Welcome to

Log analyse med Elasticsearch, Logstash og Kibana

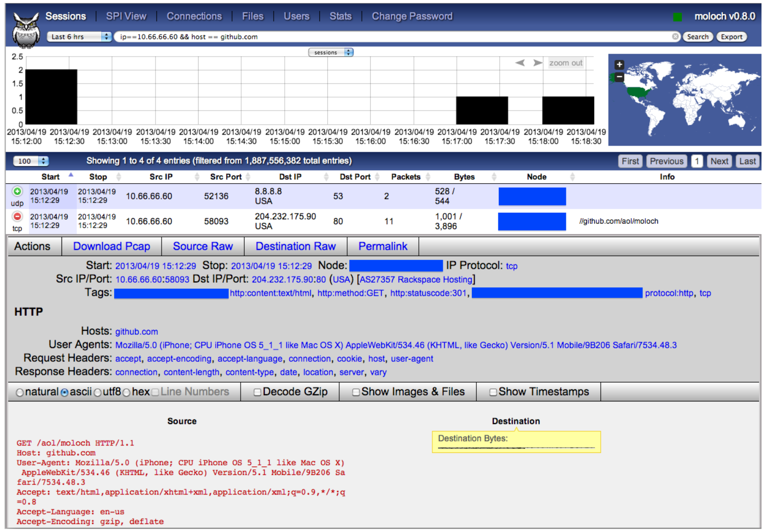
TheCamp.dk 2015

Henrik Lund Kramshøj hlk@kramse.org

THECAMP.DK - 7 open source days

Slides are available as PDF, kramshoej@Github

Goals of today



ip==10.66.66.60 && host == github.com

Showing 1 to 4 of 4 entries (filtered from 1,887,556,382 total entries)

	Start	Stop	Src IP	Src Port	Dst IP	Dst Port	Packets	Bytes	Node	Info
udp	2013/04/19 15:12:29	2013/04/19 15:12:29	10.66.66.60	52136	8.8.8.8 USA	53	2	528 / 544	[redacted]	
tcp	2013/04/19 15:12:29	2013/04/19 15:12:29	10.66.66.60	58093	204.232.175.90 USA	80	11	1,001 / 3,896	[redacted]	//github.com/aol/moloch

Start: 2013/04/19 15:12:29 Stop: 2013/04/19 15:12:29 Node: [redacted] IP Protocol: tcp
Src IP/Port: 10.66.66.60:58093 Dst IP/Port: 204.232.175.90:80 (USA) [AS27357 Rackspace Hosting]
Tags: [redacted] http:content:text/html, http:method:GET, http:statuscode:301, [redacted] protocol:http, tcp

HTTP

Hosts: github.com
User Agents: Mozilla/5.0 (iPhone; CPU iPhone OS 5_1_1 like Mac OS X) AppleWebKit/534.46 (KHTML, like Gecko) Version/5.1 Mobile/9B206 Safari/7534.48.3
Request Headers: accept, accept-encoding, accept-language, connection, cookie, host, user-agent
Response Headers: connection, content-length, content-type, date, location, server, vary

○ natural ● ascii ○ utf8 ○ hex □ Line Numbers □ Decode GZip □ Show Images & Files □ Show Timestamps

Source

```
GET /aol/moloch HTTP/1.1
Host: github.com
User-Agent: Mozilla/5.0 (iPhone; CPU iPhone OS 5_1_1 like Mac OS X)
 AppleWebKit/534.46 (KHTML, like Gecko) Version/5.1 Mobile/9B206 Sa
fari/7534.48.3
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q
=0.8
Accept-Language: en-us
Accept-Encoding: gzip, deflate
```

Destination

Destination Bytes:

Log analysis is required today - and we have many logs

Gather logs, parse logs, explain logs - fix stuff

Google your logs with the ELK stack

Show sample logs from Suricata, Sudo, SSH, Postgresql m.fl.

Plan for today

Kl 13:30 - 16:00 with a break, or shorter

Less presentation, more talk

Less me talking (only) and more 2.0 social media interaction

Trying to fit in demo and workshop-like stuff

The current situation

Internet security sucks

Personal computers like laptops suck at security

Mobile devices suck even more at security - less CPU/MEM/storage

We depend on cloud services and underfunded infrastructure - OpenSSL

We depend on others and the whole internet - DDoS

Goals: Internet Ninjas

Real super heroes are just ninjas

By knowing the internet, technologies and possibilities

Using technology and knowledge make it seem magical

In reality preparedness and defense in depth go a looooong way

Common sense is not magic, structured methods are king

Challenges

Less resources available for IT and infosec

Lots of new malware, virus, vulnerabilities and hacking

Dataloss ransomware, theft

Loss of confidentiality, 2014: 700 million lost accounts

Infosec charlatans, hype and lies

Your boss wants: No cost, and please show us great results

Solutions

Automate your job, Ansible is our poison - demo

Backup your life, help others backup, Duplicity is my choice

Learn self-defense for yourself, practice infosec war `http://ssd.eff.org`

Use hackertools to detect and identify

Categories, sort, prioritize, group problems - solve more

Measure, collect and present - make it pretty

Learn from devops, Elasticsearch Logstash Kibana D3.js

Use your brain

A lot will seem easy and basic from the outside, but when you are knee-deep in something you loose focus. Take a step back once in a while.

Case: Aalborg Farve og Lak.

> ”Vi skulle alligevel have nyt Navision-system i maj, så vi måtte fremrykke den investering. På den måde kunne vi få tastet alt ind i det nye system. I hele sagen har vi dog tabt omkring en million kroner med de mistede ordrer, ny software og revisionsbistand,”

Medejer og salgs- og personaleansvarlig hos Aalborg Farve- og Lak, Pernille Skall

Break-in through Windows Xp

Ransomware infection - across multiple systems

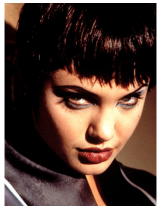
Latest backup from November (currently we are in April!)

Great that they share

Todays break-ins use yesterdays vulns, repeated and documented multiple times

`http://www.computerworld.dk/art/233684/hacker-kom-ind-via-labelprinter-tog-dans`

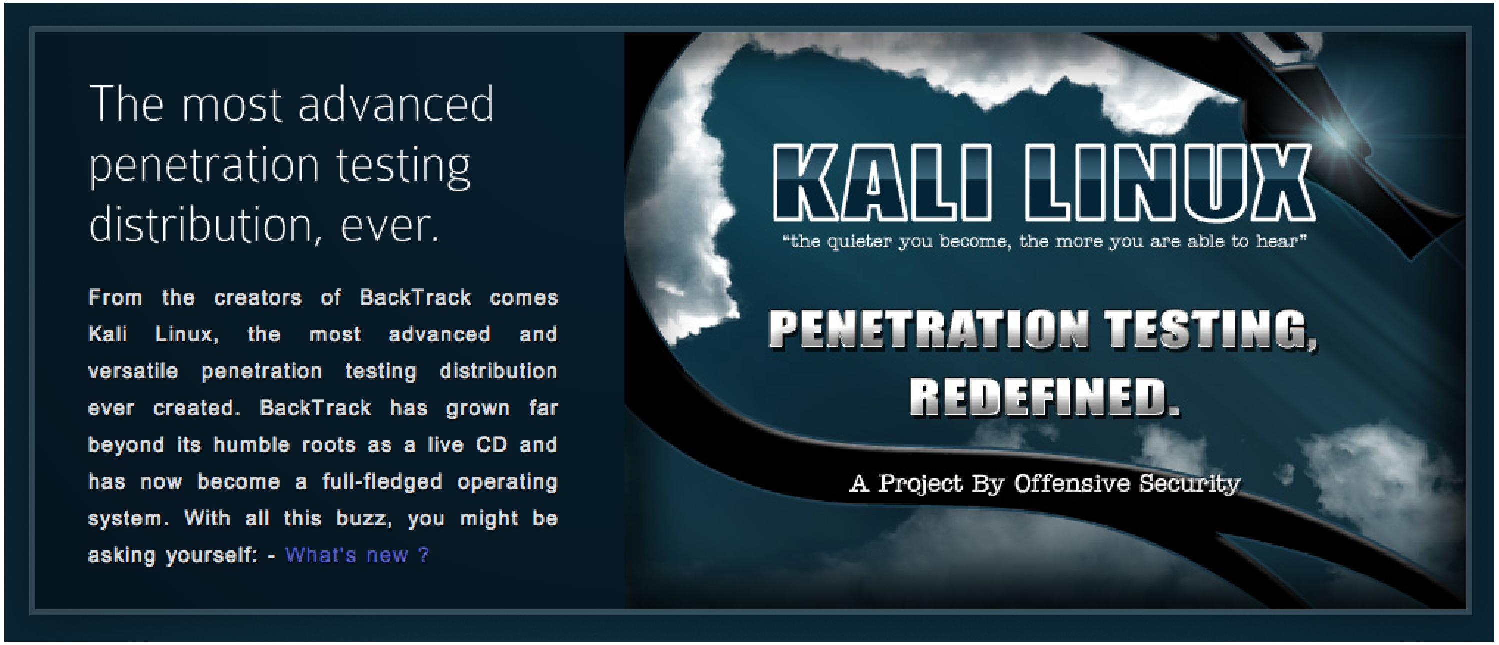
Hackertools are for everyone!



- Hackers work all the time to break stuff, Use hackertools:
- Nmap, Nping `http://nmap.org`
- Wireshark - `http://www.wireshark.org/`
- Aircrack-ng `http://www.aircrack-ng.org/`
- Metasploit Framework `http://www.metasploit.com/`
- Burpsuite `http://portswigger.net/burp/`
- Skipfish `http://code.google.com/p/skipfish/`
- Kali Linux `http://www.kali.org`

Most popular hacker tools `http://sectools.org/`

Kali Linux the pentest toolbox



The most advanced penetration testing distribution, ever.

From the creators of BackTrack comes Kali Linux, the most advanced and versatile penetration testing distribution ever created. BackTrack has grown far beyond its humble roots as a live CD and has now become a full-fledged operating system. With all this buzz, you might be asking yourself: - What's new ?

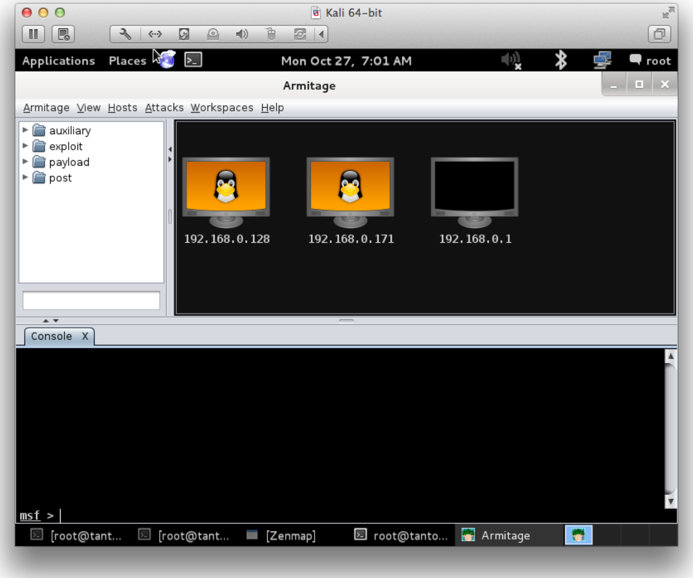
KALI LINUX
"the quieter you become, the more you are able to hear"
PENETRATION TESTING, REDEFINED.
A Project By Offensive Security

Kali `http://www.kali.org/`

100.000s of videos on youtube

Also versions for Raspberry Pi, mobile and other small computers

Metasploit and Armitage Still rocking the internet



```
msf >
```

`http://www.metasploit.com/`

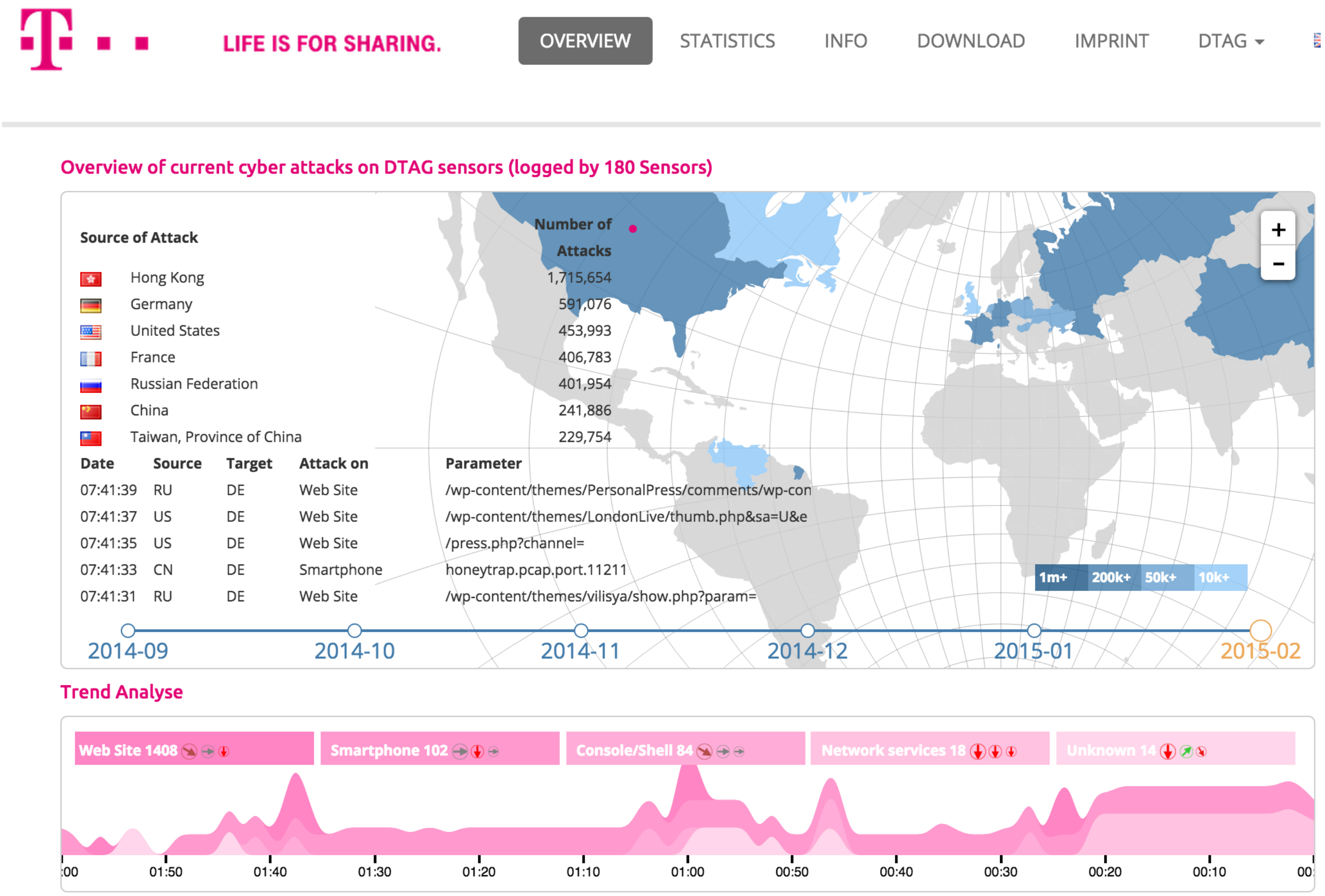
Armitage GUI fast and easy hacking for Metasploit

`http://www.fastandeasyhacking.com/`

Recommened training Metasploit Unleashed

`http://www.offensive-security.com/metasploit-unleashed/Main_Page`

Defense: Attack overview



Overview of current cyber attacks on DTAG sensors (logged by 180 Sensors)

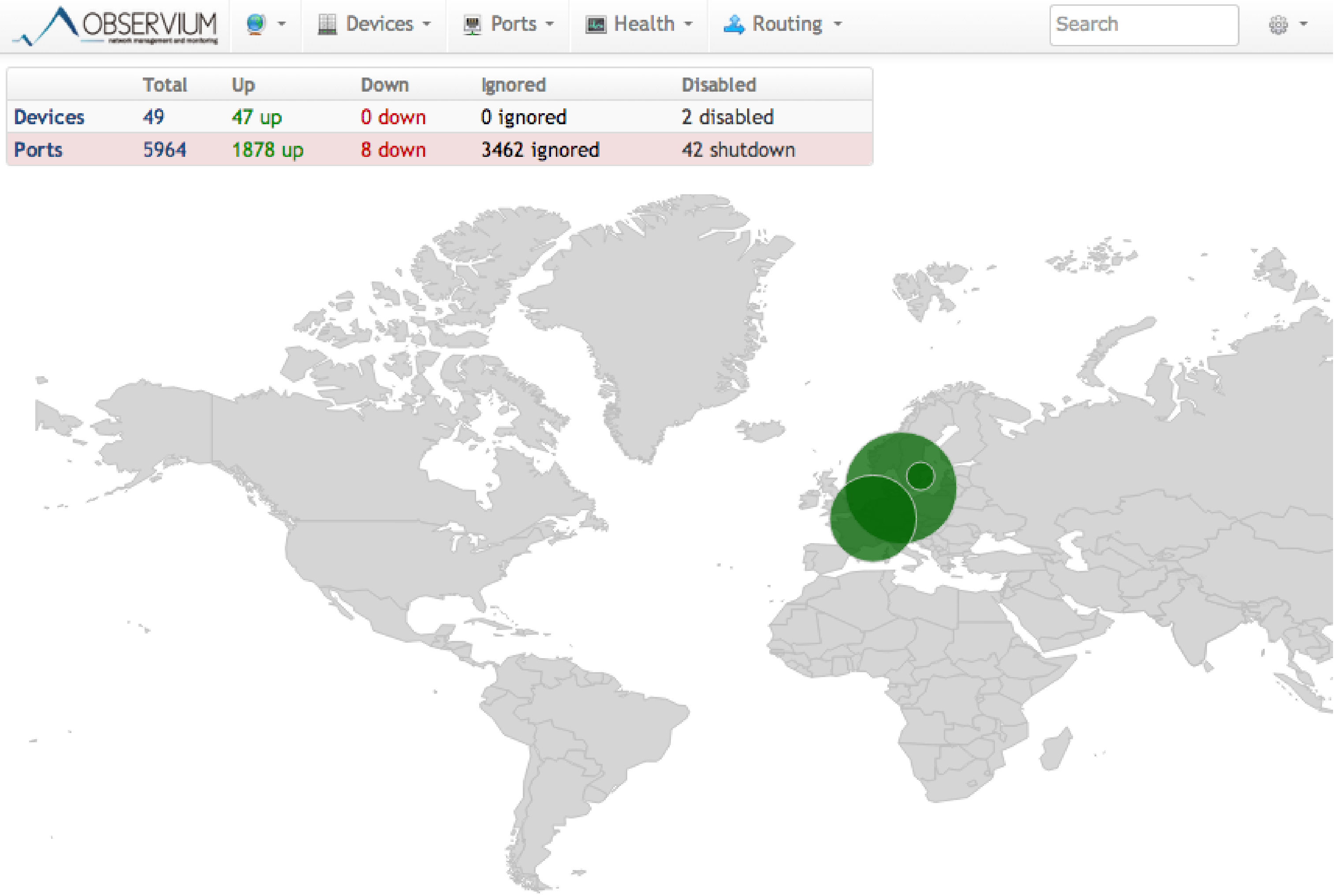
Source of Attack	Number of Attacks
Hong Kong	1,715,654
Germany	591,076
United States	453,993
France	406,783
Russian Federation	401,954
China	241,886
Taiwan, Province of China	229,754

Date	Source	Target	Attack on	Parameter
07:41:39	RU	DE	Web Site	/wp-content/themes/PersonalPress/comments/wp-con
07:41:37	US	DE	Web Site	/wp-content/themes/LondonLive/thumb.php&sa=U&e
07:41:35	US	DE	Web Site	/press.php?channel=
07:41:33	CN	DE	Smartphone	honeytrap.pcap.port.11211
07:41:31	RU	DE	Web Site	/wp-content/themes/vilisya/show.php?param=

Trend Analyse

```
http://www.sicherheitstacho.eu/?lang=en
```

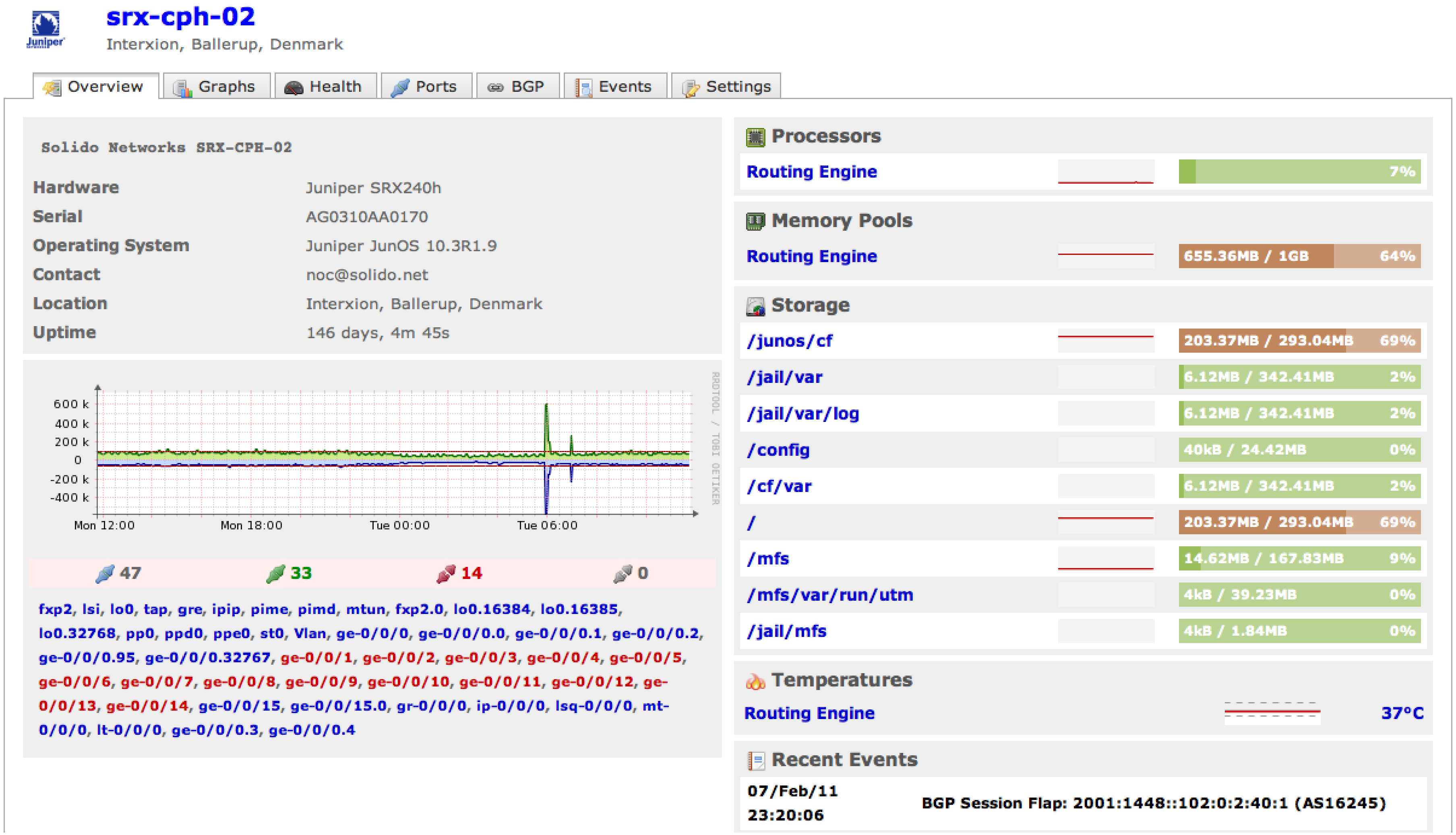
Graphs and Dashboards!



	Total	Up	Down	Ignored	Disabled
Devices	49	47 up	0 down	0 ignored	2 disabled
Ports	5964	1878 up	8 down	3462 ignored	42 shutdown

Observium

Observium example router overview



srx-cph-02

Interxion, Ballerup, Denmark

Solido Networks SRX-CPH-02

Hardware	Juniper SRX240h
Serial	AG0310AA0170
Operating System	Juniper JunOS 10.3R1.9
Contact	noc@solido.net
Location	Interxion, Ballerup, Denmark
Uptime	146 days, 4m 45s

47 33 14 0

fxp2, lsi, lo0, tap, gre, ipip, pime, pimd, mtun, fxp2.0, lo0.16384, lo0.16385, lo0.32768, pp0, ppd0, ppe0, st0, Vlan, ge-0/0/0, ge-0/0/0.0, ge-0/0/0.1, ge-0/0/0.2, ge-0/0/0.95, ge-0/0/0.32767, ge-0/0/1, ge-0/0/2, ge-0/0/3, ge-0/0/4, ge-0/0/5, ge-0/0/6, ge-0/0/7, ge-0/0/8, ge-0/0/9, ge-0/0/10, ge-0/0/11, ge-0/0/12, ge-0/0/13, ge-0/0/14, ge-0/0/15, ge-0/0/15.0, gr-0/0/0, ip-0/0/0, lsq-0/0/0, mt-0/0/0, lt-0/0/0, ge-0/0/0.3, ge-0/0/0.4

Processors

Routing Engine		7%

Memory Pools

Routing Engine	655.36MB / 1GB	64%

Storage

/junos/cf	203.37MB / 293.04MB	69%
/jail/var	6.12MB / 342.41MB	2%
/jail/var/log	6.12MB / 342.41MB	2%
/config	40kB / 24.42MB	0%
/cf/var	6.12MB / 342.41MB	2%
/	203.37MB / 293.04MB	69%
/mfs	14.62MB / 167.83MB	9%
/mfs/var/run/utm	4kB / 39.23MB	0%
/jail/mfs	4kB / 1.84MB	0%

Temperatures

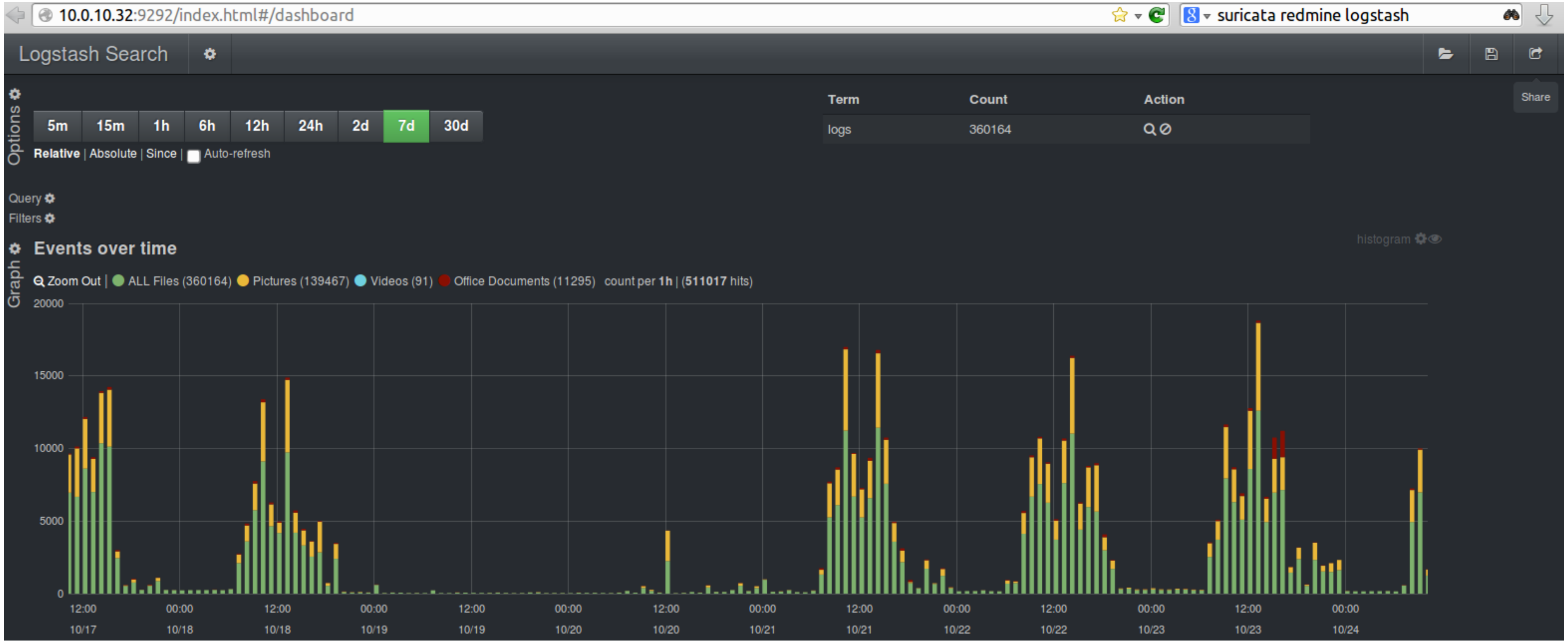
Routing Engine	37°C

Recent Events

07/Feb/11 23:20:06	BGP Session Flap: 2001:1448::102:0:2:40:1 (AS16245)

More useful information than default vendor interface! (flash)

Graphs and Dashboards!


10.0.10.32:9292/index.html#/dashboard

suricata redmine logstash

Logstash Search

Options
5m 15m 1h 6h 12h 24h 2d 7d 30d
Relative | Absolute | Since | □ Auto-refresh
Query
Filters
Share
histogram

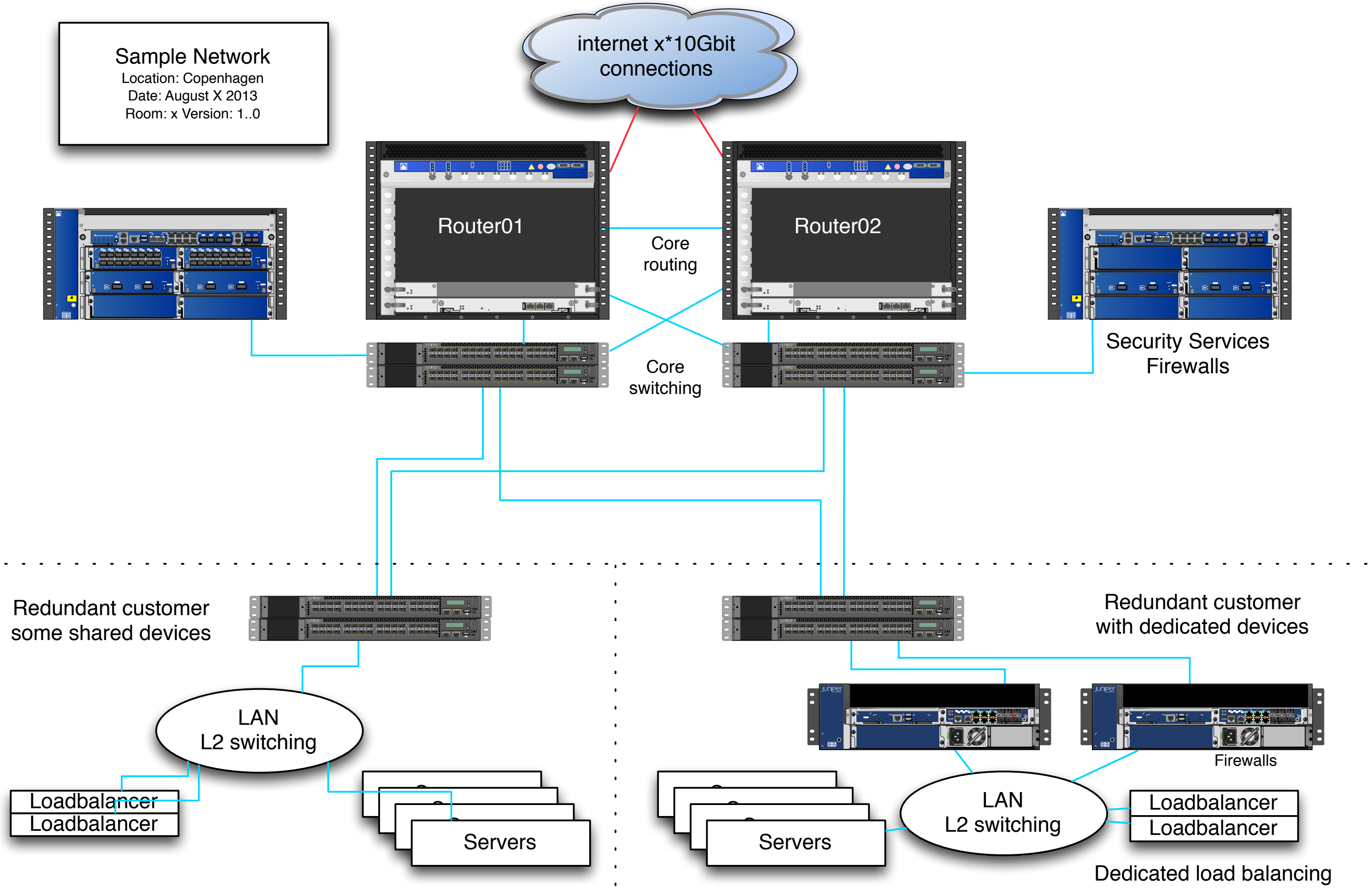
Term	Count	Action
logs	360164	

Events over time

🔍 Zoom Out | ● ALL Files (360164) ● Pictures (139467) ● Videos (91) ● Office Documents (11295) count per **1h** | (**511017** hits)

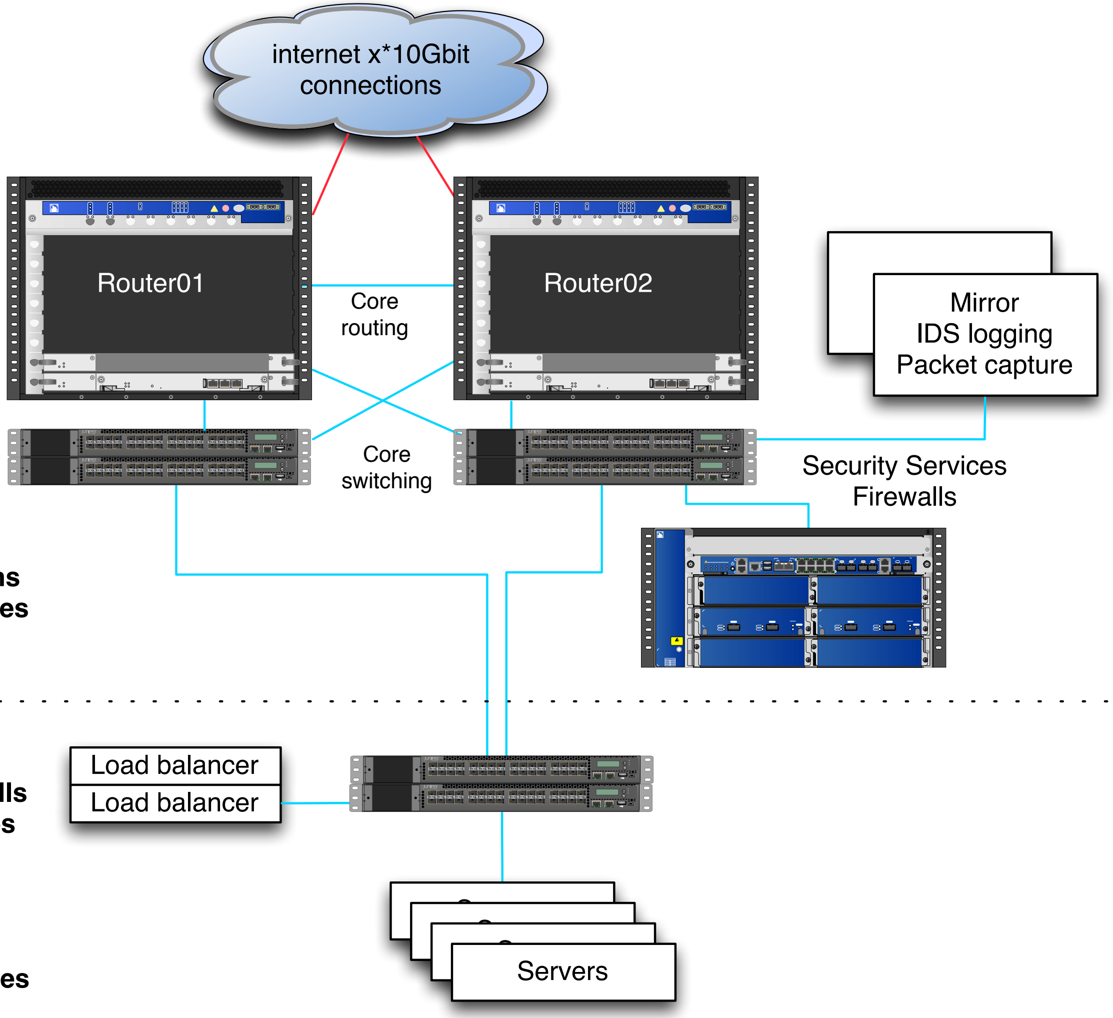
- Screenshot from Peter Manev, OISF
- Shown are Suricata IDS alerts processed by Logstash and Kibana

Networks today

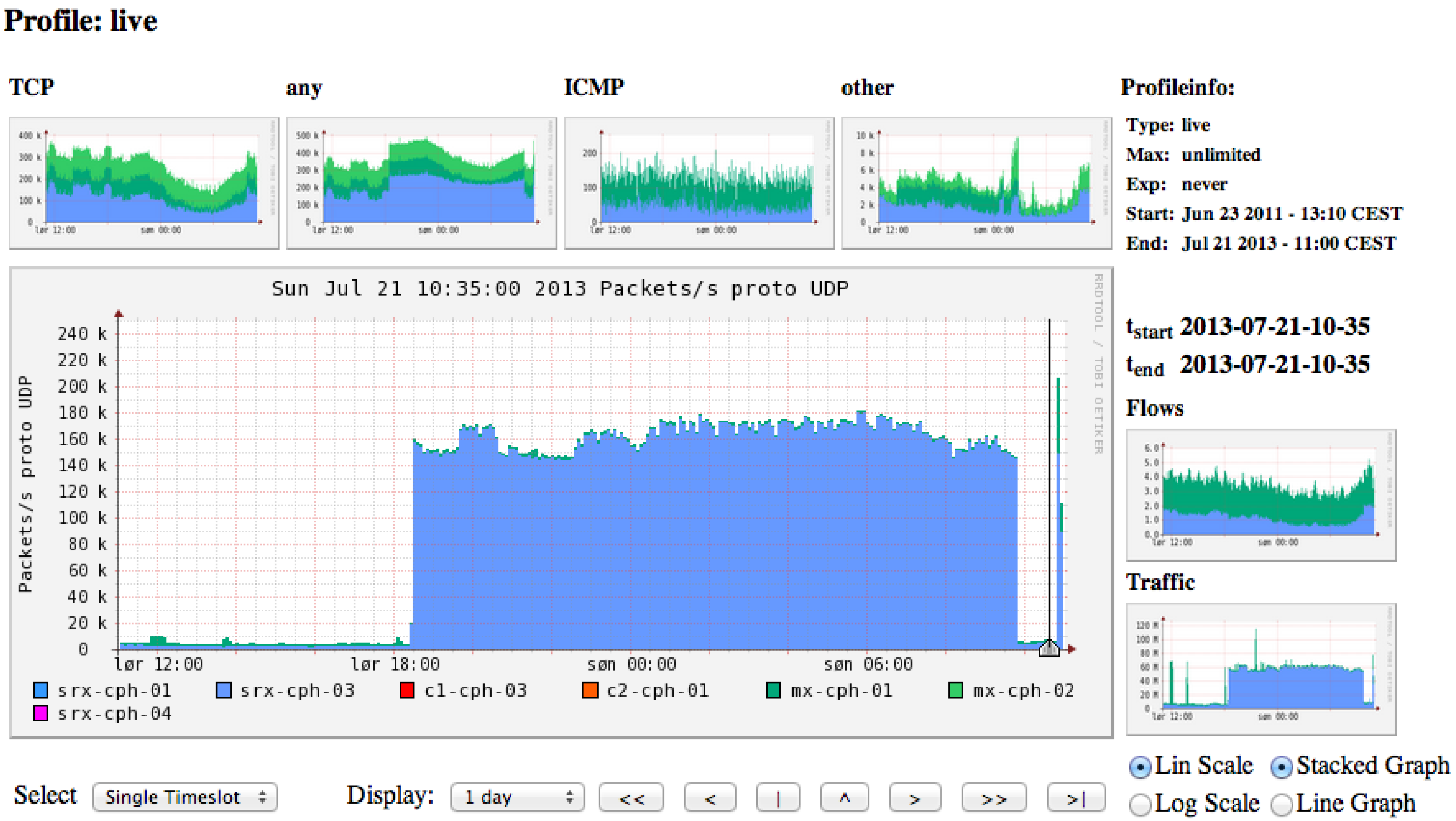


Defense in depth - multiple layers of security

Knowledge about traffic	Proposed actions to be done
	Get more bandwidth
Individual packets	Stateless filtering
Valid source IP ?	Null-routing RTBH Blackholing
Identify traffic by Ports/Protocols	Shaping and fair distribution
Sessions #sessions/IP	Stateful filtering screens IDS/IDP security services
Full request with parameters and cookies	Application tuning Next generation firewalls Load balancer features
Logged in user authenticated?	Server security Input validation
	Stack protection features



Netflow NFSen



Profile: live

Profileinfo:

Type: live
Max: unlimited
Exp: never
Start: Jun 23 2011 - 13:10 CEST
End: Jul 21 2013 - 11:00 CEST

t_{start} 2013-07-21-10-35
t_{end} 2013-07-21-10-35

● Lin Scale ● Stacked Graph
○ Log Scale ○ Line Graph

Select: Single Timeslot

Display: 1 day

An extra 100k packets per second from this netflow source (source is a router)

How to get started

How to get started searching for security events?

Collect basic data from your devices and networks

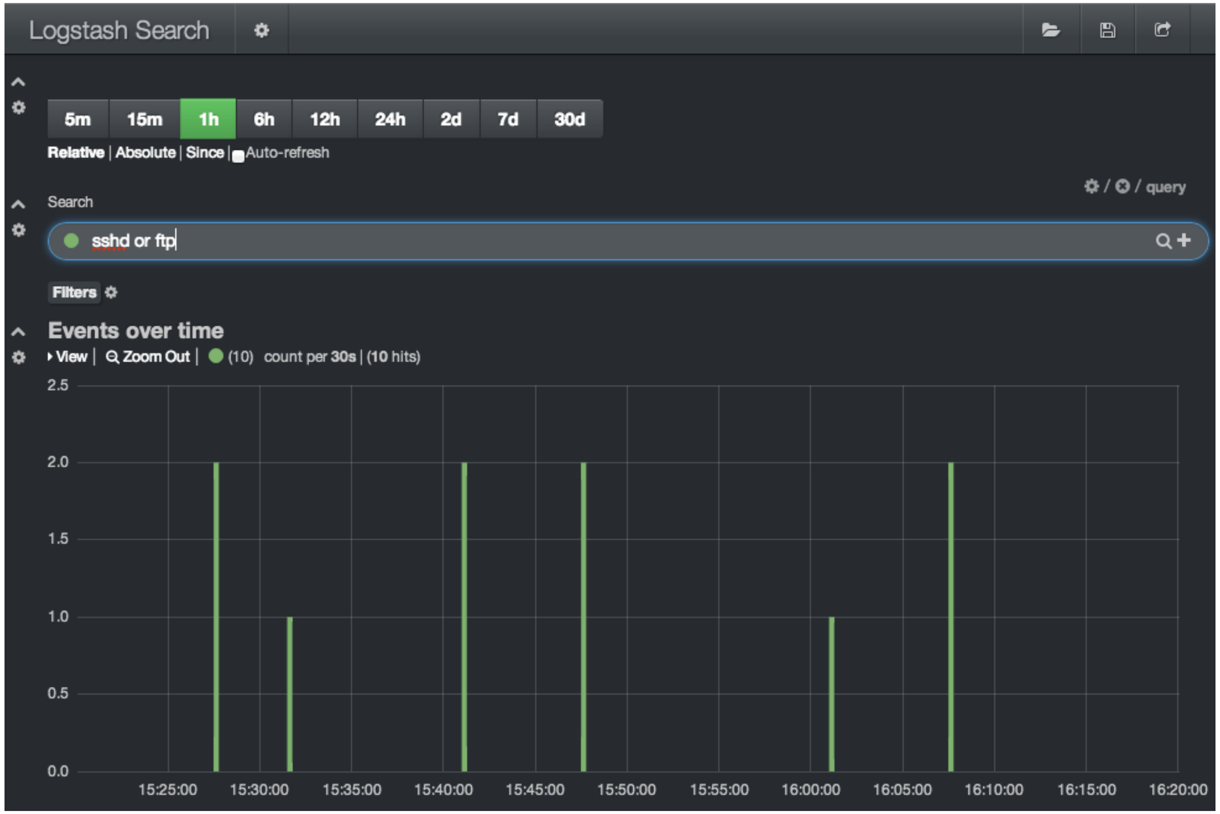
- Netflow data from routers
- Session data from firewalls
- Logging from applications: email, web, proxy systems

Centralize!

Process data

- Top 10: interesting due to high frequency, occurs often, brute-force attacks
- *ignore*
- Bottom 10: least-frequent messages are interesting

View data efficiently



Search

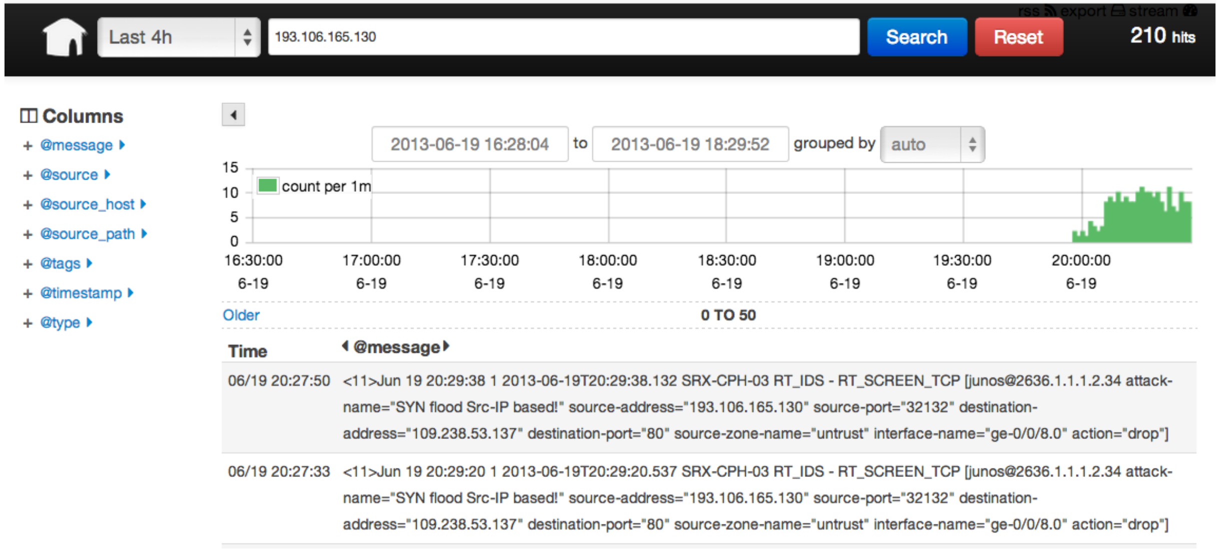
sshd or ftp

● (10) count per 30s | (10 hits)

View data by digging into it easily - must be fast

Logstash and Kibana are just examples, but use indexing to make it fast!

Network tools - examples



Time	@message
06/19 20:27:50	<11>Jun 19 20:29:38 1 2013-06-19T20:29:38.132 SRX-CPH-03 RT_IDS - RT_SCREEN_TCP [junos@2636.1.1.1.2.34 attack-name="SYN flood Src-IP based!" source-address="193.106.165.130" source-port="32132" destination-address="109.238.53.137" destination-port="80" source-zone-name="untrust" interface-name="ge-0/0/8.0" action="drop"]
06/19 20:27:33	<11>Jun 19 20:29:20 1 2013-06-19T20:29:20.537 SRX-CPH-03 RT_IDS - RT_SCREEN_TCP [junos@2636.1.1.1.2.34 attack-name="SYN flood Src-IP based!" source-address="193.106.165.130" source-port="32132" destination-address="109.238.53.137" destination-port="80" source-zone-name="untrust" interface-name="ge-0/0/8.0" action="drop"]

Net: Bro `http://www.bro-ids.org` Suricata `http://suricata-ids.org`

DNS: DSC and PacketQ `https://github.com/dotse/packetq/wiki`

Syslog: Elasticsearch, Logstash, and Kibana

Packetbeat `https://www.elastic.co/products/beats/packetbeat`

Collect and present data more easily - non-programmers

Security devops

We need devops skillz in security - automate, security is also big data

integrate tools, transfer, sort, search, pattern matching, statistics, ...

tools, languages, databases, protocols, data formats

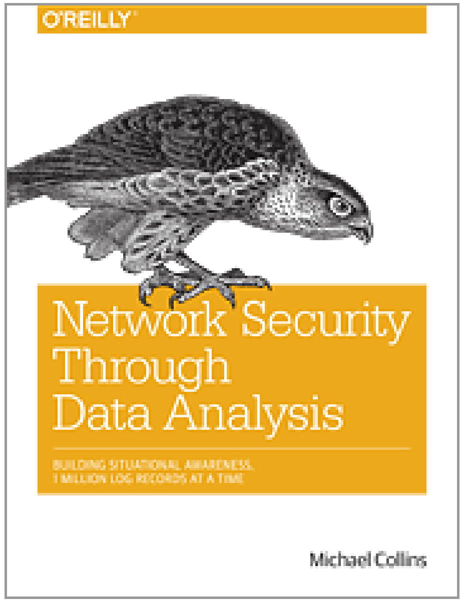
Example introductions:

- Seven languages/database/web frameworks in Seven Weeks
- Elasticsearch the definitive guide `http://www.elastic.co/guide/en/elasticsearch/guide/current/index.html`
- `https://www.elastic.co/products/kibana`
- `https://www.elastic.co/products/logstash`

We are all Devops now, even security people!

Do you even Github? ☺`https://github.com/stars`

Network Security Through Data Analysis



O'REILLY

Network Security Through Data Analysis

BUILDING SITUATIONAL AWARENESS, 1 MILLION LOG RECORDS AT A TIME

Michael Collins

Low page count, but high value! Recommended.

Network Security Through Data Analysis: Building Situational Awareness
By Michael Collins
Publisher: O'Reilly Media Released: February 2014 Pages: 348

BRO IDS

The Bro Network Security Monitor

Bro is a powerful network analysis framework that is much different from the typical IDS you may know.

While focusing on network security monitoring, Bro provides a comprehensive platform for more general network traffic analysis as well. Well grounded in more than 15 years of research, Bro has successfully bridged the traditional gap between academia and operations since its inception.

`https://www.bro.org/`

BRO more than an IDS

> The key point that helped me understand was the explanation that Bro is a domain-specific language for networking applications and that Bro-IDS (http://bro-ids.org/) is an application written with Bro.

Why I think you should try Bro

`https://isc.sans.edu/diary.html?storyid=15259`

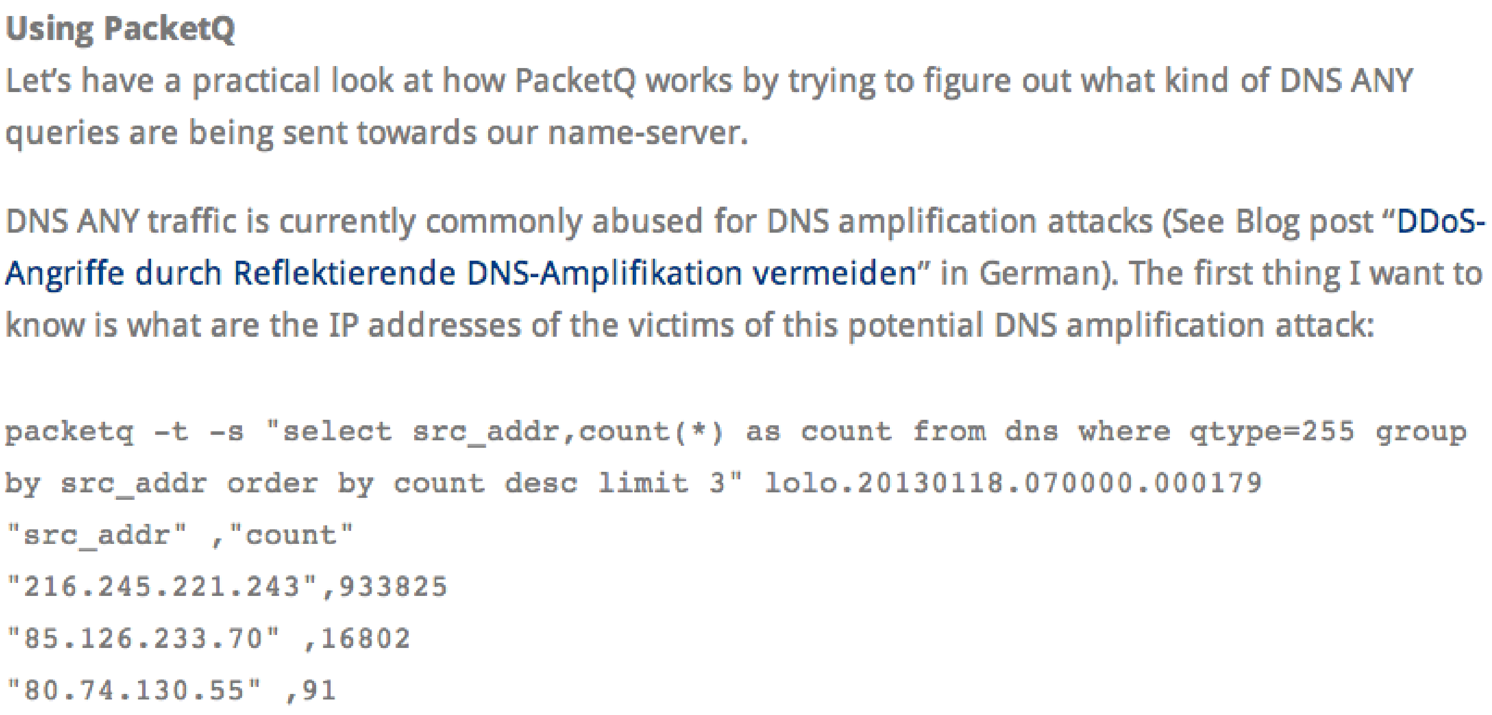
Bro scripts

```
global dns_A_reply_count=0;
global dns_AAAA_reply_count=0;
...
event dns_A_reply(c: connection, msg: dns_msg, ans: dns_answer, a: addr)

++dns_A_reply_count;


event dns_AAAA_reply(c: connection, msg: dns_msg, ans: dns_answer, a: addr)

++dns_AAAA_reply_count;
```

Source: dns-fire-count.bro from

`https://github.com/LiamRandall/bro-scripts/tree/master/fire-scripts`

Example, Using tools similar to PacketQ



Using PacketQ

Let's have a practical look at how PacketQ works by trying to figure out what kind of DNS ANY queries are being sent towards our name-server.

DNS ANY traffic is currently commonly abused for DNS amplification attacks (See Blog post "DDoS-Angriffe durch Reflektierende DNS-Amplifikation vermeiden" in German). The first thing I want to know is what are the IP addresses of the victims of this potential DNS amplification attack:

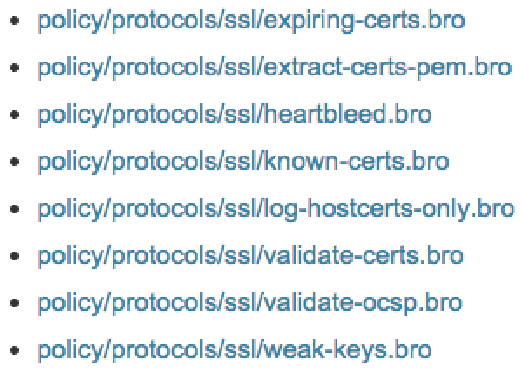
```
packetq -t -s "select src_addr,count(*) as count from dns where qtype=255 group
by src_addr order by count desc limit 3" lolo.20130118.070000.000179
"src_addr" ,"count"
"216.245.221.243",933825
"85.126.233.70" ,16802
"80.74.130.55" ,91
```

Are you using your brain and existing tools? Building own specialised tools?
Discussion: bridging the gaps between Devops and Security? Good thing, easy?

`http://securityblog.switch.ch/2013/01/22/using-packetq/`

`http://jpmens.net/2013/05/27/server-agnostic-logging-of-dns-queries-responses/`

Storing query logs, old school or needed?



- policy/protocols/ssl/expiring-certs.bro
- policy/protocols/ssl/extract-certs-pem.bro
- policy/protocols/ssl/heartbleed.bro
- policy/protocols/ssl/known-certs.bro
- policy/protocols/ssl/log-hostcerts-only.bro
- policy/protocols/ssl/validate-certs.bro
- policy/protocols/ssl/validate-ocsp.bro
- policy/protocols/ssl/weak-keys.bro

Looking at DNS PacketQ it was an Older link, but thinking the time is now for doing:

- DNS query logs, keep it for at least a week? - with DSC and PacketQ
- SSL/TLS full logs over sessions, certs, keys - with Bro/Suricata
  `https://www.bro.org/sphinx-git/script-reference/scripts.html`
- Log and search with Elasticsearch?
  `https://www.elastic.co/guide/en/elasticsearch/guide/current/index.html`
- Even netflow session logging, full 1:1 - NFSen, Suricata Flow mode?

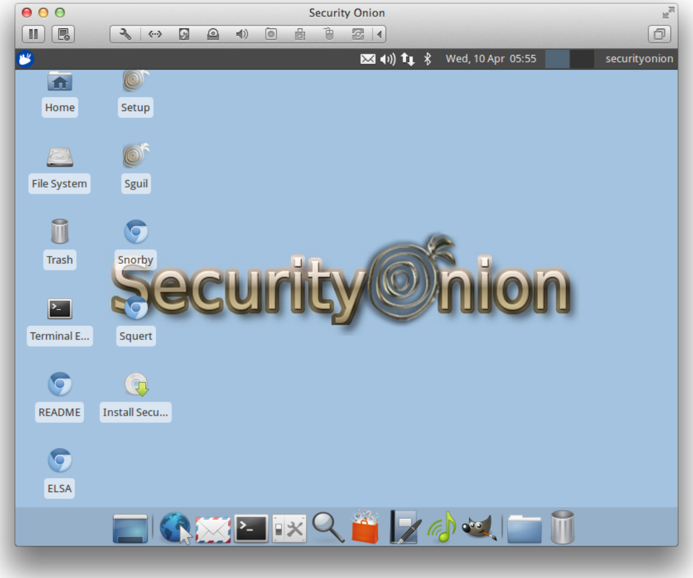
February 2015: Finding infected sources

> ”We were contacted by a client to help with their incident response in tracking down an infection on a clients machine with the new CTB-Locker ransomware (Curve-Tor-Bitcoin Locker) aka Critroni which had no signatures available at the time of infection for this variant.
>
> LANGuardian includes a file share activity monitoring module which provided a very detailed forensic analysis of the ransomware and the paths it had taken in order to encrypt the clients system and also the fileserver in which it was connected to, the initial infection came from the opening of an attachment in an e-mail.”

It has become critical to identify vulnerable or infected ASAP!

Source: `https://www.netfort.com/support-team-stories-detecting-the-source-of-ransomware/`

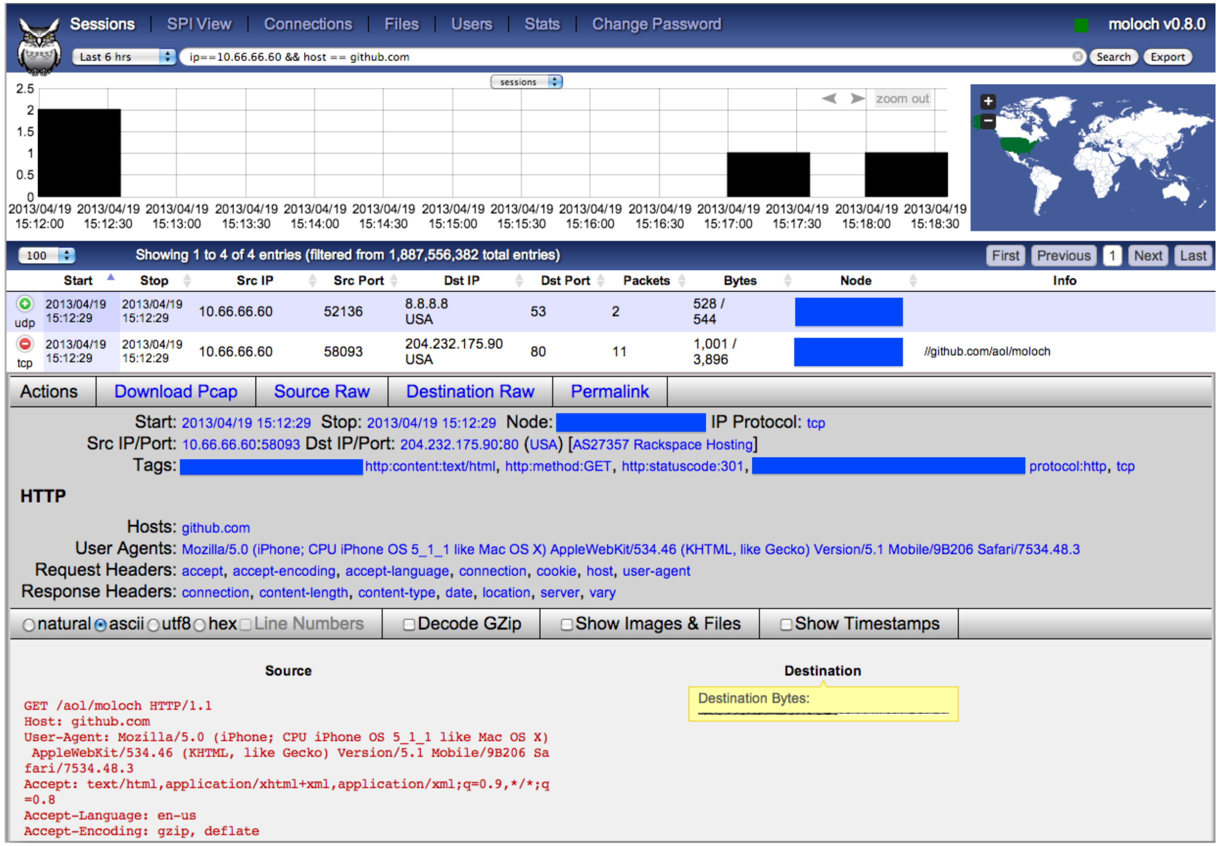
Security Onion



- Security Onion is a Linux distro for IDS, NSM, and log management
- Learn NSM with Security Onion today - its free

Nice starting point for researching dashboards/network packets

Moloch



ip==10.66.66.60 && host == github.com

Showing 1 to 4 of 4 entries (filtered from 1,887,556,382 total entries)

	Start	Stop	Src IP	Src Port	Dst IP	Dst Port	Packets	Bytes	Node	Info
udp	2013/04/19 15:12:29	2013/04/19 15:12:29	10.66.66.60	52136	8.8.8.8 USA	53	2	528 / 544	[redacted]	
tcp	2013/04/19 15:12:29	2013/04/19 15:12:29	10.66.66.60	58093	204.232.175.90 USA	80	11	1,001 / 3,896	[redacted]	//github.com/aol/moloch

Actions | Download Pcap | Source Raw | Destination Raw | Permalink

Start: 2013/04/19 15:12:29 Stop: 2013/04/19 15:12:29 Node: [redacted] IP Protocol: tcp
Src IP/Port: 10.66.66.60:58093 Dst IP/Port: 204.232.175.90:80 (USA) [AS27357 Rackspace Hosting]
Tags: [redacted] http:content:text/html, http:method:GET, http:statuscode:301, [redacted] protocol:http, tcp

HTTP

Hosts: github.com
User Agents: Mozilla/5.0 (iPhone; CPU iPhone OS 5_1_1 like Mac OS X) AppleWebKit/534.46 (KHTML, like Gecko) Version/5.1 Mobile/9B206 Safari/7534.48.3
Request Headers: accept, accept-encoding, accept-language, connection, cookie, host, user-agent
Response Headers: connection, content-length, content-type, date, location, server, vary

○natural ●ascii ○utf8 ○hex □Line Numbers | □Decode GZip | □Show Images & Files | □Show Timestamps

Source

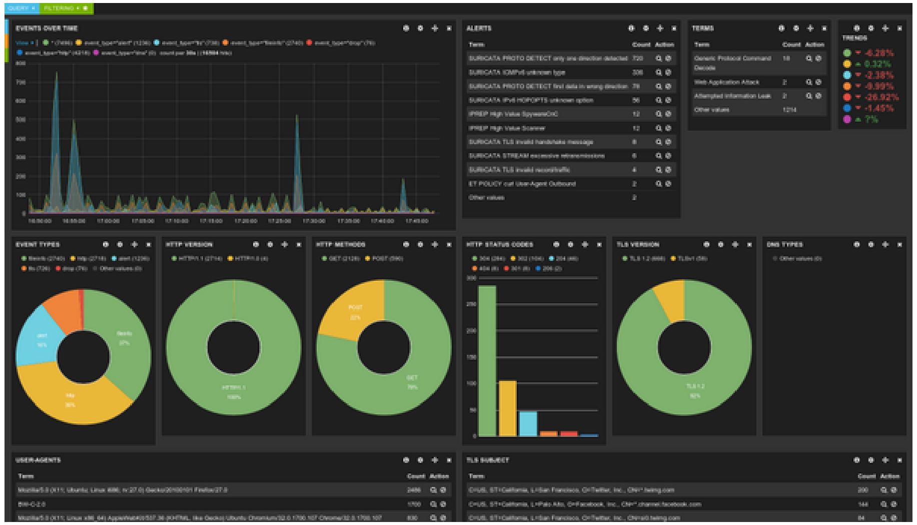
```
GET /aol/moloch HTTP/1.1
Host: github.com
User-Agent: Mozilla/5.0 (iPhone; CPU iPhone OS 5_1_1 like Mac OS X)
 AppleWebKit/534.46 (KHTML, like Gecko) Version/5.1 Mobile/9B206 Sa
fari/7534.48.3
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q
=0.8
Accept-Language: en-us
Accept-Encoding: gzip, deflate
```

Destination

Destination Bytes:

Picture from `https://github.com/aol/moloch`

Suricata with Dashboards



ALERTS

Term	Count	Action
SURICATA PROTO DETECT only one direction detected	720	
SURICATA ICMPv6 unknown type	306	
SURICATA PROTO DETECT first data in wrong direction	78	
SURICATA IPv6 HOPOPTS unknown option	56	
IPREP High Value SpywareCnC	12	
IPREP High Value Scanner	12	
SURICATA TLS invalid handshake message	8	
SURICATA STREAM excessive retransmissions	8	
SURICATA TLS invalid record/traffic	4	
ET POLICY curl User-Agent Outbound	2	
Other values	2	

TERMS

Term	Count	Action
Generic Protocol Command Decode	18	
Web Application Attack	2	
Attempted Information Leak	2	
Other values	1214	

USER-AGENTS

Term	Count	Action
Mozilla/5.0 (X11; Ubuntu; Linux i686; rv:27.0) Gecko/20100101 Firefox/27.0	2486	
BW-C-2.0	1700	
Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Ubuntu Chromium/32.0.1700.107 Chrome/32.0.1700.107	830	

TLS SUBJECT

Term	Count	Action
C=US, ST=California, L=San Francisco, O=Twitter, Inc., CN=*.twimg.com	200	
C=US, ST=California, L=Palo Alto, O=Facebook, Inc., CN=*.channel.facebook.com	144	
C=US, ST=California, L=San Francisco, O=Twitter, Inc., CN=si0.twimg.com	84	

Picture from Twitter

`https://twitter.com/nullthreat/status/445969209840128000`

New link March 2014: 10Gbits

`http://pevma.blogspot.se/2014/03/suricata-prepearing-10gbps-network.html`

`http://suricata-ids.org/2014/03/25/suricata-2-0-available/`

Big Data tools: Elasticsearch

elasticsearch

the definitive guide

clinton gormley zachary tong Copyright © 2014 Elasticsearch

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivs 3.0 Unported License.

`http://www.elasticsearch.org/guide/en/elasticsearch/guide/current/index.html`

`http://www.elasticsearch.org/overview/kibana/`

`http://www.elasticsearch.org/overview/logstash/`

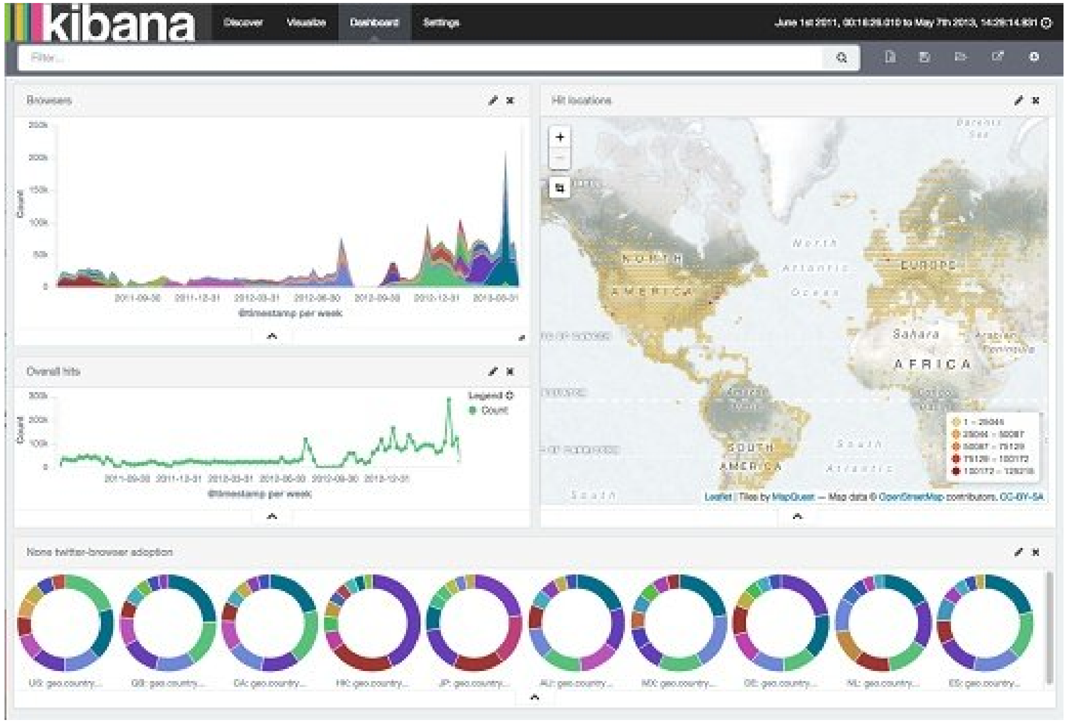
We are all Devops now, even security people!

Ansible configuration management

```
- apt: name= item  state=latest
  with_items:
        - unzip
        - elasticsearch
        - logstash
        - redis-server
        - nginx
- lineinfile: "dest=/etc/elasticsearch/elasticsearch.yml state=present
  regexp='script.disable_dynamic: true' line='script.disable_dynamic: true'"
- lineinfile: "dest=/etc/elasticsearch/elasticsearch.yml state=present
  regexp='network.host: localhost' line='network.host: localhost'"
- name: Move elasticsearch data into /data
  command: creates=/data/elasticsearch mv /var/lib/elasticsearch /data/
- name: Make link to /data/elasticsearch
  file: state=link src=/data/elasticsearch path=/var/lib/elasticsearch
```

only requires SSH+python `http://www.ansible.com`

Kibana 4 february 2015



Highly recommended for a lot of data visualisation

Non-programmers can create, save, and share dashboards

Source: `https://www.elastic.co/products/kibana`

Lets get to work!

- Get Kibana working
- Get access to Kibana
- Produce some data
- Create dashboards

While demoing Ansible, and vagrant

Lots of examples
`https://github.com/geerlingguy/ansible-vagrant-examples/`

Next steps

In our network we are always improving things:

Suricata IDS `http://www.openinfosecfoundation.org/`

More graphs, with **automatic identification** of IPs under attack

Identification of **short sessions without data** - spoofed addresses

Alerting from **existing** devices

Dashboards with key measurements

Conclusion: Combine tools!

Logstash pipeline

```
input { stdin { } }
output {
  elasticsearch { host => localhost }
  stdout { codec => rubydebug }
}
```

Source: Config snippet from recommended link

`http://logstash.net/docs/1.4.1/tutorials/getting-started-with-logstash`

- Logstash receives via **input**
- Processes with **filters** - grok
- Forward events with **output**

Logstash as SNMPtrap and syslog server

```
input {
  snmptrap {
    host => "0.0.0.0"
    type => "snmptrap"
    port => 1062
    community => "xxxxx"
  }
  tcp {
    port => 5000
    type => syslog
  }
  udp {
    port => 5000
    type => syslog
  }
}
```

- We run logstash on port 5000 - but use IPtables port forwarding

IPtables forwarding

```
*nat
:PREROUTING ACCEPT [0:0]
# redirect all incoming requests on port 514 to port 5000
-A PREROUTING -p tcp --dport 514 -j REDIRECT --to-port 5000
-A PREROUTING -p udp --dport 514 -j REDIRECT --to-port 5000
-A PREROUTING -p udp --dport 162 -j REDIRECT --to-port 1062
COMMIT
```

Inserted near beginning of /etc/ufw/before.rules on Ubuntu

Grok expresssions

```
filter {
  if [type] == "syslog" {
    grok {
      match => { "message" => "%{SYSLOGTIMESTAMP:syslog_timestamp}
      %{SYSLOGHOST:syslog_hostname} %{DATA:syslog_program}
      (?:\[%{POSINT:syslog_pid}\])?: %{GREEDYDATA:syslog_message}" }
      add_field => [ "received_at", "%{@timestamp}" ]
      add_field => [ "received_from", "%{host}" ]
    }
    syslog_pri { }
    date {
      match => [ "syslog_timestamp", "MMM  d HH:mm:ss", "MMM dd HH:mm:ss" ]
    }
  }
}
```

- Logstash filter expressions grok can normalize and split data into fields

Source: Config snippet from recommended link

`http://logstash.net/docs/1.4.1/tutorials/getting-started-with-logstash`

Grok expresssions, sample from my archive

```
filter {
# decode some SSHD
if [syslog_program] == "sshd" {
  grok {
# May 20 10:27:08 odn1-nsm-01 sshd[4554]: Accepted publickey for hlk from
10.50.11.17 port 50365 ssh2: DSA 9e:fd:3b:3d:fc:11:0e:b9:bd:22:71:a9:36:d8:06:c7

match => { "message" => "%{SYSLOGTIMESTAMP:timestamp} %{HOSTNAME:host_target}
sshd\[%{BASE10NUM}\]: Accepted publickey for %{USERNAME:username} from
  %{IP:src_ip} port %{BASE10NUM:port} ssh2" }

# "May 20 10:27:08 odn1-nsm-01 sshd[4554]: pam_unix(sshd:session):
session opened for user hlk by (uid=0)"
match => { "message" => "%{SYSLOGTIMESTAMP:timestamp} %{HOSTNAME:host_target}
sshd\[%{BASE10NUM}\]: pam_unix\(sshd:session\): session opened for user
%{USERNAME:username}" }
```

- Logstash filter expressions grok can normalize and split data into fields

Questions?

Henrik Lund Kramshøj hlk@kramse.org

THECAMP.DK - 7 open source days