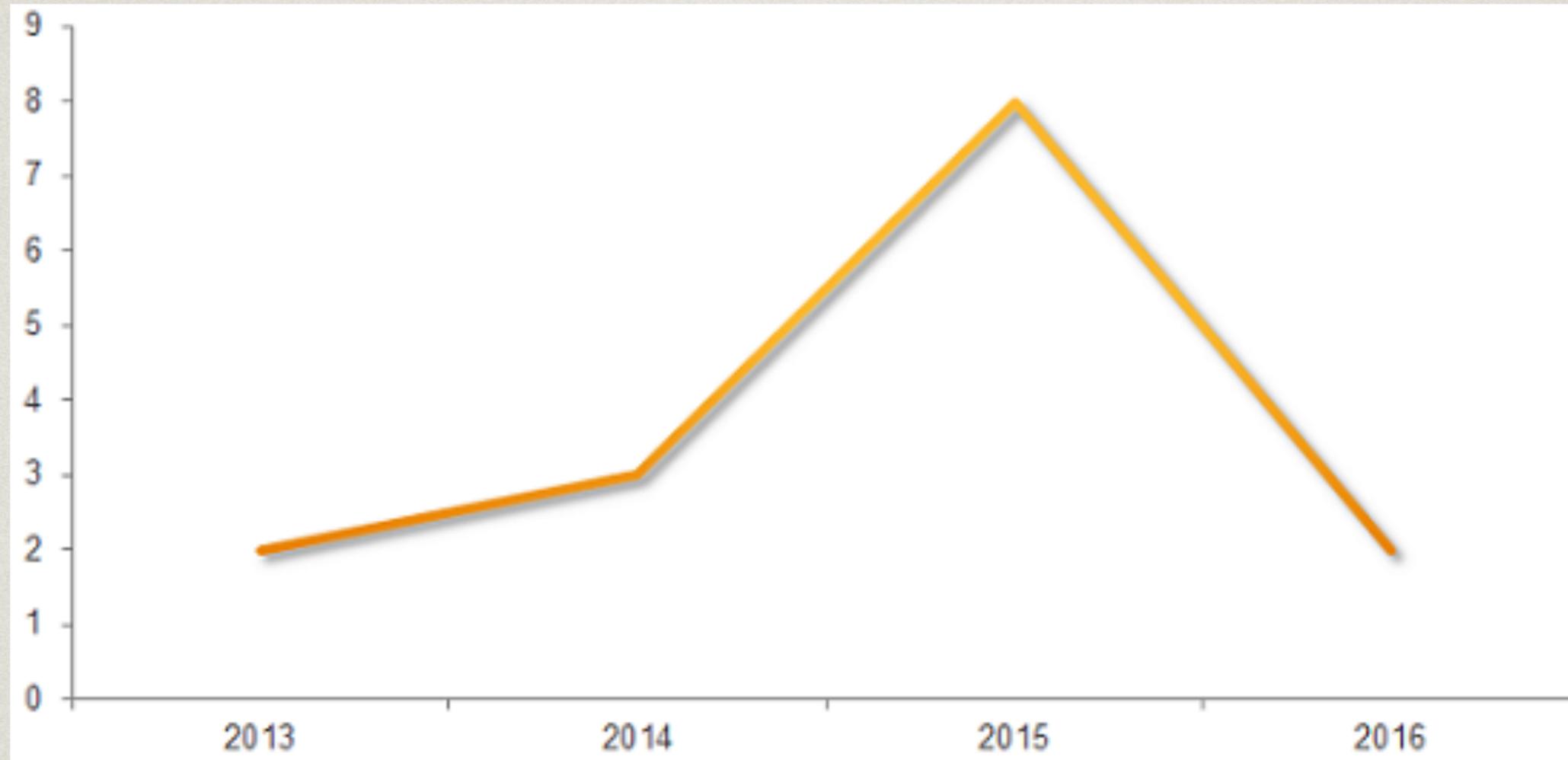


# How



# Protect the Legacy

# Wait. Do we need protection?

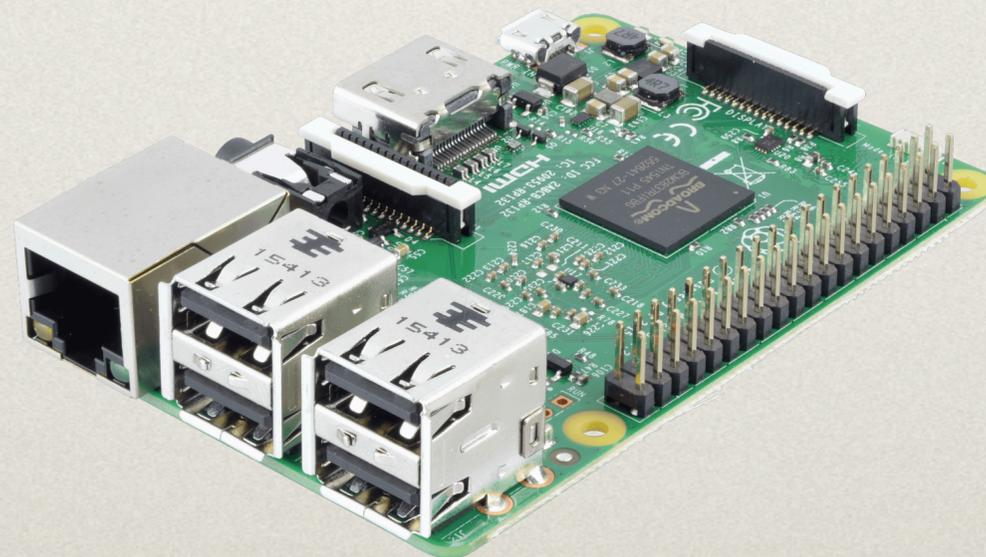


*New IoT malware families by year.*

*The number IoT threats jumped in 2015 and many of these threats continue to be active into 2016*

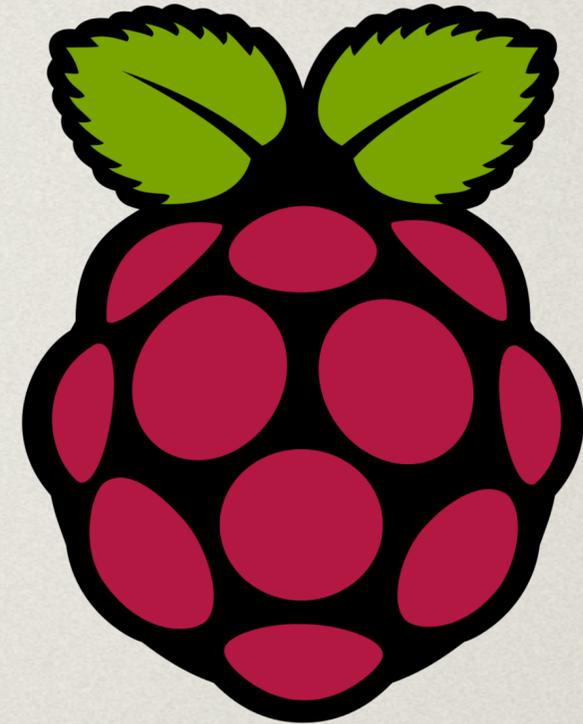
# Tools for the trade

- Raspberry Pi 3 with case
- MicroSD Card
- Power Adaptor for pi (Important!)



# Install Image

- Raspbian (Debian Wheezy)
- NOOBS



But what about my network?



# Configure Network : Part 1

## Gateway!

- **Pro:**
  - No additional hardware needed
  - Simple setup
- **Con**
  - Attackers can bypass device by connecting directly to actual gateway/router
  - Performance implications

# Configure Network : Part 2

## Mirror Port!

- **Pro:**
  - No additional hardware needed
  - All traffic will be monitored
  - Raspberry Pi isn't inline
- **Con:**
  - Home/SMB network equipment may not support Span/Mirror ports

# Configure Network : Part 3

## Grad Student Way (In-Line)

- **Pro:**
  - All traffic will be monitored
- **Con:**
  - Raspberry Pi is in-line with all network traffic
  - Performance implications

# Getting BRO Onboard

## Install Required Dependencies

◆ `$ sudo apt-get install cmake make gcc g++ flex bison libpcap-dev swig zlib1g-dev`

## Download Bro Source Code

◆ `$ wget https://www.bro.org/downloads/release/bro-2.4.tar.gz`

## Unpack

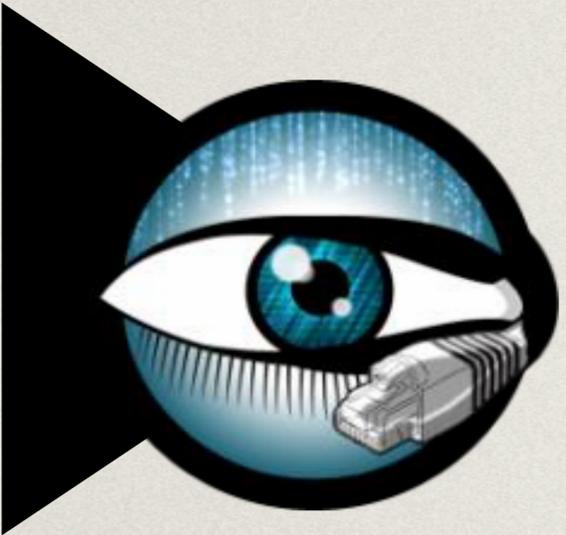
`$ sudo ./configure --prefix=/opt/nsm/bro`

`$ sudo make`

`$ sudo make install`

```
26 #Install Bro
27 echo "Installing Bro"
28 sudo wget https://www.bro.org/downloads/release/bro-2.4.1.tar.gz
29 sudo tar -xzf bro-2.4.1.tar.gz
30 sudo mkdir /opt/nsm
31 sudo mkdir /opt/nsm/bro
32 cd bro-2.4.1
33 sudo ./configure --prefix=/opt/nsm/bro
34 sudo make
35 sudo make install
36 cd ..
37 sudo rm bro-2.4.1.tar.gz
38 sudo rm -rf bro-2.4.1/
```

# BRO Intrusion Detection System



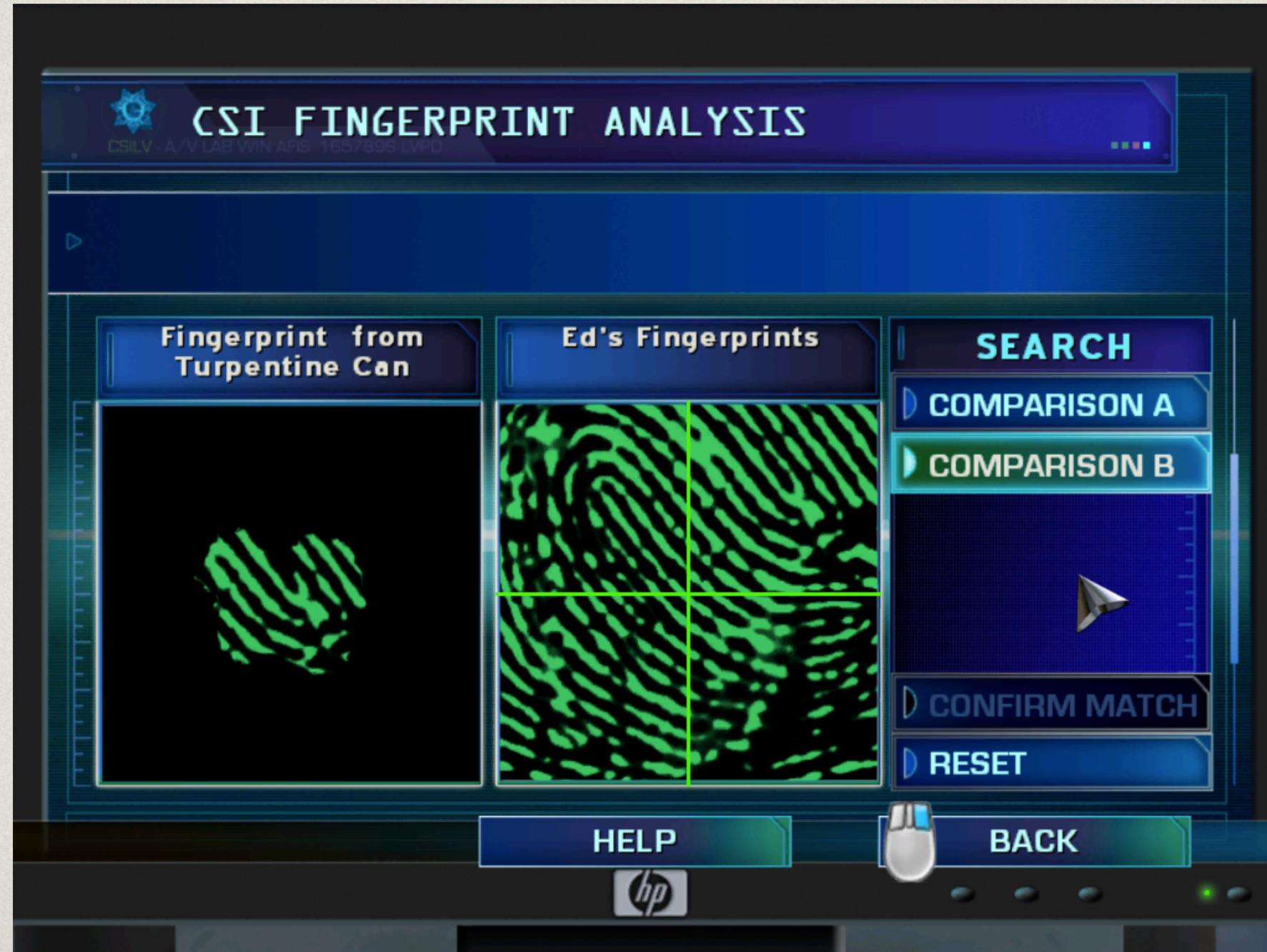
```

Full Packet Capture
73 65 72 20 72 6F 6F 74 20 62 79 20 28 75 69 64 ser root by (uid
3D 30 29 89 70 94 50 E4 ED 0A 00 99 00 00 99 =0).p.P.....
00 00 00 52 54 00 DA 2C 4C 52 54 00 DA 98 99 08 ...RT...LRT.....
00 45 00 00 8B 00 00 40 00 40 11 43 37 .E.....@.@.C7...
      A4 DF 02 02 00 77 79 82 3C 37 38 .....my.<78
3E 4E 6F 76 20 20 33 20 31 32 3A 31 37 3A 30 31 >Nov 3 12:17:01
20 64 61 74 61 62 61 73 65 20 2F 55 53 52 2F 53 database /USR/S
42 49 4E 2F 43 52 4F 4E 5B 31 38 31 33 35 5D 3A BIN/CRON[18135]:
20 28 72 6F 6F 74 29 20 43 4D 44 20 28 20 20 20 (root) CMD (
63 64 20 2F 20 26 26 20 72 75 6E 2D 70 61 72 74 cd / && run-part
73 20 2D 2D 72 65 70 6F 72 74 20 2F 65 74 63 2F s --report /etc/
63 72 6F 6E 2E 68 6F 75 72 6C 79 29 89 70 94 50 cron.hourly).p.P
13 04 0B 00 88 00 00 00 88 00 00 00 52 54 00 DA .....RT..
2C 4C 52 54 00 DA 98 99 08 00 45 00 00 7A 00 00 ,LRT.....E.,z..
40 00 40 11 43 48      A4 DF @.@.CH.....
02 02 00 66 AD DD 3C 38 36 3E 4E 6F 76 20 20 33 ...f..<86>Nov 3
20 31 32 3A 31 37 3A 30 31 20 64 61 74 61 62 61 12:17:01 databa
73 65 20 43 52 4F 4E 5B 31 38 31 33 34 5D 3A 20 se CRON[18134]:
70 61 6D 5F 75 6E 69 78 28 63 72 6F 6E 3A 73 65 pam_unix(cron:se
73 73 69 6F 6E 29 3A 20 73 65 73 73 69 6F 6E 20 ssion): session
63 6C 6F 73 65 64 20 66 6F 72 20 75 73 65 72 20 closed for user
72 6F 6F 74 42 71 94 50 62 6E 05 00 3C 00 00 00 rootBq,Pbn..<...
3C 00 00 00 52 54 00 0D 5E C5 52 54 00 DA 2C 4C <...RT..^RT...L
08 06 00 01 08 00 06 04 00 01 52 54 00 DA 2C 4C .....RT...L
      00 00 00 00 00 00      00 00 .....+..
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

```

- conn.log dhcp.log
- dnp3.log dns.log
- ftp.log http.log
- irc.log
- known\_services.lo
- g
- modbus.log ius.log
- smtp.log snmp.log
- ssh.log ssl.log
- syslog.log
- tunnel.log **intel.log**
- notice.log**

# Make BRO Great Again



# Integrate Critical Stack

- Create New Collection
- My Feeds (0)
- Add More Feeds
- MANAGE COLLECTION
  - Edit This Collection
  - Delete This Collection
- ORDER FEEDS
  - Most Indicators
  - Highest Rating
  - Most Subscribers
  - Recently Added
  - Recently Updated
  - Name Ascending
  - Name Descending

**bambenekconsulting.com DGA Domains**

886,281 1,908

★★★★★ (3)

**hosts-file.net Malware Domains**

140,261 1,658

★★★★★ (1)

**hosts-file.net Fraud Domains**

136,692 1,274

★★★★★ (0)

**hosts-file.net Phishing Domains**

127,439 1,338

★★★★★ (0)

**Subscribe to your first feed.**

Select the Feeds you want to add by hovering over each. Choose Subscribe to select a Feed. Repeat for all the Feeds you want. As you add Feeds, they will disappear from this page. They will now be present in the My Feeds section.

**nullsecure.org Threat Feed**

31,451 1,074

★★★★★ (1)

**blocklist.de IP Blocklist**

29,692 1,323

★★★★★ (0)

**ET: Botnet Command and Control**

25,628 1,932

★★★★★ (3)

**hosts-file.net Ad/Tracking Domains**

23,008 1,075

★★★★★ (0)

**hosts-file.net Illegal Pharmacy Domains**

20,139 1,063

★★★★★ (0)

**Malware Domains**

18,667 2,873

★★★★★ (4)

**Abuse Reporting and Blacklisting**

16,029 1,594

★★★★★ (1)

# Integrate Critical Stack

```
$ wget https://intel.criticalstack.com/client/critical-stack-intel-arm.deb  
sudo dpkg -i critical-stack-intel-arm.deb
```

Add the API Key

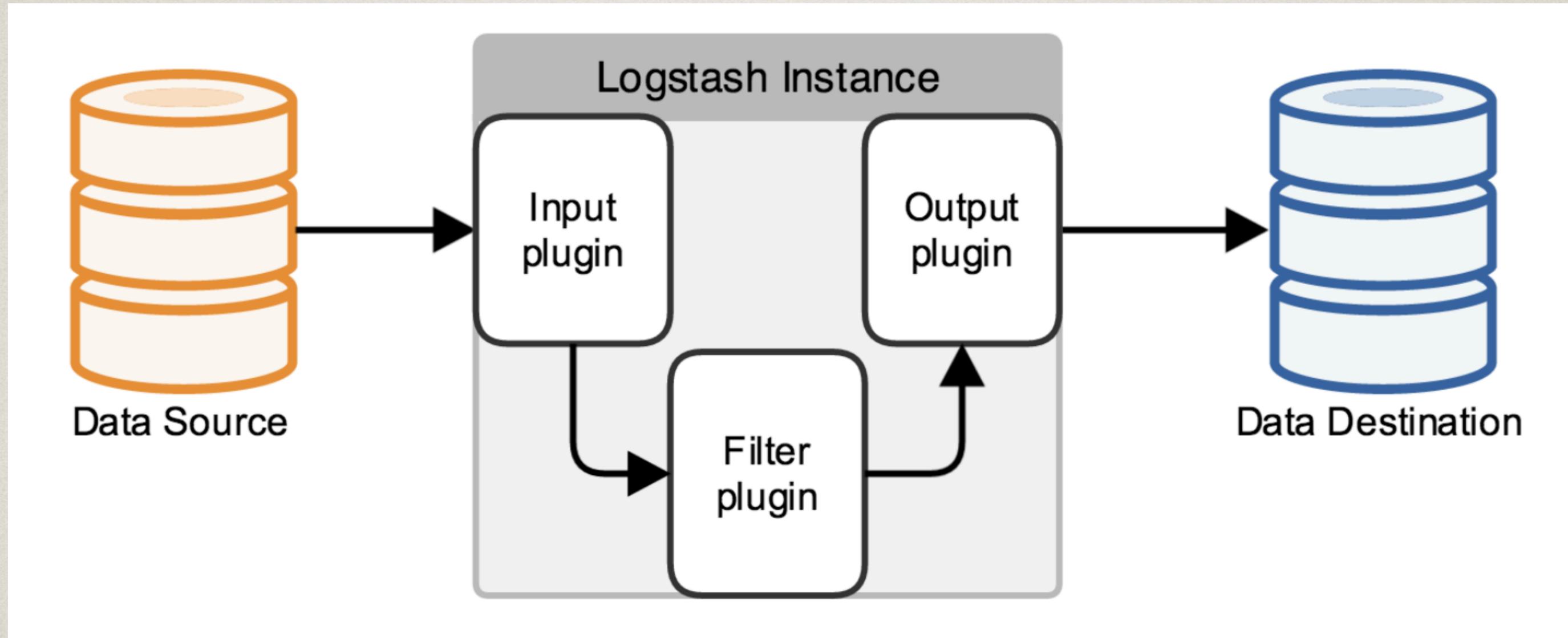
```
◆ $ sudo -u critical-stack critical-stack-intel api <key>
```

```
41 #Install Critical Stack  
42 echo "Installing Critical Stack Agent"  
43 sudo wget https://intel.criticalstack.com/client/critical-stack-intel-arm.deb  
44 sudo dpkg -i critical-stack-intel-arm.deb  
45 sudo -u critical-stack critical-stack-intel api $cs_api  
46 sudo rm critical-stack-intel-arm.deb
```

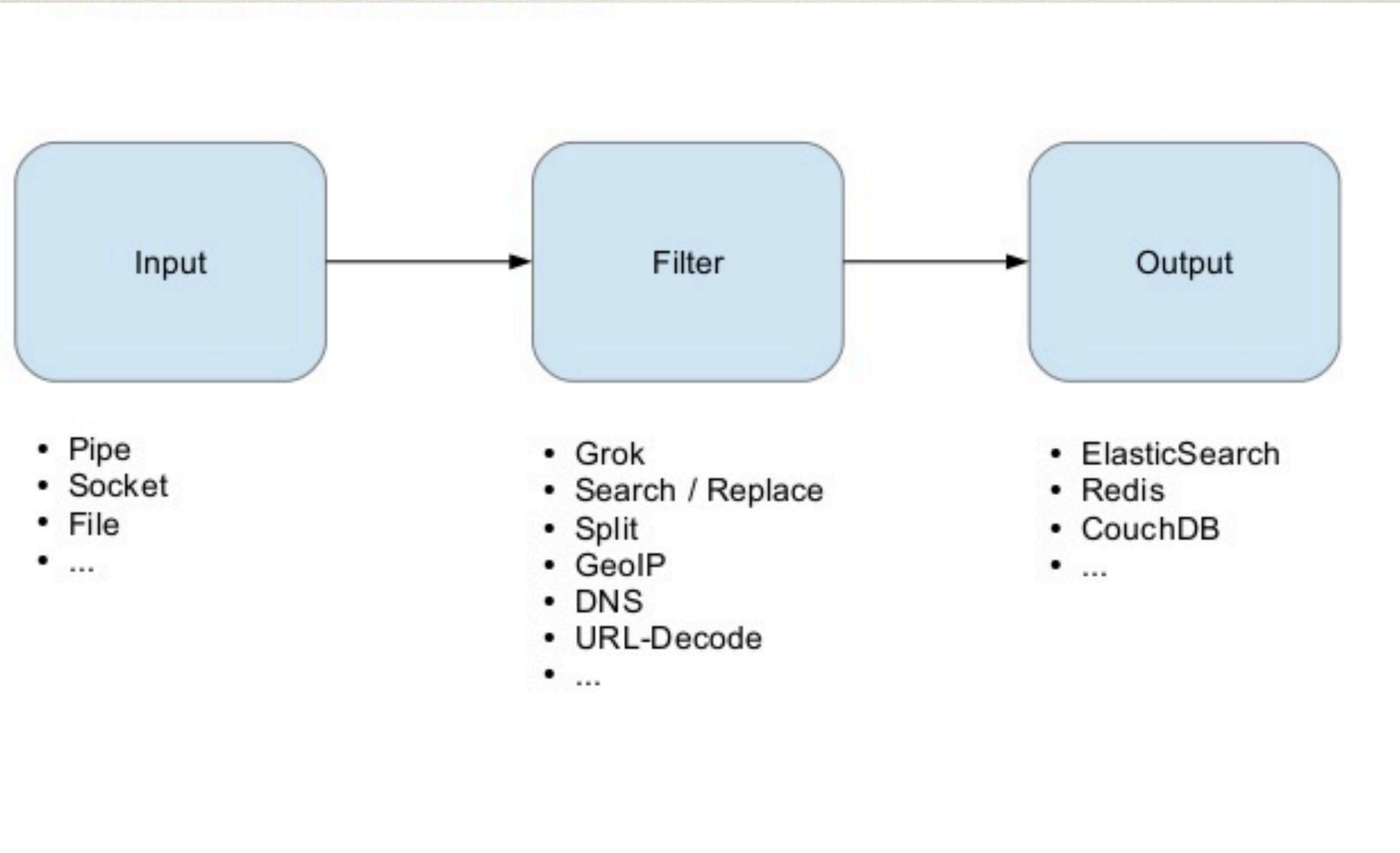
# What about my logs?



# Stash The Logs



# In Short: Logstash



# What we will do!

## Overview

- Utilizing Custom Patterns
- GROK Message Filtering
- Adding Custom Fields
- Adding Geo IP Data
- Date Match
- Using Translations for Threat Intel

# Get LogStash



```
$ wget https://download.elastic.co/logstash/logstash/logstash-1.5.3.tar.gz
$ sudo mv /opt/logstash-1.5.3/ /opt/logstash
$ cd /opt/logstash
$ bin/logstash -e 'input { stdin { } } output { stdout { } }'
```

# How do I see the logs?



# elastic

## Install

```
$ wget
```

```
https://download.elastic.co/elasticsearch/elasticsearch/elasticsearch-1.7.1.deb
```

```
$ sudo dpkg -i elasticsearch-1.7.1.deb
```

\*Update cluster name in yml file

# I wanted to “See”!



## Install

```
$ wget https://download.elastic.co/kibana/kibana/kibana-4.1.0-linux-x86.tar.gz
```

```
$ sudo mkdir /opt/kibana
```

```
$ cd /opt/kibana
```

```
$ bin/kibana
```

**Another error?? Your node needs another ARM!**

# I wanted to “See”!



## Custom ARM Install

```
$ wget http://node-arm.herokuapp.com/node\_latest\_armhf.deb
$ sudo dpkg -i node_latest_armhf.deb
$ sudo mv /opt/kibana/node/bin/node /opt/kibana/node/bin/node.orig
$ sudo mv /opt/kibana/node/bin/npm /opt/kibana/node/bin/npm.orig
$ sudo ln -s /usr/local/bin/node /opt/kibana/node/bin/node
$ sudo ln -s /usr/local/bin/npm /opt/kibana/node/bin/npm
$ /opt/kibana/bin/kibana
```



Discover Visualize Dashboard **Settings**

Indices **Advanced** Objects About

Index Patterns

**Warning** No default index pattern. You must select or create one to continue.

## Configure an index pattern

In order to use Kibana you must configure at least one index pattern. Index patterns are used to identify the Elasticsearch index to run search and analytics against. They are also used to configure fields.

**Index contains time-based events**

**Use event times to create index names**

**Index name or pattern**

Patterns allow you to define dynamic index names using \* as a wildcard. Example: logstash-\*

Unable to fetch mapping. Do you have indices matching the pattern?



## Configuration

```
input {
  file {
    path =>          "/opt/bro/logs/current/*.lo
                    start_position "beginning
                    } =>          "
  }
}
output {
  elasticsearch {
    host => localhost
                    cluster          "elasticsearch-
                    =>          clustername "
  }
}
```



## Configuration

```
filter {
  grok {
    match => {
      "message" => "%{IP :client}%{WORD :method }
                  {URIPATHPARAM: request}%{NUMBER :bytes}%{NUMBER:duration }"
                  }
    }
  }
}
```



## Configuration

```
filter {  
  grok {  
  
  }  
}
```

```
patterns_dir => "/opt/logstash/custom_patterns"  
  match => {  
    message => "%{291009}"  
  }
```



- Configuration
- Create a Rule File
- /opt/logstash/custom\_patterns/bro.rule
- 291009
- (?<start\_time>\d+\.\d{6})\s+(?<uid>\S+)\s+(?:(?<evt\_srcip>[\d\.]+)|(?<evt\_srcipv6>[\w
- :]+)|-)\s+(?:(?<evt\_srcport>\d+)|-)\s+(?:(?<evt\_dstip>[\d\.]+)|(?<evt\_dstipv6>[\w:]
- )\s+(?:(?<evt\_dstport>\d+)|-
- )\s+(?<fuid>\S+)\s+(?<file\_mime\_type>\S+)\s+(?<file\_description>\S+)\s+(?<seen\_in
- dicator>\S+)\s+(?<seen\_indicator\_type>[A:]+::\S+)\s+(?<seen\_where>[A:]+::\S+)\s+(?
- <source>\S+(?:\s\S+)\*)\$



# Configuration

```
filter {
```

```
if [message] =~ /^(?\d{10}\.\d{6})\t(?<evt_srcip>[\d\.]+\t(?<evt_dstip>[\d\.]+\t(?<evt_srcport>\d+)\t...
```

```
grok {
```

```
patterns_dir => "/opt/logstash/custom_patterns"
```

```
match => {
```

```
message => "%{291001}"
```

```
}
```

```
}
```

```
}
```

```
}
```



291001 (?<start\_time>\d{10}\.\d{6})\t(?<evt\_srcip>[\d\.]+\t(?<evt\_dstip>[\d\.]+\t(?<evt\_srcport>\d+)\t...



## Configuration

```

filter {
  if [message] =- /A(\d+\.\d{6}\s+\S+\s+(?:[\d\.]|[\w:]+)-)\s+(?:\d+)-\s+(?:[\d\.]|[\w:]+|
    •   )\s+(?:\d+)-\s+\S+\s+\S+\s+\S+\s+\S+\s+\S+\s+[A:]+:\S+\s+[A:]+:\S+\s+\S+(?:\s\S+)*$/ {
      grok{
        patterns_dir => "/opt/logstash/custom_patterns"
        match => {
          message => "%{291009}"
        }
        add field      => [ "rule_id", "291009" ]
        add field      => [ "Device Type", "IPSIDSDevice" ]
        add field      => [ "Object", "Process" ]
        add field      => [ "Action", "General" ]
        add field      => [ "Status", "Informational" ]
      }
    }
  }
}

```



```
filter {
  ....all normalization code above here...
  geoup {
    source => "evt_dstip"
    target => "geoup_dst"
    database => "/opt/logstash/GeoLiteCity.dat"
    add_field => ["[geoup_dst][coordinates]", "%{[geoup_dst][longitude]}"]
    add_field => ["[geoup_dst][coordinates]", "%{[geoup_dst][latitude]}"]
    add_field => ["[geoup_dst][coordinates]", "%{[geoup_dst][city\_name]}"]
    add_field => ["[geoup_dst][coordinates]", "%{[geoup_dst][continent\_code]}"]
    add_field => ["[geoup_dst][coordinates]", "%{[geoup_dst][country\_name]}"]
    add_field => ["[geoup_dst][coordinates]", "%{[geoup_dst][postal\_code]}"]
  }
  mutate {
    convert => [ "[geoup_dst][coordinates]", "float" ]
  }
}
```

New Elasticsearch Template  
Needed



- Configuration

- filter {
- ...bro normalization stuff... translate {
- field => "evt\_dstip"
- destination => "badIP" dictionary\_path => '/opt/logstash/IP.yaml '
- }
- }
- But what goes in IP.yaml?



## Configuration

- Dictionary Hash in standard YAML format

"1.2 .3 .4":

Bad IP

"ab c123":

Very Bad IP

- Install the translate plugin
- `$ cd /opt/logstash`
  - `$ bin/plugin install logstash-filter-translate`



## Configuration

- TOR Exit IP: <https://check.torproject.org/exit-addresses>
- Malicious IP: <http://www.malwaredomainlist.com/hostslist/ip.txt>
- Automate the scraping of available intel
- Populate the YAML Files

torexit.yaml

```
"162.247.72.201": "YES"  
"24.187.20.8": "YES"  
"193.34.117.51": "YES"
```

# What do I know?

*My*  
**STROKE**  
*of*  
**INSIGHT**





## Configuration

```
if      "YES" in [tor_IP] { email {
options => [ "smtpiporHost", "SMTP_HOST",
"port", "SMTP_PORT",
"userName ", "EMAIL_USER",
"password", "EMAIL_PASS", "authenticationType",
"plain",
"starttls", "true"]
from => "<EMAIL_USER>"
subject => "Tor Exit IP Detected on Home
Network" to => "<EMAIL_USER>"
via => "smtp"
htmlbody => htmlBody }}
```

There is a TOR device in my network!



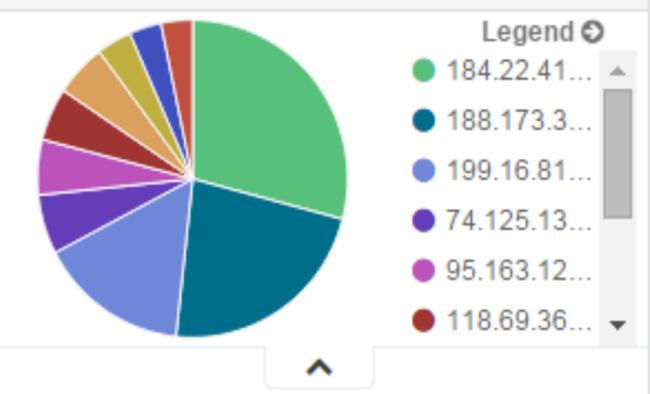
## Alerts

- TOR IP Addresses
- Malicious IP Addresses
- Malicious File Hashes
- Bro IDS intel.log results
- Bro IDS notice.log results
- Connections to different countries
- Device Specific Connection segregation

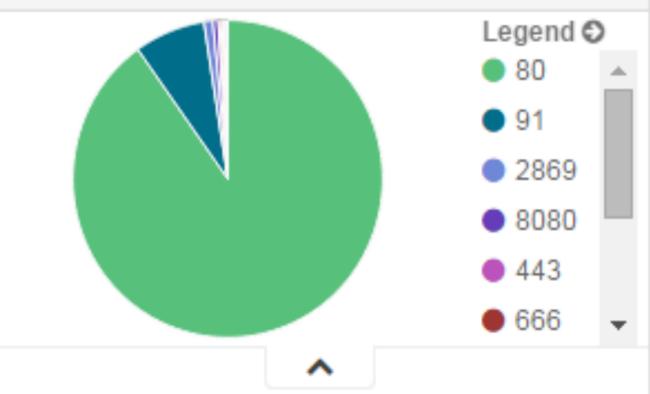
Date Chart of Log Sources



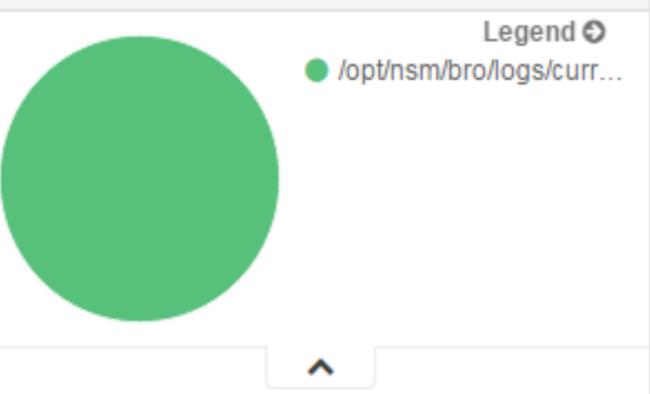
Top Destination IP



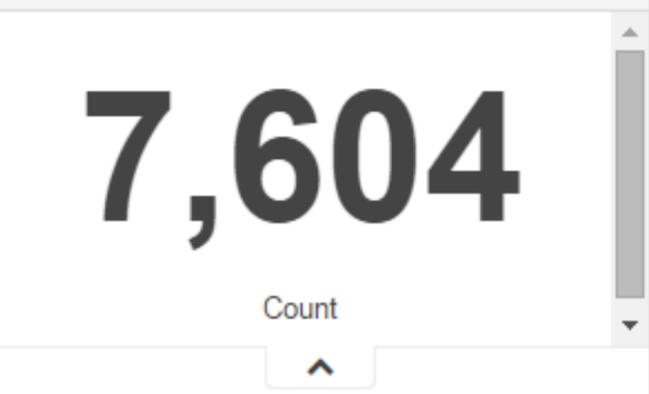
Top 10 Destination Ports



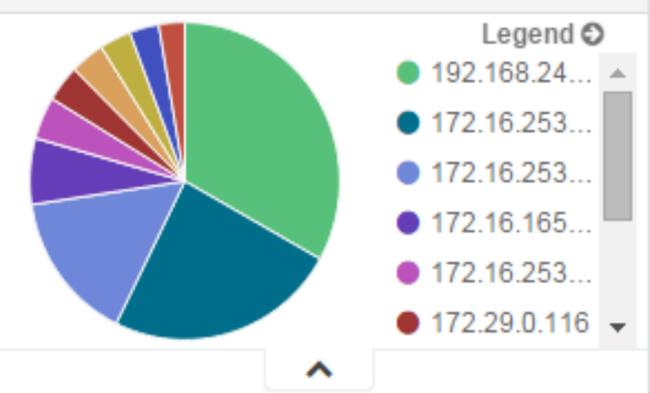
Top Log Sources



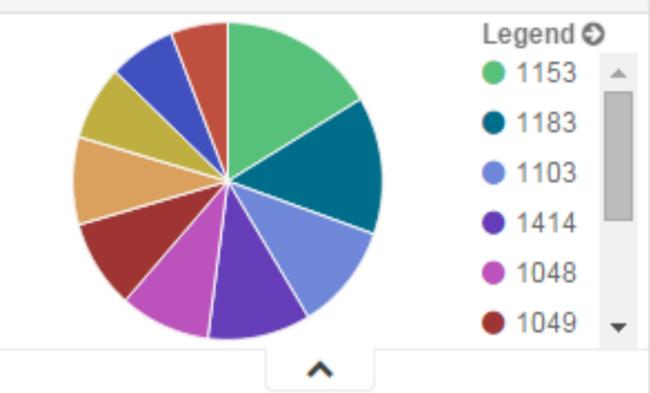
Log Count



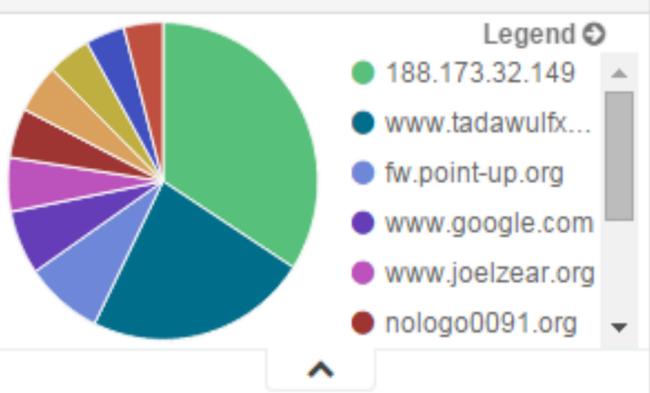
Top Source IP



Top 10 Source Ports

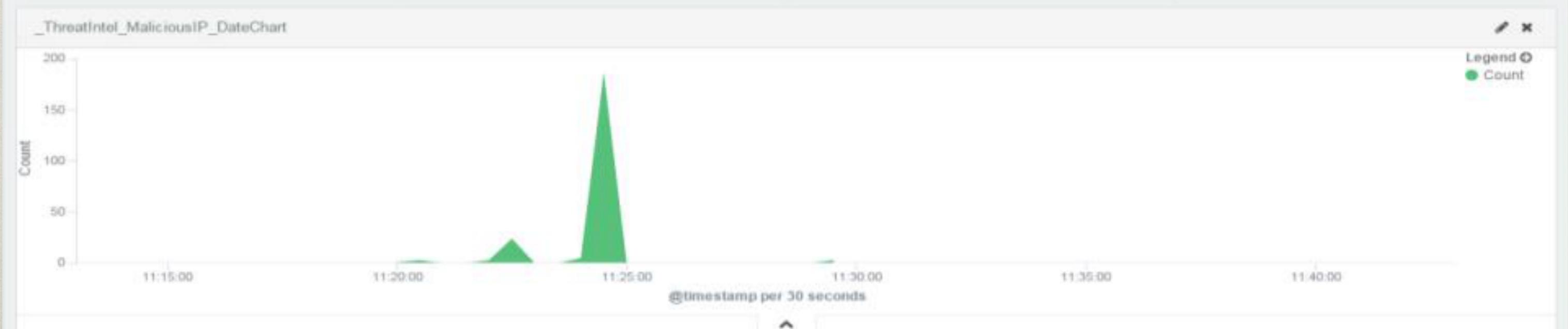
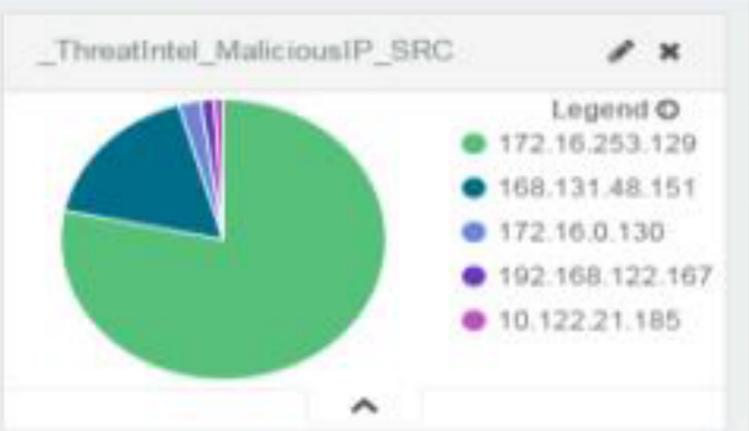
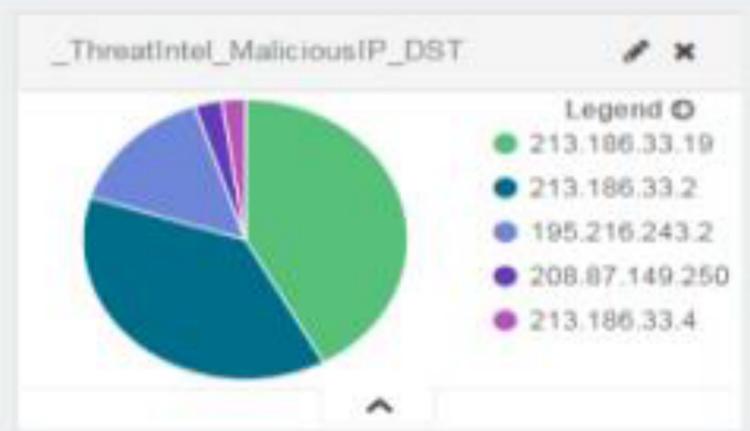


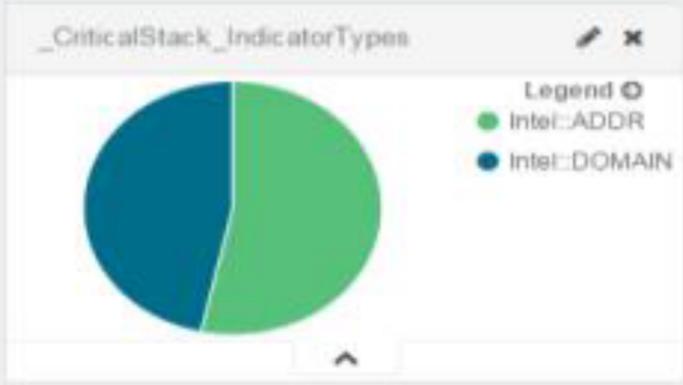
Top 10 Web Sites



Monthly Log Count

@timestamp per month	Count
December 31st 2010, 23:00:00.000	10
July 1st 2011, 00:00:00.000	99
August 1st 2011, 00:00:00.000	8





# What about proactivity?

## NMAP

- Scheduled nmap scan of subnet
  - `sudo nmap -sn 192.168.0.1/255.255.255.0`
  - `-ox nmap .xml`
- Parse **XML** file for new devices
  - New devices added to SQLite DB
    - IP Address & MAC Address
  - Email alerts when new devices found

# Show me the code!



<https://goo.gl/ks3p9Q>

# Learn more!

## Extract Features from log

```
[WARN] USER user1 connected concurrently with 2228 miles different location
- Connection1
  * IP: 162.223.5.123 (Toronto, Ontario, Canada)
  * Time: 2015-08-14 05:49:07.0 ~ 2015-08-14 08:05:27.0
- Connection2
  * IP: 69.12.4.10 (Livermore, California, United States)
  * Time: 2015-08-14 06:03:35.0 ~ 2015-08-14 08:05:27.0

# [Rule: Concurrent Login from different location] Finished
* Found 4438 Warning, 0 Critical bad actions

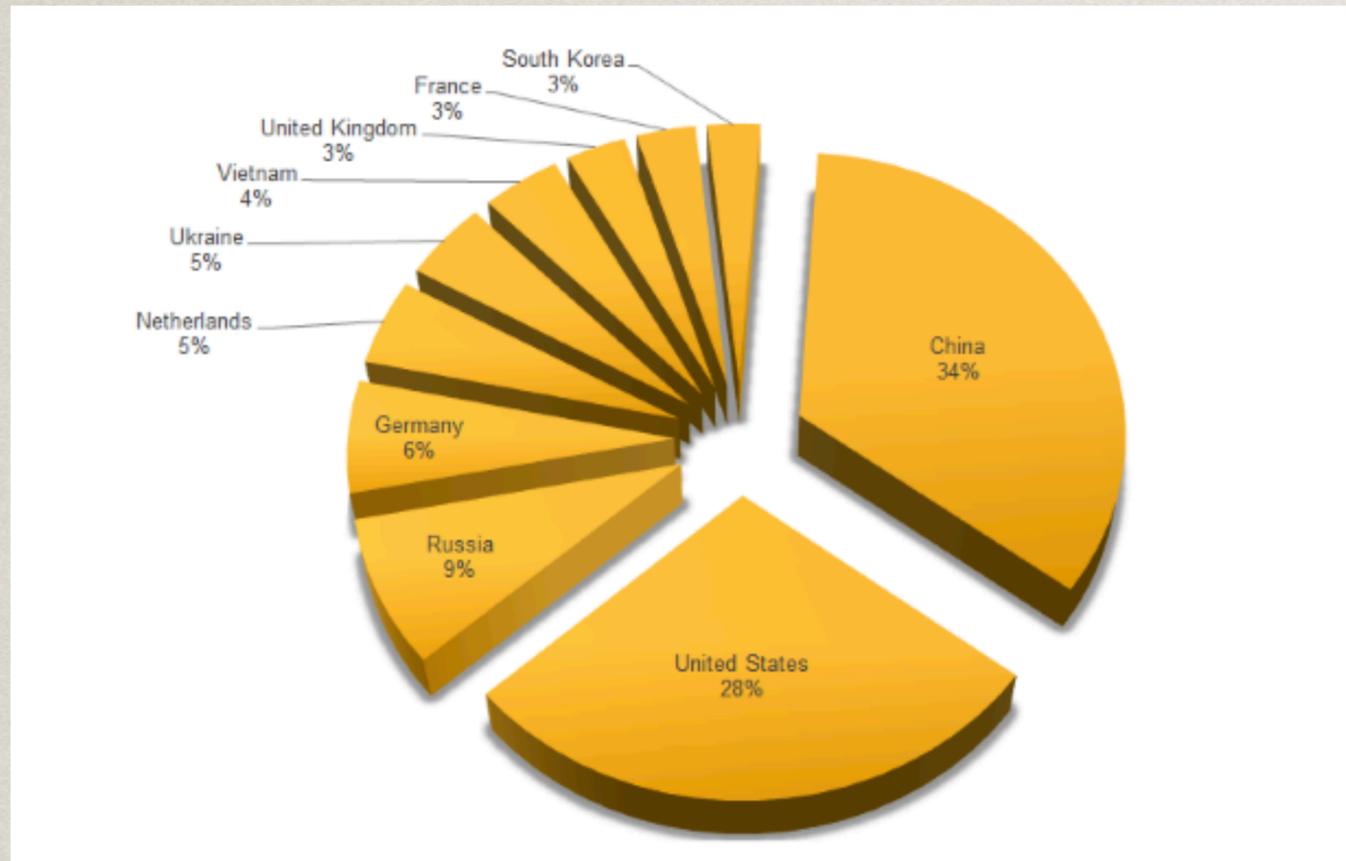
# [Rule: Successful login from Suspicious IP]

# [Rule: Successful login from Suspicious IP] Finished
* Found NO Bad action

# [Total Summary] Finished checking for 2 rules
* Found 4438 Warning, 0 Critical bad actions
```



# Does it work?



*Top ten attack origins on monitored IoT honeypot in 2016,  
by count of unique attackers*

Country	Attempts number	Country	Attempts number
China	423 (23.5 %)	U.S.	800 (26.8 %)
U.S.	366 (20 %)	China	548 (18.4 %)
Brazil	123 (6.8 %)	France	162 (5.4 %)
Russia	90 (5 %)	Canada	151(5 %)
India	85 (4.7%)	Germany	141 (4.7 %)
Canada	56 (3.1%)	Brazil	135 (4.5 %)
South Korea	43 (2.4%)	Russia	106 (3.55 %)
Germany	40 (2.2%)	India	92 (3 %)
Italy	39 (2.16%)	U.K.	78 (2.61 %)
Seychelles	38 (2.11%)	Hong Kong	67 (2.2 %)

*SSH brute force attempts on my RaspberryPi - \_ -*

# Show me the code again!



<https://goo.gl/5ufCUF>

# Commercial Solutions

<p>For Gamers</p>  <p>GT-AC5300</p> <p><a href="#">Know More</a></p>	<p>For Families</p>  <p>RT-AC88U</p> <p><a href="#">Know More</a></p>	<p>For Everyone</p>  <p>RT-AC68U</p> <p><a href="#">Know More</a></p>
---------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------

Has AiProtection (Costs \$140 ~ \$350)  
By Asus and Trend Micro

# Thank You!

@rabimba | [karanjai.moz@gmail.com](mailto:karanjai.moz@gmail.com)