Welcome to

Infrastruktursikkerhed og hacking

Passende Paranoia for IT-folk

Henrik Lund Kramshøj hlk@zencurity.dk

Slides are available as PDF, kramshoej@Github

Try searching for svanninge-bjerge.tex in the repo

Goals: Infrastructure security, what is that?

Pentest introduction, hacking introduction

Using hacker tools for protection

Defense in depth using big data - inspiration

Please give feedback and join me in discussions, dialogue ☺

Agenda for today

Kl 15:00-18:00 with breaks

Less presentation, more talk, towards the end

Trying to fit in demo and workshop-like stuff

Generic advice

Recommendations

- Lock your devices, phones, tables and computers
- Update software and apps
- Do NOT use the same password everywhere
- Watch out when using open wifi-networks
- Multiple browsers: one for Facebook, one for banking apps?
- Multiple laptops? One for private data, one for work?
- Think of the data you produce - where is it stored
- Use pseudonyms and aliases, do not use your real name everywhere
- Enable encryption: IMAP**S**, POP3**S**, HTTP**S** and full disk encryption
- Use Tor `http://torproject.org/`

The current situation

Internet security sucks, 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

New vulnerabilities, while we are already dealing with those from yesterday

Challenges

Less resources available for IT and infosec

Lots of new malware, virus, vulnerabilities and hacking

Dataloss ransomware, theft

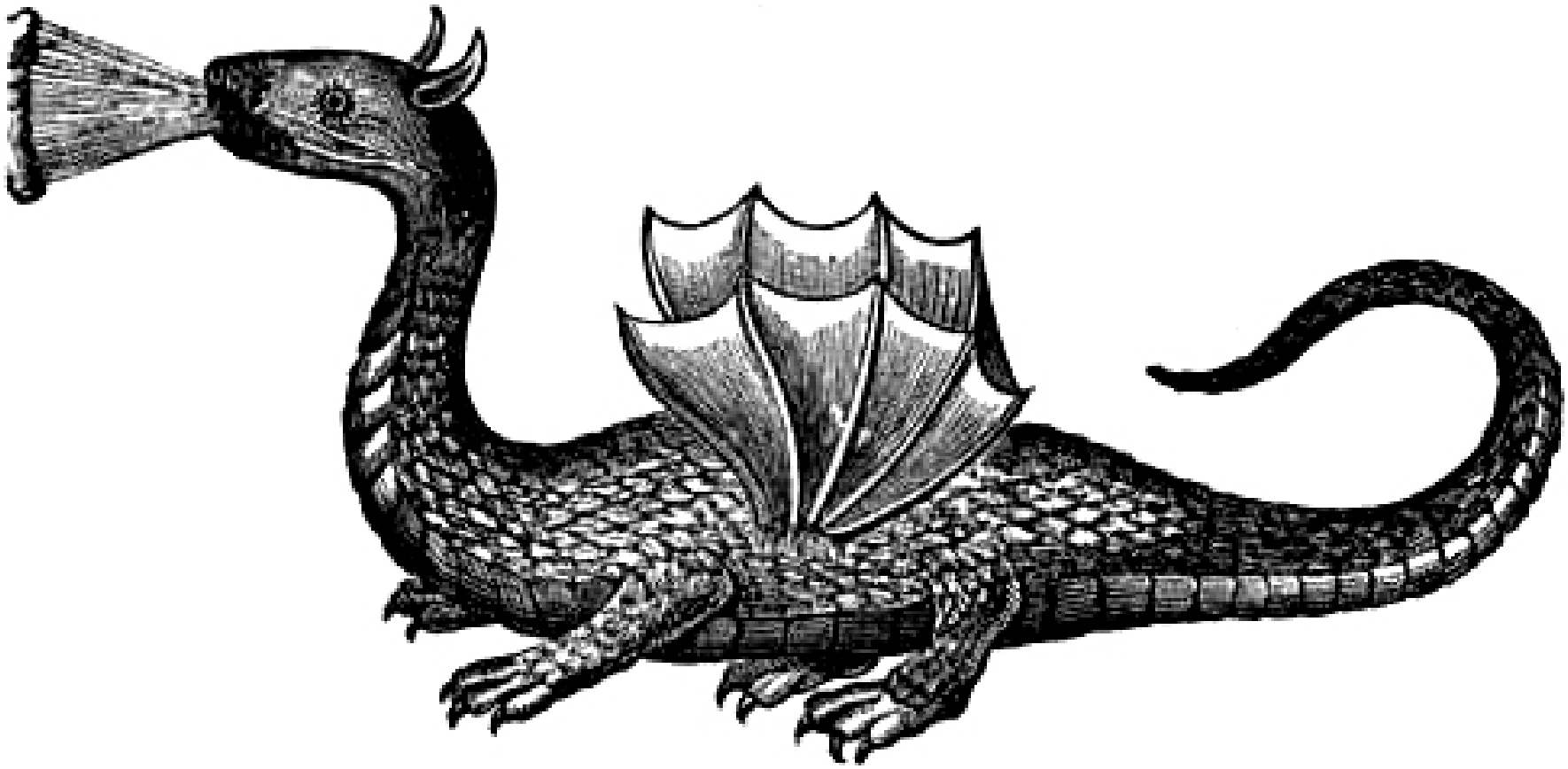
Loss of confidentiality, 2014: 700 million lost accounts, now even more

Yahoo Says 1 Billion User Accounts Were Hacked, December 2016
`https://www.nytimes.com/2016/12/14/technology/yahoo-hack.html`

Infosec charlatans, hype and lies

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

Internet - Here be dragons



Solutions

Automate your job, Ansible is our preferred tool

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

Use hackertools to detect and identify

Categorise, sort, prioritize, group problems - solve more

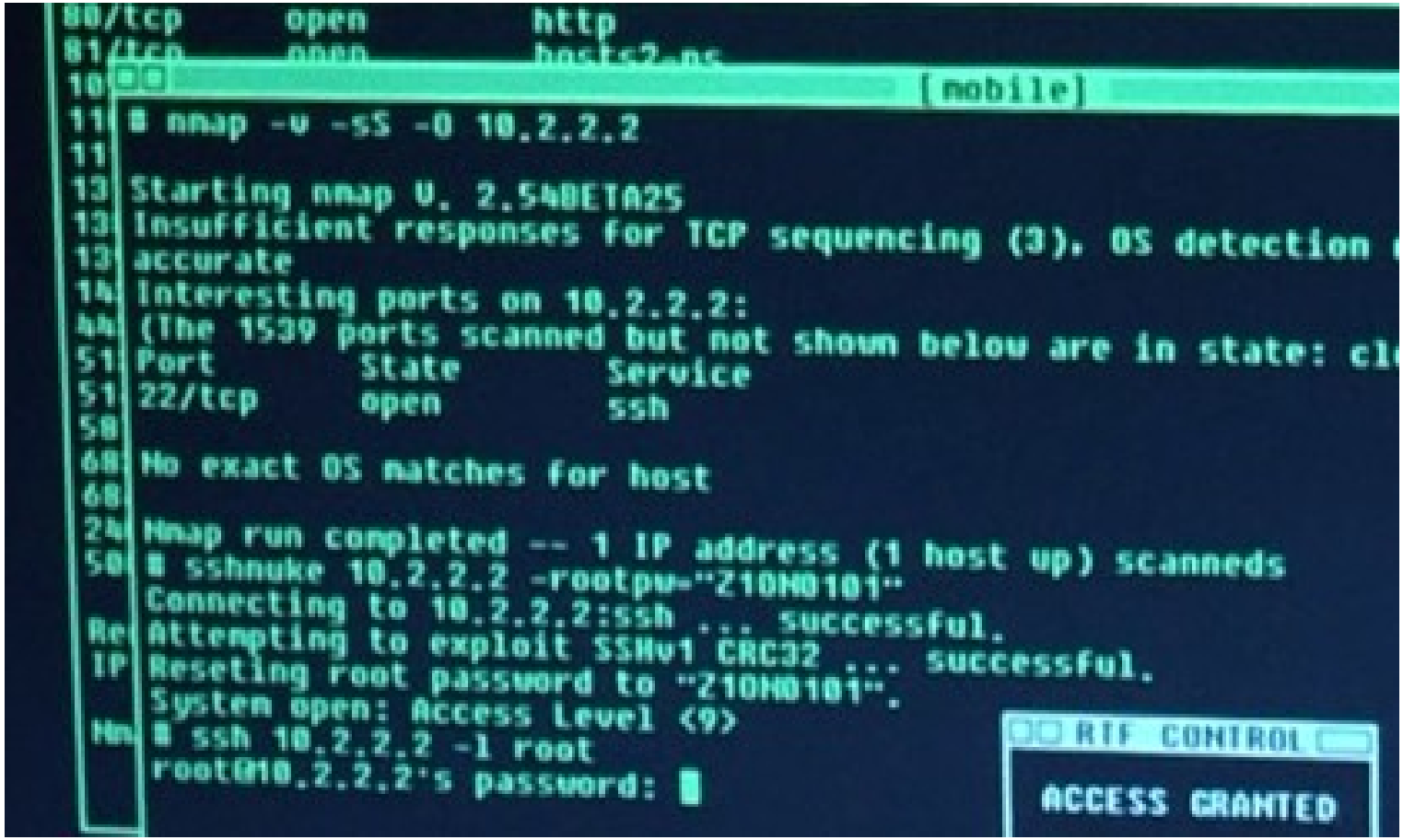
Measure, collect and present - make it pretty

Learn from devops, Elasticsearch Logstash Kibana D3.js

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

Matrix style hacking anno 2003

Trinity breaking in



`http://nmap.org/movies.html`

Meget realistisk `https://www.youtube.com/watch?v=0PxTAn4g20U`

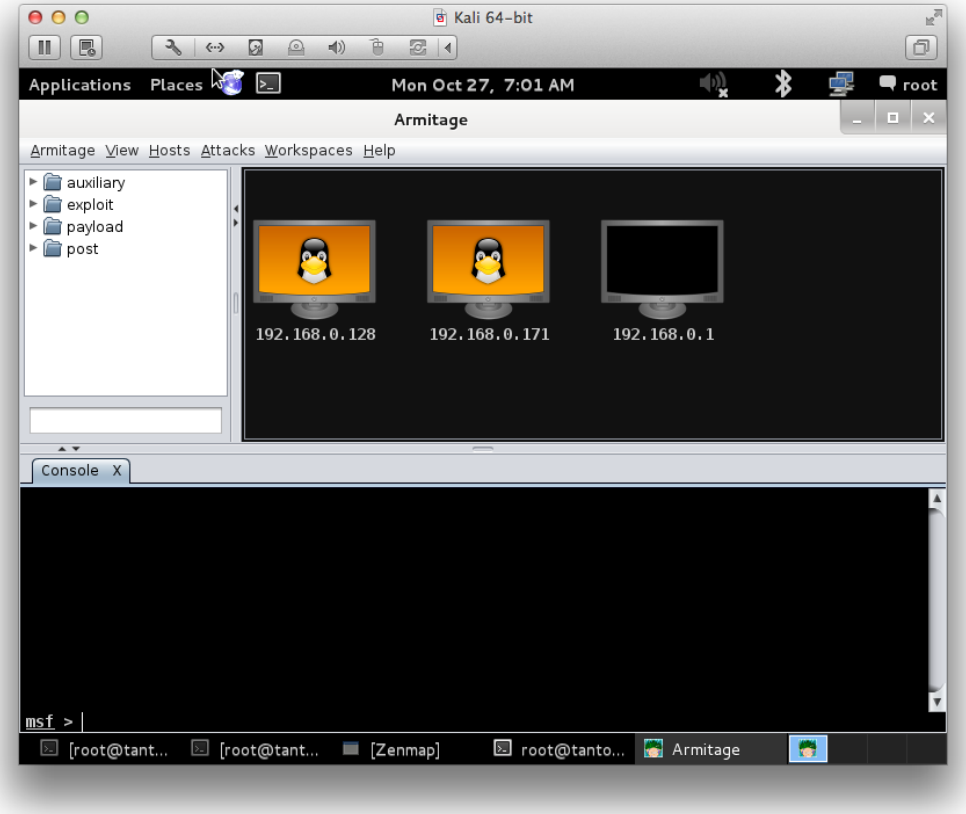
Demo: Wireless hacking

But before we get to the tools, lets try something

Typical 802.11 based wireless network with Access-Point (AP)

Using a TP-LINK TL-WN722N Wireless USB Adapter
cheap and great for demo - only 2.4GHz

Demo: Hacking by pressing enter



Even script kiddies today can download the newest exploits and run them

Paranoia defined

par·a·noi·a

/ˌparəˈnoiə/

noun

noun: **paranoia**

1. a mental condition characterized by delusions of persecution, unwarranted jealousy, or exaggerated self-importance, typically elaborated into an organized system. It may be an aspect of chronic personality disorder, of drug abuse, or of a serious condition such as schizophrenia in which the person loses touch with reality.
   synonyms: persecution complex, delusions, obsession, psychosis More
- suspicion and mistrust of people or their actions without evidence or justification.
  "the global paranoia about hackers and viruses"

Origin

GREEK
para
irregular

GREEK
noos
mind

GREEK
paranoos
distracted

MODERN LATIN
paranoia
early 19th cent.

More

Source: google paranoia definition

Face reality

From the definition:

> suspicion and mistrust of people or their actions **without evidence or justification**. **”the global paranoia about hackers and viruses”**

It is not paranoia when:

- Criminals sell your credit card information and identity theft
- Trade infected computers like a commodity
- Governments write laws that allows them to introduce back-doors - and use these
- Governments do blanket surveillance of their population
- Governments implement censorship, threaten citizens and journalist

You are not paranoid when there are people actively attacking you!

Security is not magic

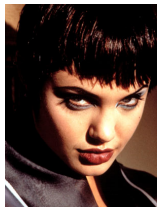
.

Think security, it may seem like magic - but it is not

Follow news about security

Support communities, join and learn

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/`

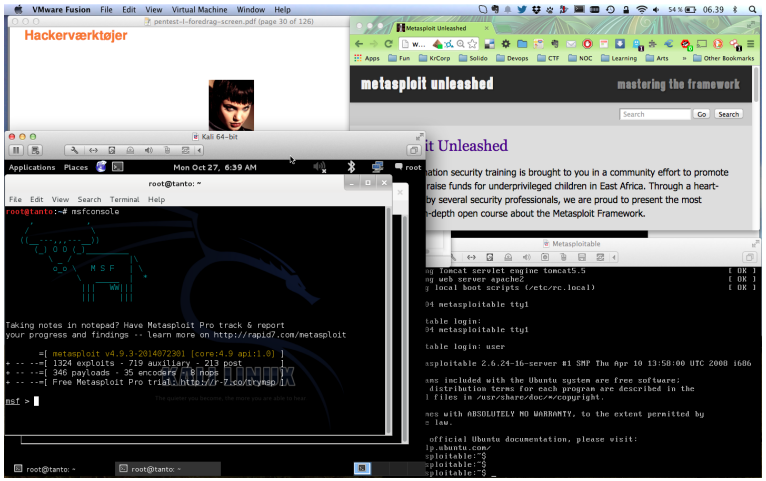
Kali Linux the pentest toolbox

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

100.000s of videos on youtube alone, searching for kali and $TOOL

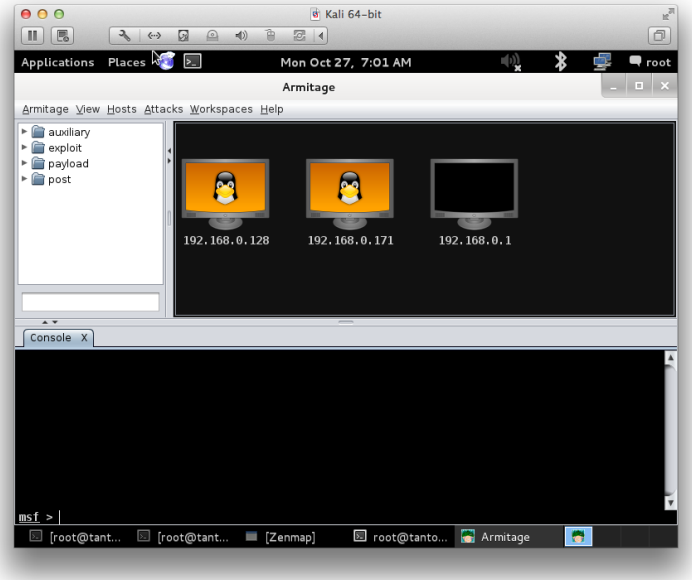
Also versions for Raspberry Pi, mobile and other small computers

Hackerlab opsætning



- Hardware: en moderne laptop med CPU der kan bruge virtualisering
  Husk at slå virtualisering til i BIOS
- Software: dit favoritoperativsystem, Windows, Mac, Linux
- Virtualiseringssoftware: VMware, Virtual box, vælg selv
- Hackersoftware: Kali som Virtual Machine `https://www.kali.org/`
- Soft targets: Metasploitable, Windows 2000, Windows XP, ...

Metasploit and Armitage Still rocking the internet

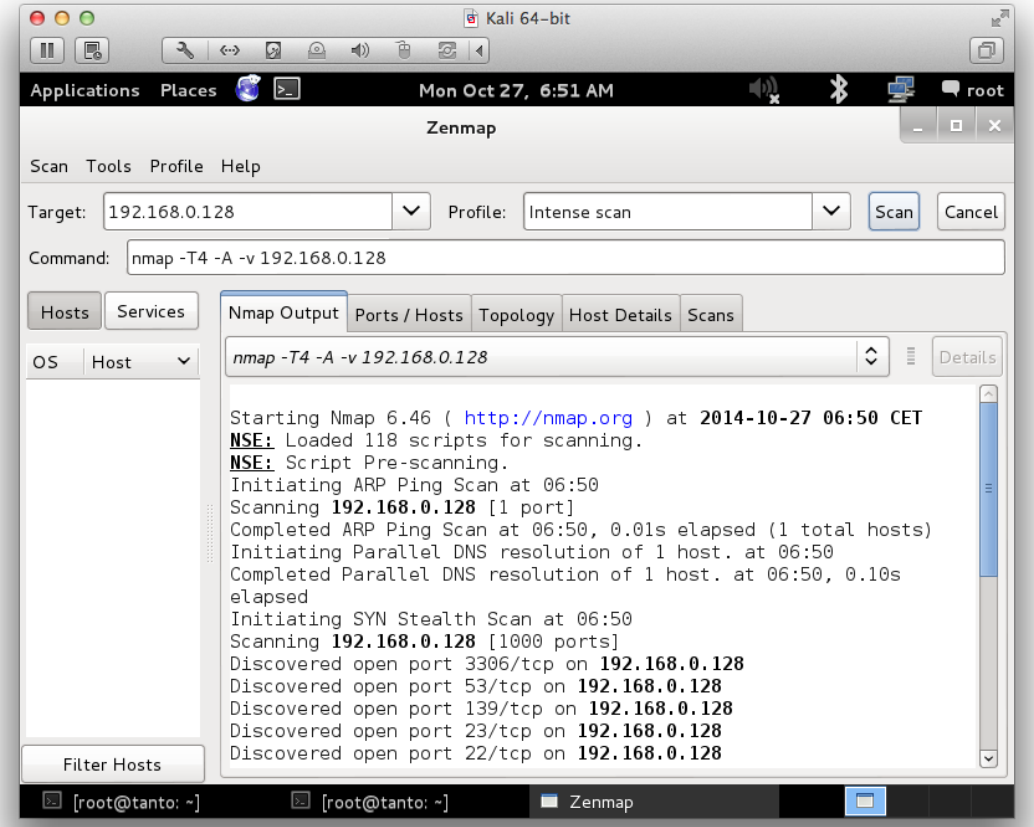


`http://www.metasploit.com/`

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`

Portscan med Zenmap GUI



Zenmap følger med i pakken når man henter Nmap `https://nmap.org`

Nmap port sweep efter webservere

```
root@cornerstone:~#  nmap -p80,443 172.29.0.0/24

Starting Nmap 6.47 ( http://nmap.org ) at 2015-02-05 07:31 CET
Nmap scan report for 172.29.0.1
Host is up (0.00016s latency).
PORT    STATE    SERVICE
80/tcp  open     http
443/tcp filtered https
MAC Address: 00:50:56:C0:00:08 (VMware)

Nmap scan report for 172.29.0.138
Host is up (0.00012s latency).
PORT    STATE  SERVICE
80/tcp  open   http
443/tcp closed https
MAC Address: 00:0C:29:46:22:FB (VMware)
```

Nmap port sweep efter SNMP port 161/UDP

```
root@cornerstone:~# nmap -sU -p 161 172.29.0.0/24
Starting Nmap 6.47 ( http://nmap.org ) at 2015-02-05 07:30 CET
Nmap scan report for 172.29.0.1
Host is up (0.00015s latency).
PORT    STATE         SERVICE
161/udp open|filtered snmp
MAC Address: 00:50:56:C0:00:08 (VMware)

Nmap scan report for 172.29.0.138
Host is up (0.00011s latency).
PORT    STATE  SERVICE
161/udp closed snmp
MAC Address: 00:0C:29:46:22:FB (VMware)
...
Nmap done: 256 IP addresses (5 hosts up) scanned in 2.18 seconds
```

Scan for Heartbleed and SSLv2/SSLv3

Example Usage

```
nmap -sV -sC <target>
```

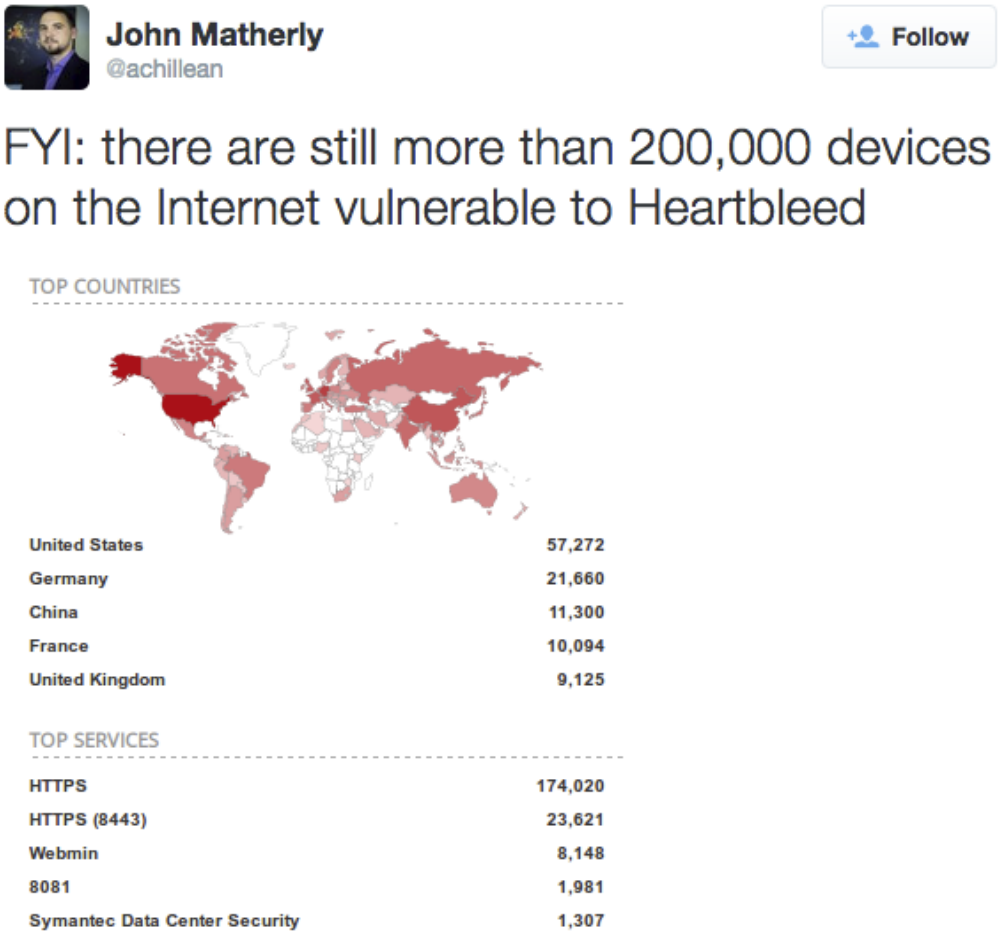
Script Output

```
443/tcp open   https   syn-ack
| sslv2:
|   SSLv2 supported
|   ciphers:
|     SSL2_DES_192_EDE3_CBC_WITH_MD5
|     SSL2_IDEA_128_CBC_WITH_MD5
|     SSL2_RC2_CBC_128_CBC_WITH_MD5
|     SSL2_RC4_128_WITH_MD5
|     SSL2_DES_64_CBC_WITH_MD5
|     SSL2_RC2_CBC_128_CBC_WITH_MD5
|_    SSL2_RC4_128_EXPORT40_WITH_MD5
```

```
nmap -p 443 --script ssl-heartbleed <target>
https://nmap.org/nsedoc/scripts/ssl-heartbleed.html

masscan 0.0.0.0/0 -p0-65535  --heartbleed
https://github.com/robertdavidgraham/masscan
```

September 2015: Heartbleed vulnerable servers



John Matherly
@achillean

Follow

FYI: there are still more than 200,000 devices on the Internet vulnerable to Heartbleed

TOP COUNTRIES

United States	57,272
Germany	21,660
China	11,300
France	10,094
United Kingdom	9,125

TOP SERVICES

HTTPS	174,020
HTTPS (8443)	23,621
Webmin	8,148
8081	1,981
Symantec Data Center Security	1,307

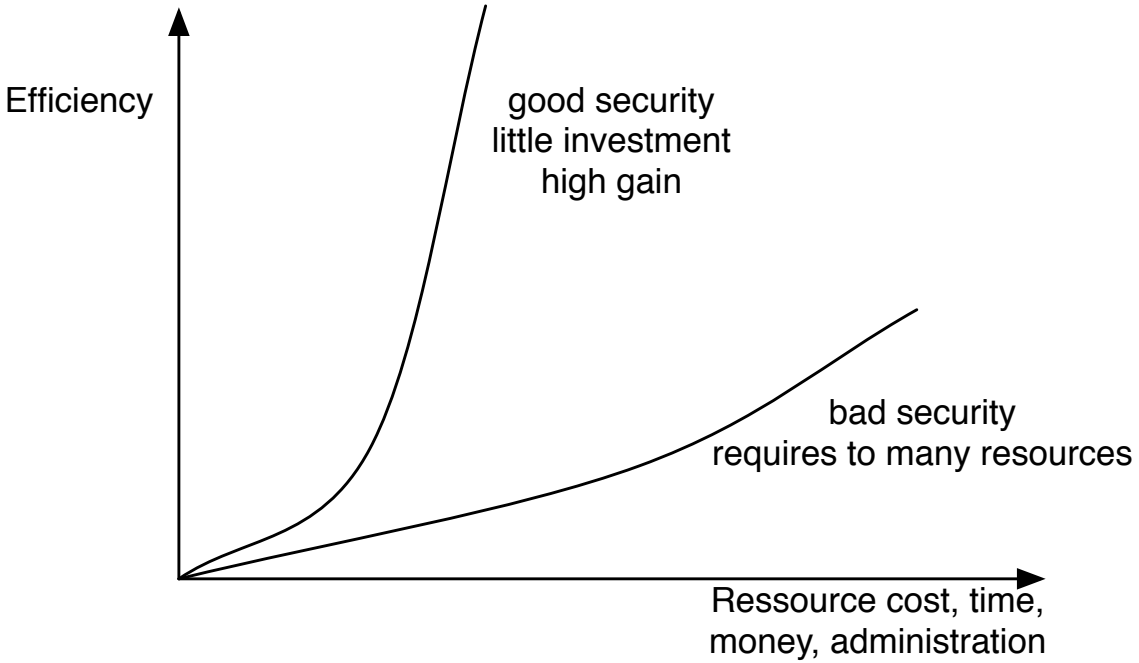
Source: Data from Shodan and Shodan Founder John Matherly

2016: Heartbleed vulnerable servers

Source: Data from Shodan and Shodan Founder John Matherly

`https://www.shodan.io/report/89bnfUyJ`

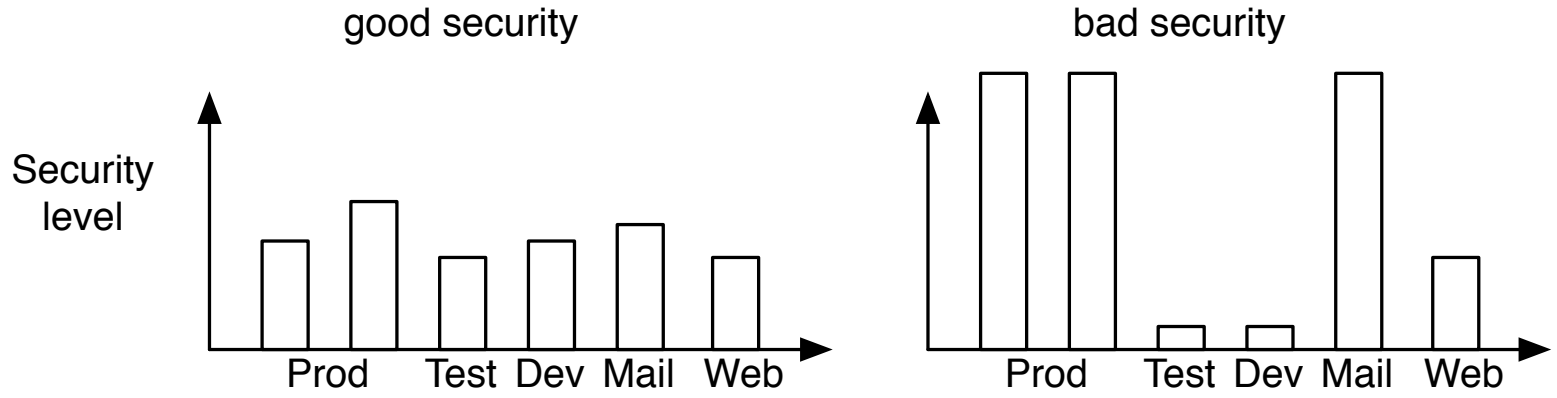
Good security



You always have limited resources for protection - use them as best as possible

Running Nmap requires almost nothing, verifies and checks a lot by itself!

Balanced security



Better to have the same level of security

If you have bad security in some part - guess where attackers will end up

Hackers are not required to take the hardest path into the network

Realize there is no such thing as 100% security

First advice

Use technology

Learn the technology - read the freaking manual

Think about the data you have, upload, facebook license?! WTF!

Think about the data you create - nude pictures taken, where will they show up?

- Turn off features you don't use
- Turn off network connections when not in use
- Update software and applications
- Turn on encryption: IMAP**S**, POP3**S**, HTTP**S** also for data at rest, full disk encryption, tablet encryption
- Lock devices automatically when not used for 10 minutes
- Dont trust fancy logins like fingerprint scanner or face recognition on cheap devices

Jumping through networks and systems

Hackers break into systems they can reach

and hackers break into systems they can reach from hacked systems ☺

In real life networks, a remote root exploit from the internet to the main database is rare

but jumping through others can possibly break the whole network

There are a lot of hackers which have presented how they hacked companies, example

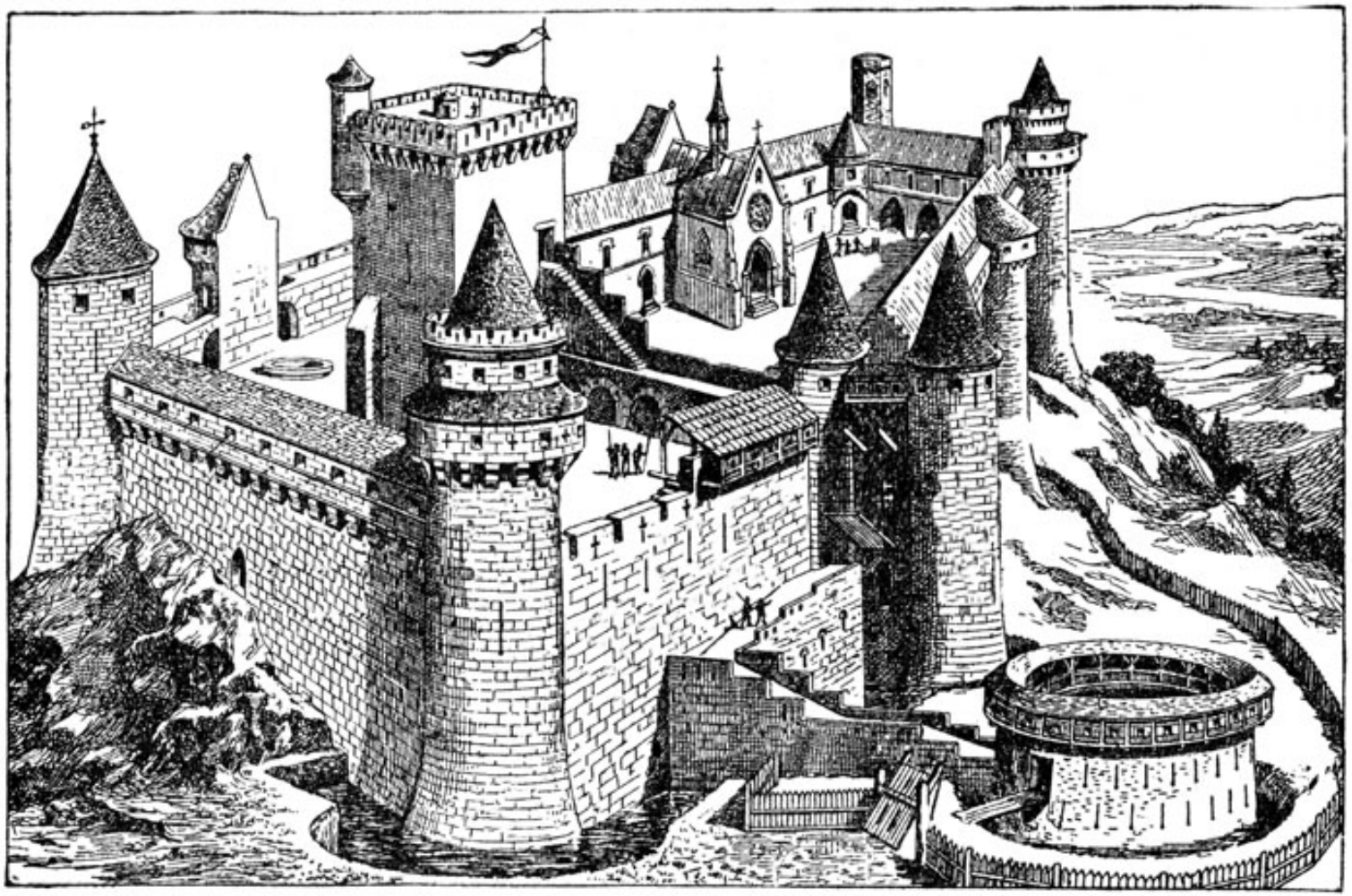
`https://arstechnica.com/security/2016/04/how-hacking-team-got-hacked-phineas-phisher/`

Example hacks we have done during testing

My own examples include pentest assignments where we:

- Two servers under test, Solaris
  had NFS world export, we could mount, found a backup of the other system, extracted /etc/passwd from backip, user logins with empty password, Solaris Telnet allows login, your password is empty - enter one
- Another test had SRX firewall as a core component. Due to misconfiguration we could access with SNMP, and thereby found the complete network structure revealed, how many network segments, subnets, netmask, ARP, interface counters - which lead to access files with password pictures, it was a bank
- Countless times we have found either a password file for a database, and used the same passwords for the operating system, or vice versa
- Countless times we have found either a password file in test systems, and people have used the same password in production, or vice versa

Defense in depth



Picture originally from: `http://karenswhimsy.com/public-domain-images`

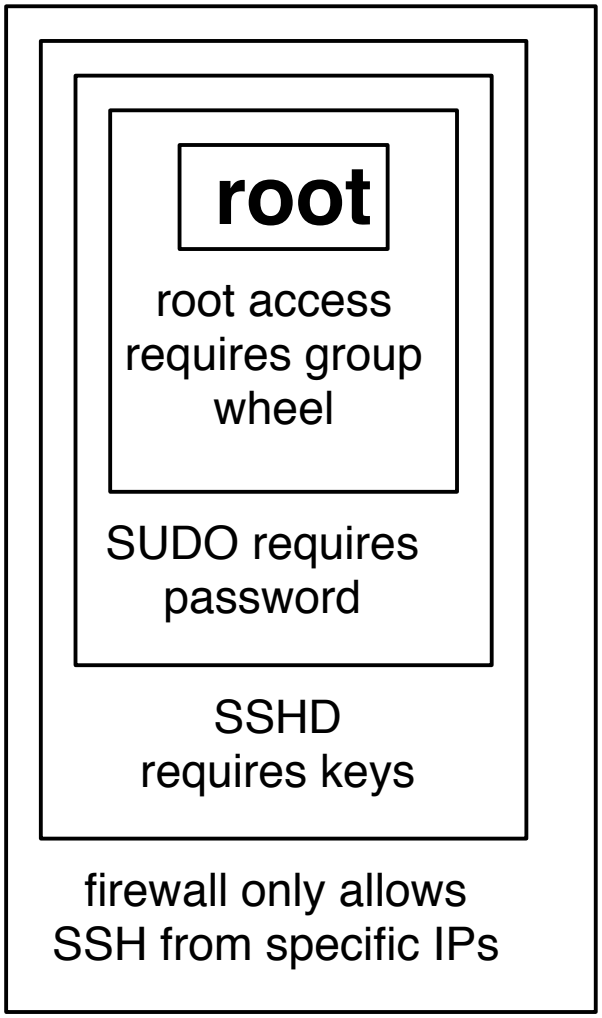
Chroot, Jails and Zones

Many types of *jails* in Unix-like operating systems

Ideas from Unix chroot, not a security feature originally

- Unix chroot - still used, but often combined with other things privsep
- FreeBSD Jails
- SELinux Mandatory Access Controls
- Solaris Containers og Zones
- VMware virtual servers, is that a jail?
- Docker, is that a jail or just a process?

Defense in depth - layered security



Multiple layers of security! Isolation!

First advice use the modern operating systems

Newer versions of Microsoft Windows, Mac OS X and Linux

- Buffer overflow protection
- Stack protection, non-executable stack
- Heap protection, non-executable heap
- *Randomization of parameters* stack gap m.v.

Note: these still have errors and bugs, but are better than older versions

OpenBSD has shown the way in many cases
`http://www.openbsd.org/papers/`

Always try to make life worse and more costly for attackers

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

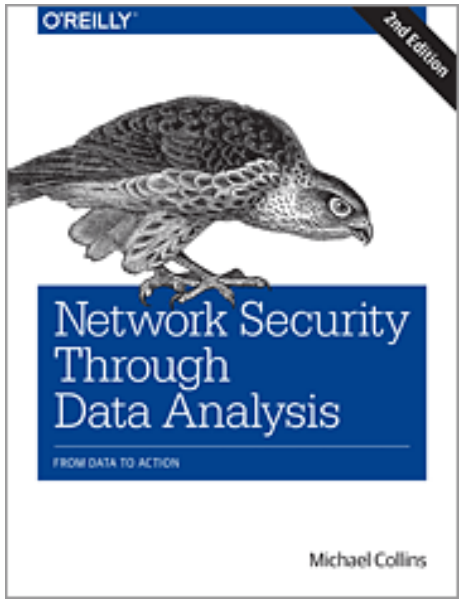
Example introductions:

- 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

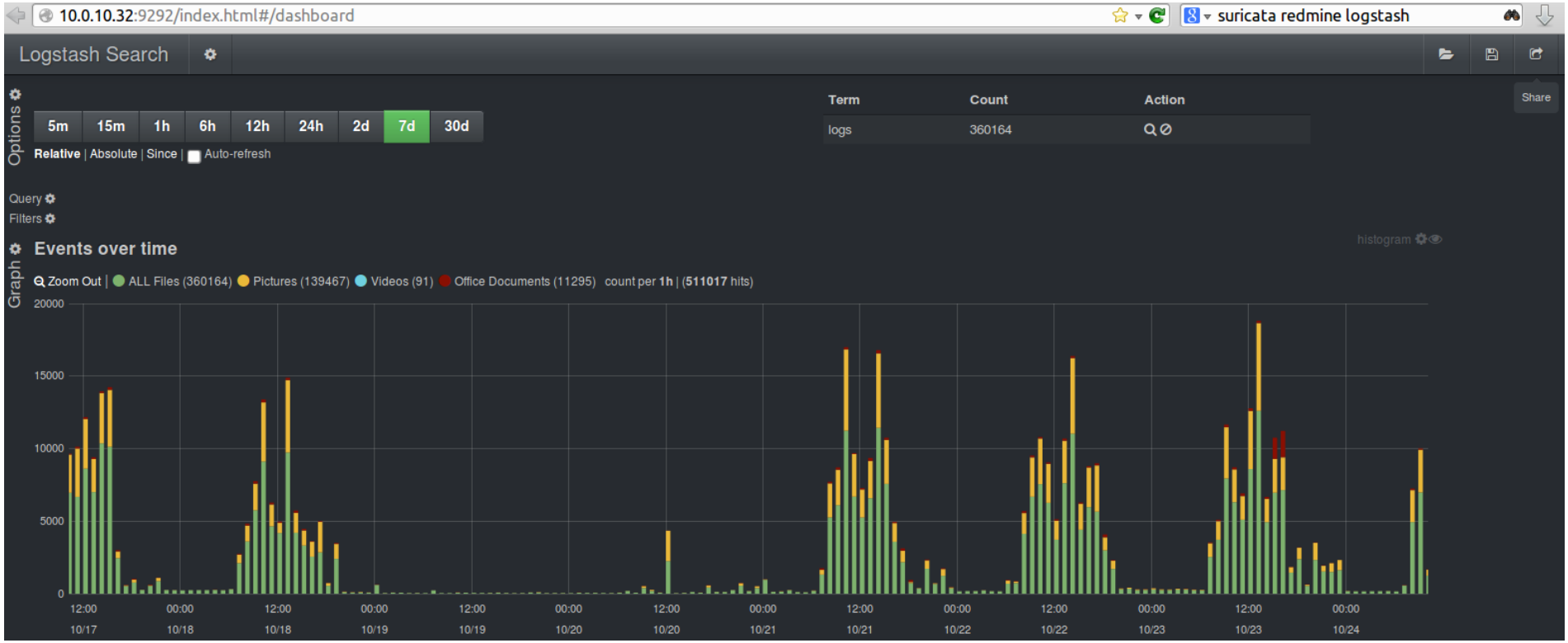


Low page count, but high value! Recommended.

Network Security through Data Analysis, 2nd Edition By Michael S Collins Publisher: O'Reilly Media 2015-05-01: Second release, 348 Pages

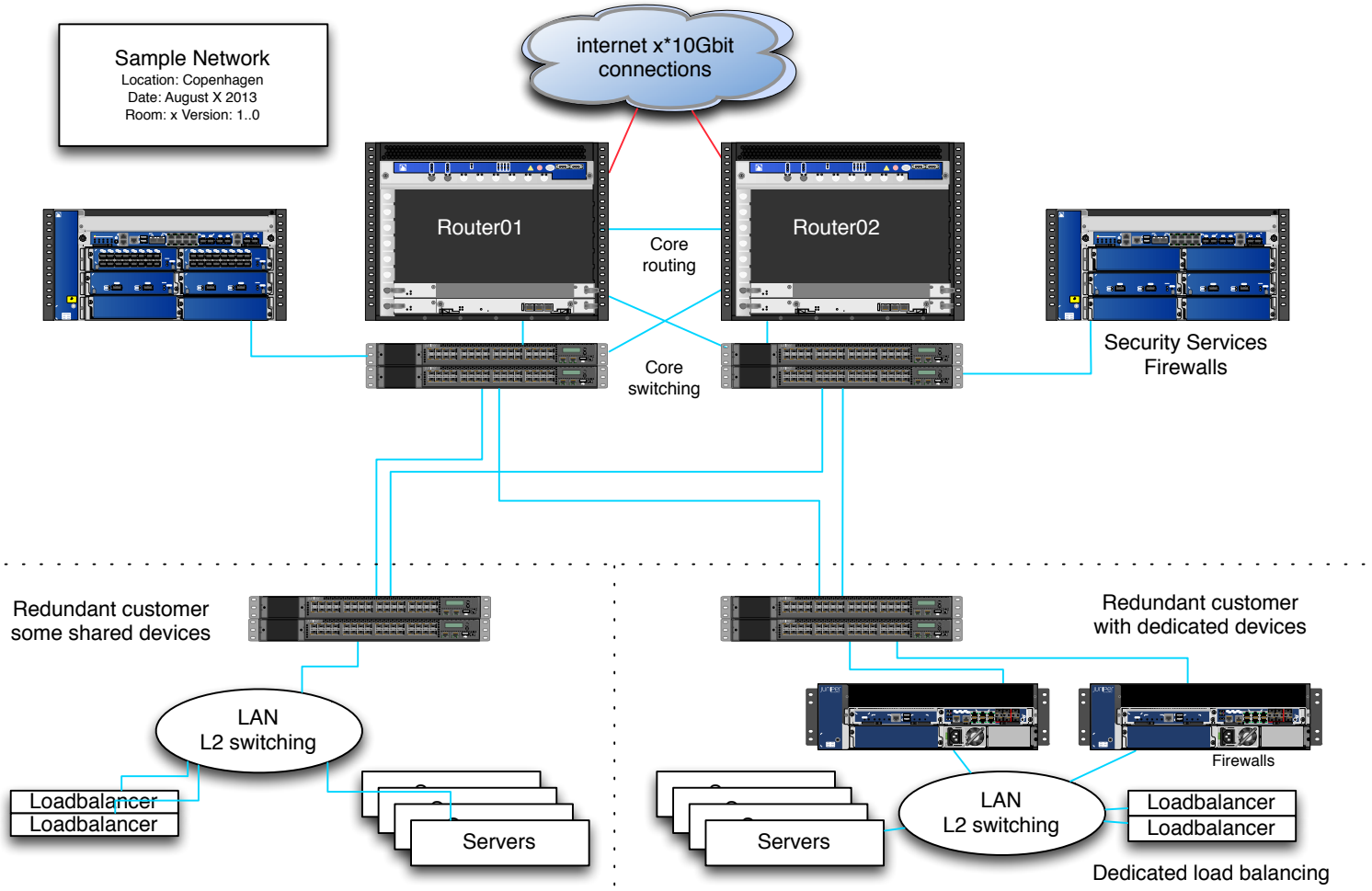
New Release Date: August 2017

Graphs and Dashboards!

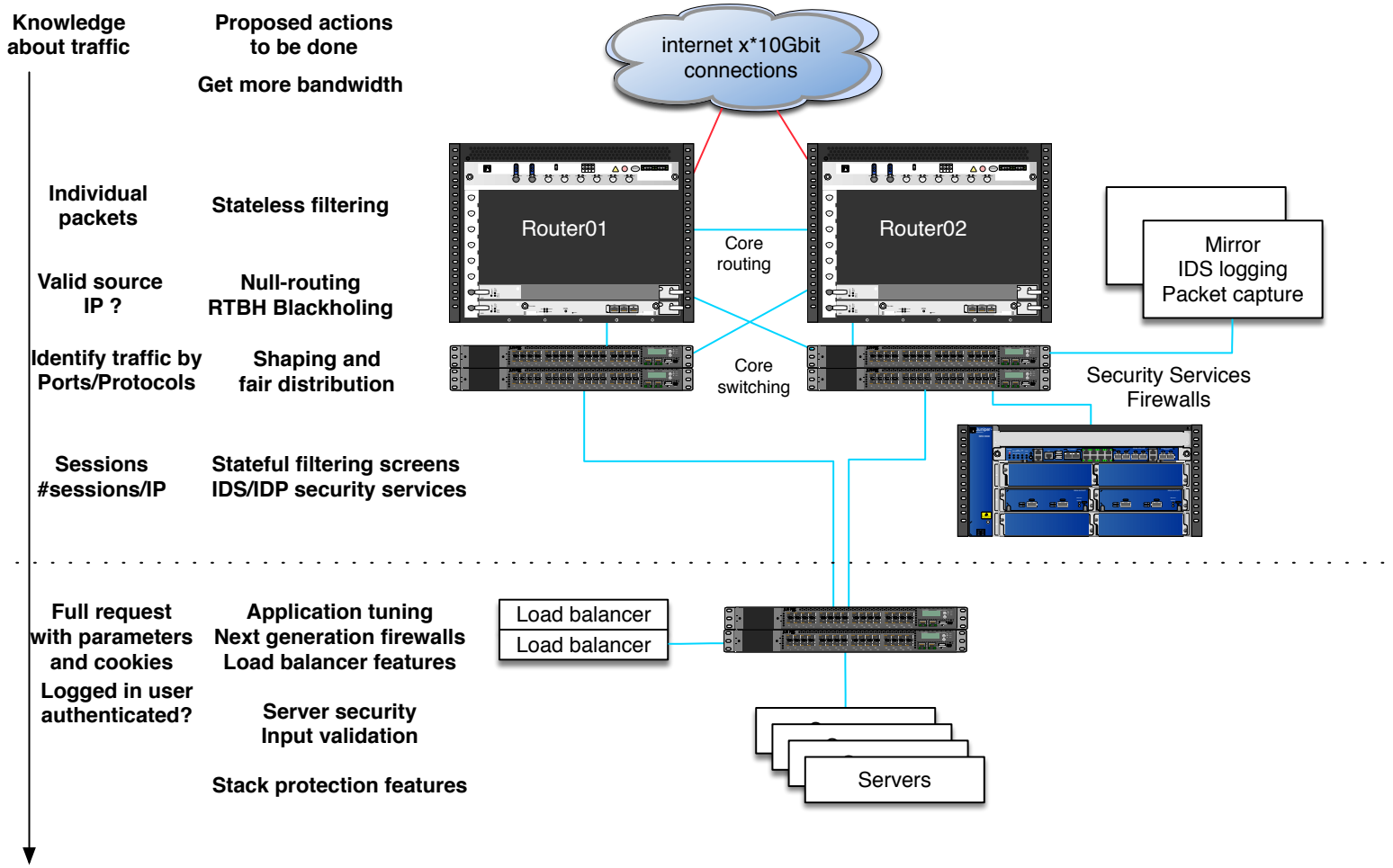


- Screenshot from Peter Manev, OISF
- Shown are Suricata IDS alerts processed by Logstash and Kibana

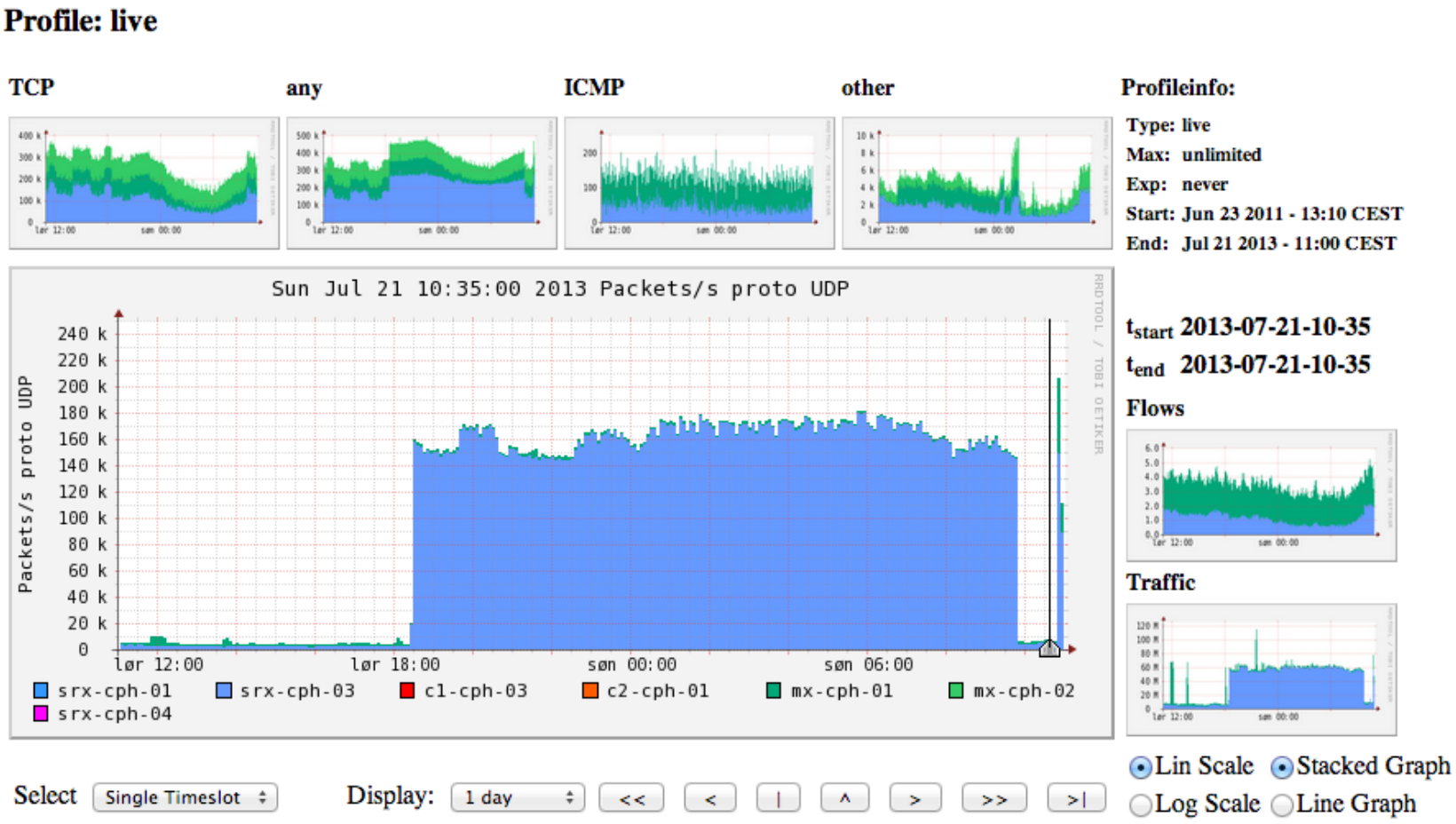
Networks today



Defense in depth - multiple layers of security



Network visibility: Netflow with NFSen



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

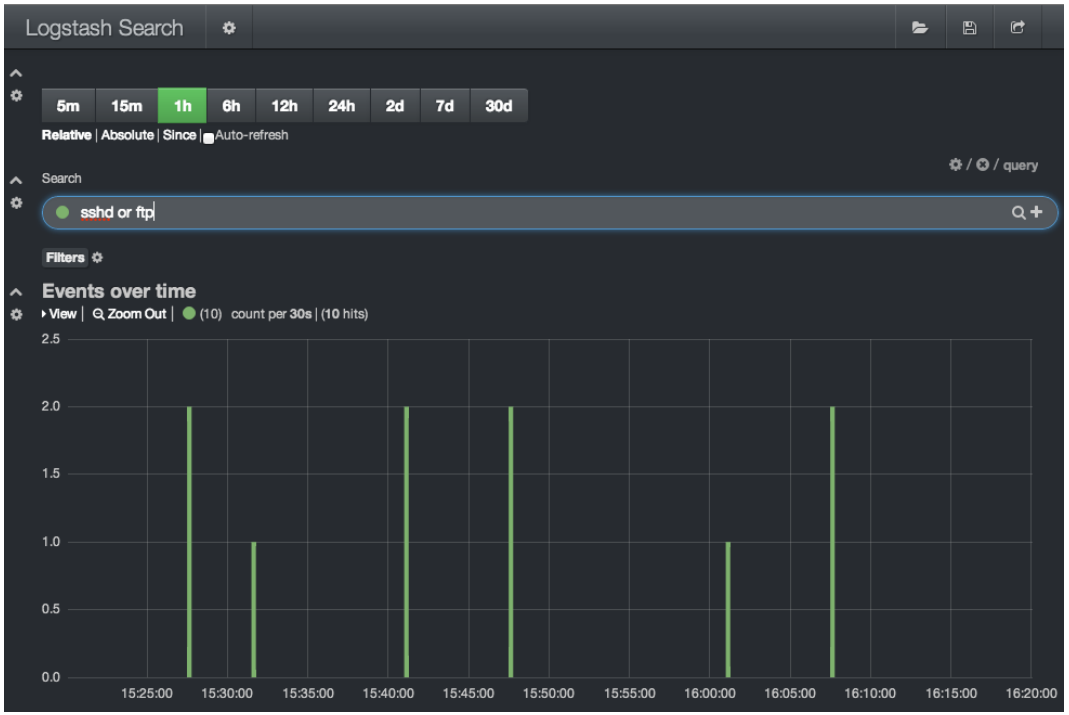
- Netflow data from routers
- Session data from firewalls
- Logging from applications: email, web, proxy systems

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

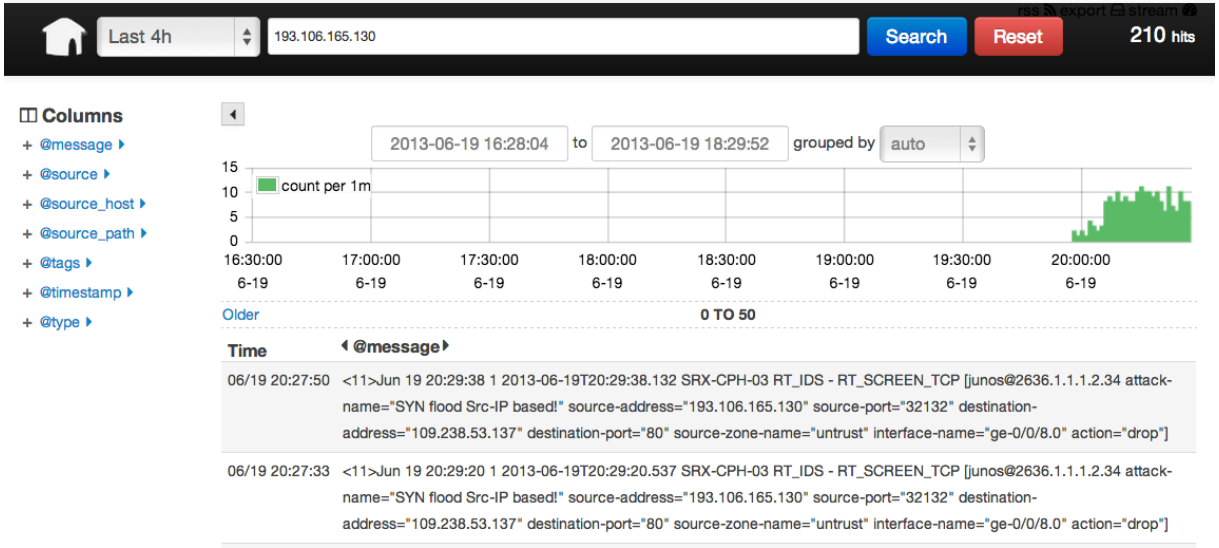


View data by digging into it easily - must be fast

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

Other popular examples include Graylog and Grafana

Network tools - examples



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, called ELK stack or Elastic stack

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

Collect and present data more easily - non-programmers

Example tool, let see what BRO IDS is

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`

Trust me, this IS better than having to write network parsers in C ☺

Bro demo (on a Mac)

```
kunoichi:~ root# brew install bro

kunoichi:~ root# broctl
Hint: Run the broctl "deploy" command to get started.

Welcome to BroControl 1.5

Type "help" for help.

[BroControl] > install
creating policy directories ...
installing site policies ...
generating standalone-layout.bro ...
generating local-networks.bro ...
generating broctl-config.bro ...
generating broctl-config.sh ...
```

Bro demo: Run bro

```
kunoichi:etc root# pwd
/usr/local/etc
kunoichi:etc root# grep eth0 node.cfg
interface=eth0
#interface=eth0
#interface=eth0
// My mac is not a Linux system, uses another interface naming scheme
kunoichi:etc root# vi node.cfg
kunoichi:etc root# grep en0 node.cfg
interface=en0
```

Bro demo: Run bro

```
// back to Broctl and start it
[BroControl] > start
starting bro
// and then
kunoichi:bro root# pwd
/usr/local/var/spool/bro
kunoichi:bro root# tail -f dns.log
```

More examples at:

https://www.bro.org/sphinx/script-reference/log-files.html

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 existing tools? or build your own specialised tools from scratch?

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 or 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.

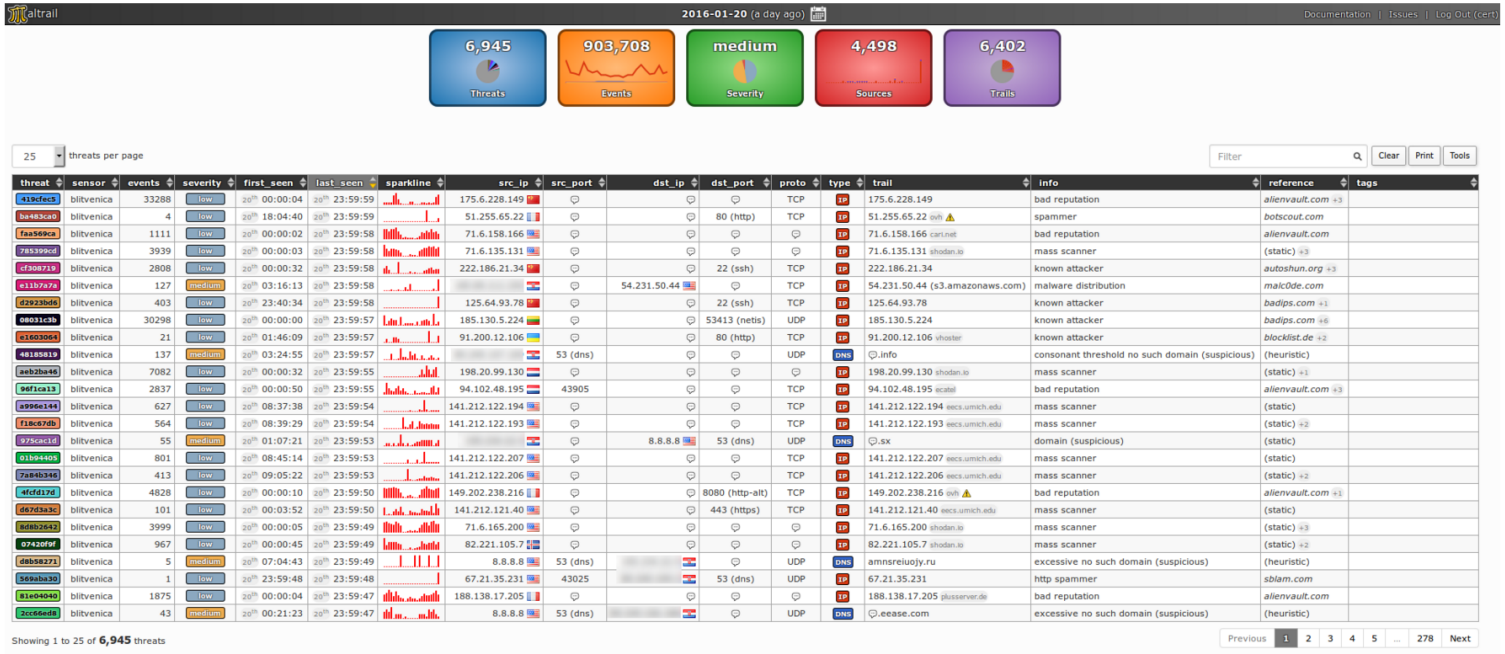
LANGuardian includes a file share activity monitoring module which provided a very detailed forensic analysis of the ransomware and the paths it had taken in order to encrypt the clients system and also the fileserver in which it was connected to, the initial infection came from the opening of an attachment 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/`

Case: Maltrail

Maltrail is a malicious traffic detection system, utilizing publicly available (black)lists containing malicious and/or generally suspicious trails, along with static trails compiled from various AV reports and custom user defined lists, where trail can be anything from domain name (e.g. `zvpprsensinaix.com` for Banjori malware), URL (e.g. `http://109.162.38.120/harsh02.exe` for known malicious executable), IP address (e.g. `185.130.5.231` for known attacker) or HTTP User-Agent header value (e.g. `sqlmap` for automatic SQL injection and database takeover tool). Also, it uses (optional) advanced heuristic mechanisms that can help in discovery of unknown threats (e.g. new malware).



`https://github.com/stamparm/maltrail`, demo hvis tid

Suricata with Dashboards

Picture from Twitter

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

`http://suricata-ids.org/`

Security Onion

- Security Onion is a Linux distro for IDS, NSM, and log management
- `http://securityonion.blogspot.dk`
- `http://blog.securityonion.net/p/securityonion.html`
- Not so great in production, focus on fewer tools, or buy BIG CPU ☺

Nice starting point for researching dashboards/network packets

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 }
}
```

- 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

Have you even configured SNMP traps?

Maybe you have a device sending SNMP traps right now ...

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

Remember defense in depth, dont run a priveleged Java VM as root ☺

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

Are passwords dead?

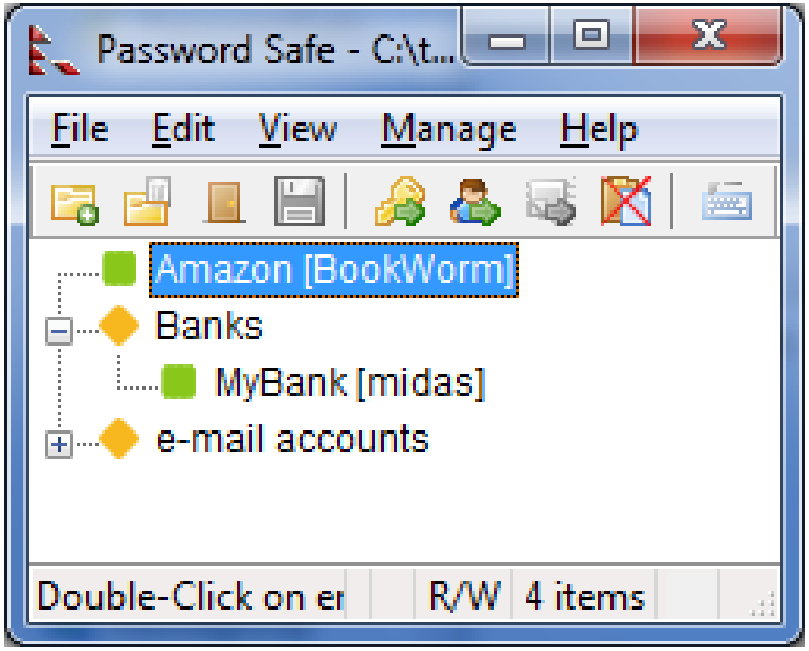
google: passwords are dead
About 6,580,000 results (0.22 seconds)

Can we stop using passwords?

Muffett on Passwords has a long list of password related information, from the author of crack `http://en.wikipedia.org/wiki/Crack_(password_software)`

`http://dropsafe.crypticide.com/muffett-passwords`

Storing passwords



PasswordSafe `http://passwordsafe.sourceforge.net/`

Apple Keychain provides an encrypted storage

Browsere, Firefox Master Password, Chrome passwords, ... who do YOU trust

Google looks to ditch passwords for good

”Google is currently running a pilot that uses a YubiKey cryptographic card developed by Yubico

The YubiKey NEO can be tapped on an NFC-enabled smartphone, which reads an encrypted one-time password emitted from the key fob.”

Source: `http://www.zdnet.com/google-looks-to-ditch-passwords-for-good-with-nfc-based-replacement`

Yubico Yubikey

› YubiKey Standard

Our flagship product, making strong two-factor authentication, easy and affordable for everyone.

› YubiKey NEO

Our premium YubiKey, combining USB, NFC, one-time password and PKI technology.

› YubiKey Nano

The world's smallest one-time password token, designed to stay inside the USB-slot.

› YubiKey VIP

A YubiKey Standard pre-configured with a Symantec VIP credential, enabling two-factor authentication against Symantec VIP enabled services, such as PayPal.

› LastPass YubiKey

LastPass Premium is the leading cross platform password manager supporting the YubiKey. We offer a number of discounted bundles of YubiKey + LastPass Premium Subscriptions.

› Password Safe YubiKey

Pasword Safe is an open source password manager initiated by Bruce Schneier. The YubiKey is used in Challenge-response mode to for 2 factor encryption of the database.

A Yubico OTP is unique sequence of characters generated every time the YubiKey button is touched. The Yubico OTP is comprised of a sequence of 32 Modhex characters representing information encrypted with a 128 bit AES-128 key

`http://www.yubico.com/products/yubikey-hardware/`

Duosecurity

Push Notification
Quickly view login or transaction details and tap "Approve" on your iOS or Android device.
Learn more at duosecurity.com/duo-push

Smartphone Passcodes
Easily generate login passcodes — no cell service required. Duo Mobile is available for free on all smartphone platforms.

Text Message
Login passcodes sent via text message. Works on all phones with SMS support.

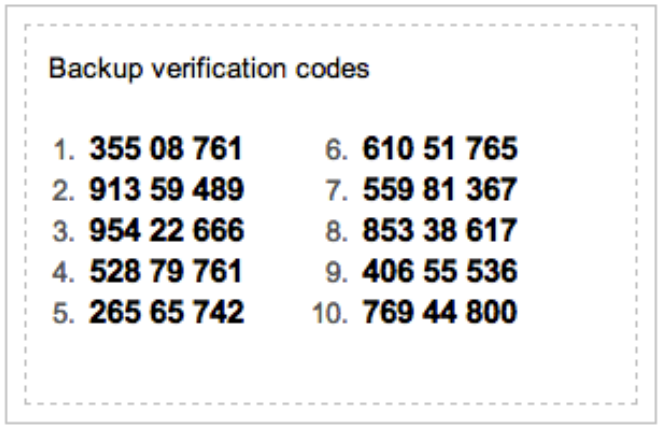
Phone Call
Simply answer a phone call and press a key to authenticate.

Video `https://www.duosecurity.com/duo-push`

`https://www.duosecurity.com/`

Low tech 2-step verification

Printing code on paper, low level pragmatic



Login from new devices today often requires two-factor - email sent to user

Google 2-factor auth. SMS with backup codes

Also read about S/KEY developed at Bellcore **in the late 1980s**
`http://en.wikipedia.org/wiki/S/KEY`

Conclusion passwords: integrate with authentication, not reinvent

Integrate or develop?

From previous slide:

Conclusion passwords: integrate with authentication, not reinvent

Dont:

- Reinvent the wheel - too many times, unless you can maintain it afterwards
- Never invent cryptography yourself
- No copy paste of functionality, harder to maintain in the future

Do:

- Integrate with existing solutions
- Use existing well-tested code: cryptography, authentication, hashing
- Centralize security in your code
- Fine to hide which authentication framework is being used, easy to replace later

Conclusion

If there is any conclusion ... it might be

There are existing - free tools

Open Source tools may integrate better with other open source tools

Syslog is de facto, but standards in other areas are coming along

Dont use block lists alone, but they may help in identifying problems

Questions?

Henrik Lund Kramshøj hlk@zencurity.dk

Need DDoS testing or pentest, ask me!

You are always welcome to send me questions later via email

Did you notice how a lot of the links in this presentation use HTTPS - encrypted