Welcome to

Core Infrastructure and BGP Intro

Building a production network

Henrik Lund Kramshøj hlk@zencurity.com @kramse 🐦

Slides are available as PDF, kramse@Github
core-infrastructure.tex in the repo security-courses

Don't Panic!

Spend some hours setting up a production network:

Use VLANs, IP subnetting, routing, BGP

Use Automated provisioning scripts for network devices

Use SNMP and setup LibreNMS server

Configure monitoring

Mirror data and start an IDS

We try to do a lot, feel free to focus on specific parts

Packet sniffing tools

Tcpdump for capturing packets

Wireshark for dissecting packets manually with GUI

Zeek Network Security Monitor

Snort, old timer Intrusion Detection Engine (IDS)

Suricata, modern robust capable of IDS and IPS (prevention)

ntopng High-speed web-based traffic analysis

Maltrail Malicious traffic detection system `https://github.com/stamparm/MalTrail`

Often a combination of tools and methods used in practice

Full packet capture big data tools also exist

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/` brings together 100s of tools

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

Use this to generate bad traffic

Hackerlab setup
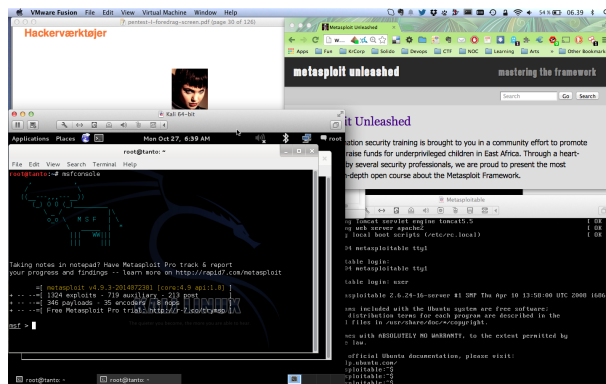
- **Hardware:** most modern laptops has CPU with virtualization
  May need to enale it in BIOS
- **Software:** use your favorite operating system, Windows, Mac, Linux
- **Virtualization software:** VMware, Virtual box, choose your poison
- **Hackersoftware:** Kali as a Virtual Machine `https://www.kali.org/`
- Install sniffing VM - put into *bridge mode*

What happens today?

Think like a blue team member find hacker traffic

Get basic tools running

Improve situation
- See where the data end up
- What kind of data and metadata can we extract
- How can we collect and make use of it
- Databases and web interfaces, examples shown
- Consider what your deployment could be

Today focus on the lower parts, but user interfaces are important too

Security devops

We need devops skillz in security

automate, security is also big data

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

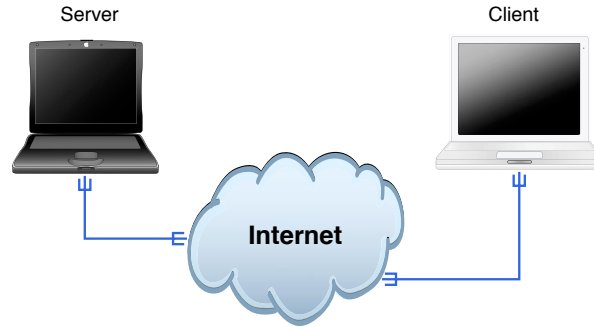tools, languages, databases, protocols, data formats

Use Github! So many libraries and programs that can help, maybe solve 90% of your problem, and you can glue the rest together

Example introductions:
- Seven languages/database/web frameworks in Seven Weeks
- Elasticsearch the definitive guide

We are all Devops now, even security people!

Internet today

Clients and servers

Roots in academia

Protocols more than 20 years old

HTTP is becoming encrypted, but a lot other traffic is not
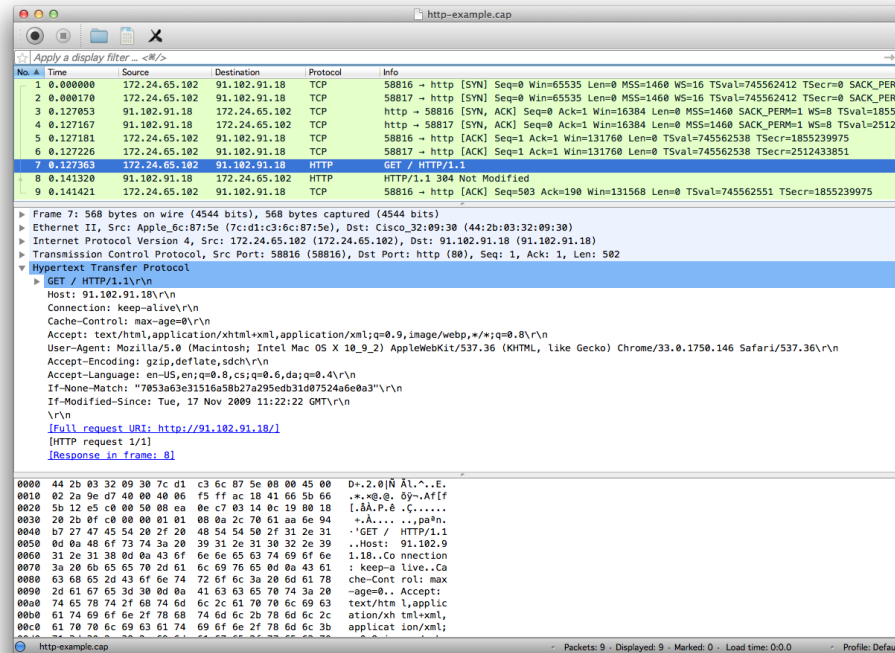
OSI og Internet modellerne

OSI Reference Model

OSI Reference Model
Application
Presentation
Session
Transport
Network
Link
Physical

Internet protocol suite

| Applications HTTP, SMTP, FTP,SNMP, | NFS |
| XDR |
| RPC |
TCP UDP	
IPv4 IPv6 ICMPv6 ICMP	
ARP RARP MAC	
Ethernet token-ring ATM ...	

Using Wireshark

https://www.wireshark.org
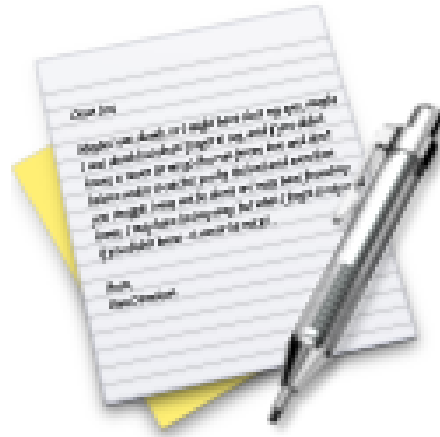
Exercise

Now lets do the exercise

Wireshark and Tcpdump 15 min

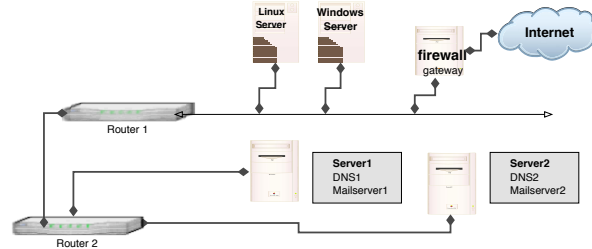which is number 4 in the exercise PDF.

Exercise

Now lets do the exercise

Capturing TCP Session packets 10 min

which is number **5** in the exercise PDF.

Network mapping

Using traceroute and similar programs it is often possible to make educated guess to network topology

Time to live (TTL) for packets are decreased when crossing a router

when it reaches zero the packet is timed out, and ICMP message sent back to source

Default Unix traceroute uses UDP, Windows tracert use ICMP

traceroute – UDP

```
 #  tcpdump -i en0 host 10.20.20.129 or host 10.0.0.11
tcpdump: listening on en0
23:23:30.426342 10.0.0.200.33849 > router.33435: udp 12  [ttl 1]
23:23:30.426742 safri > 10.0.0.200:  icmp: time exceeded in-transit
23:23:30.436069 10.0.0.200.33849 > router.33436: udp 12  [ttl 1]
23:23:30.436357 safri > 10.0.0.200:  icmp: time exceeded in-transit
23:23:30.437117 10.0.0.200.33849 > router.33437: udp 12  [ttl 1]
23:23:30.437383 safri > 10.0.0.200:  icmp: time exceeded in-transit
23:23:30.437574 10.0.0.200.33849 > router.33438: udp 12
23:23:30.438946 router > 10.0.0.200: icmp: router  udp port 33438 unreachable
23:23:30.451319 10.0.0.200.33849 > router.33439: udp 12
23:23:30.452569 router > 10.0.0.200: icmp: router  udp port 33439 unreachable
23:23:30.452813 10.0.0.200.33849 > router.33440: udp 12
23:23:30.454023 router > 10.0.0.200: icmp: router  udp port 33440 unreachable
23:23:31.379102 10.0.0.200.49214 > safri.domain:   6646+ PTR?
200.0.0.10.in-addr.arpa. (41)
23:23:31.380410 safri.domain > 10.0.0.200.49214:   6646 NXDomain* 0/1/0 (93)
14 packets received by filter
0 packets dropped by kernel
```

Low TTL, UDP, high ports above 33000 = Unix traceroute signature

Experiences gathered

Lots of information

Reveals a lot about the network, operating systems, services etc.

I use a template when getting data
- Respond to ICMP: □ echo, □ mask, □ time
- Respond to traceroute: □ ICMP, □ UDP
- Open ports TCP og UDP:
- Operating system:
- ... (banner information)

Beware when doing scans it is possible to make routers, firewalls and devices perform badly or even crash!

The Zeek Network Security Monitor

The Zeek Network Security Monitor

Why Choose Zeek? Zeek is a powerful network analysis framework that is much different from the typical IDS you may know.

Adaptable
Zeek's domain-specific scripting language enables site-specific monitoring policies.

Efficient
Zeek targets high-performance networks and is used operationally at a variety of large sites.

Flexible
Zeek is not restricted to any particular detection approach and does not rely on traditional signatures.

Forensics
Zeek comprehensively logs what it sees and provides a high-level archive of a network's activity.

In-depth Analysis
Zeek comes with analyzers for many protocols, enabling high-level semantic analysis at the application layer.

Highly Stateful
Zeek keeps extensive application-layer state about the network it monitors.

Open Interfaces
Zeek interfaces with other applications for real-time exchange of information.

Open Source
Zeek comes with a BSD license, allowing for free use with virtually no restrictions.

The Zeek Network Security Monitor is not a single tool, more of a powerful network analysis framework (former name Bro)

Source `https://www.zeek.org/`, redirects to `https://www.bro.org/zeek.html`

Zeek 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 https://www.bro.org/sphinx-git/script-refer

scripts.html

Exercise

Now lets do the exercise

Zeek on the web 10min

which is number 29 in the exercise PDF.

Exercise setup

We will use a combination of your virtual servers, my switch hardware and my virtual systems.

There will be sniffing done on traffic!
Don't abuse information gathered

We try to mimic what you would do in your own networks during the exercises.

Another way of running exercises might be:
https://github.com/jonschipp/ISLET

Recommended and used by Zeek and Suricata projects.

Get Started with Zeek

To run in "base" mode: `bro -r traffic.pcap`

To run in a "near broctl" mode: `bro -r traffic.pcap local`

To add extra scripts: `bro -r traffic.pcap myscript.bro`

Note: the project was renamed from Bro to Zeek in Oct 2018

Bro demo: Run

```
// install zeek first
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 ...
...
kunoichi:etc root# grep eth0 node.cfg
interface=eth0
```

Zeek demo: Run Zeek

```
// 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
```
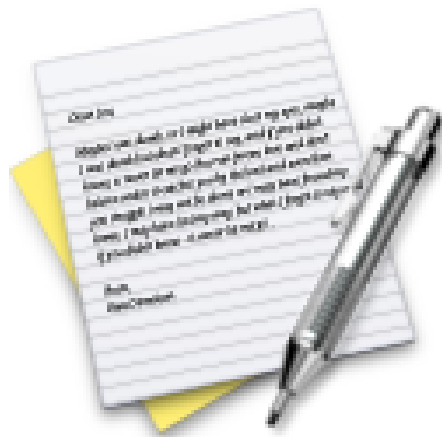
More examples at:

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

Exercise

Now lets do the exercise

Zeek DNS capturing domain names 10min

which is number 30 in the exercise PDF.

Exercise

Now lets do the exercise

Zeek TLS capturing certificates 10min

which is number 31 in the exercise PDF.

DNS is important

Another tool that provides a basic SQL-frontend to PCAP-files
https://www.dns-oarc.net/tools/packetq
https://github.com/DNS-OARC/PacketQ

Going back in time and finding systems that visited a specific domain can explain when and where an infection started.

Deciding on which tool to use, Zeek or PacketQ depends on the situation.

Suricata IDS/IPS/NSM

Suricata is a high performance Network IDS, IPS and Network Security Monitoring engine. Open Source and owned by a community run non-profit foundation, the Open Information Security Foundation (OISF). Suricata is developed by th e OISF and its supporting vendors.
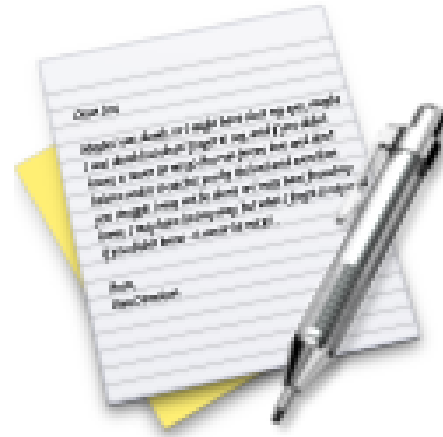
http://suricata-ids.org/ http://openinfosecfoundation.org

Exercise

Now lets do the exercise

Suricata Basic Operation 15min

which is number 32 in the exercise PDF.

Exercise

Now lets do the exercise

Basic Suricata rule configuration 15min

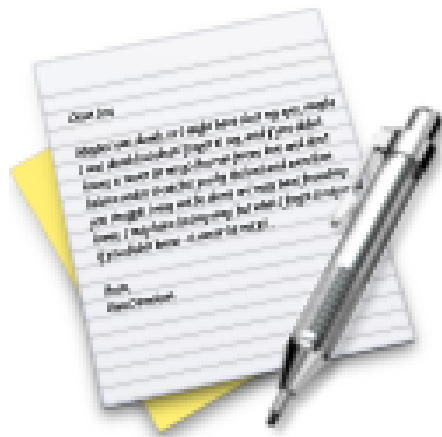which is number **33** in the exercise PDF.

Now lets do the exercise

Configure Mirror Port 10min

which is number 34 in the exercise PDF.

Exercise

Now lets do the exercise

Save Suricata JSON Output in Database 30min

which is number 35 in the exercise PDF.

Netflow

Netflow is getting more important, more data share the same links
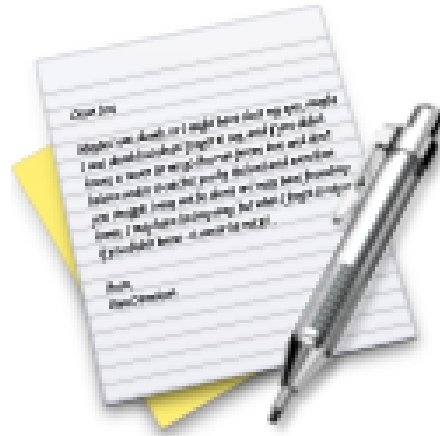
Accounting is important

Detecting DoS/DDoS and problems is essential

Netflow sampling is vital information - 123Mbit, but what kind of traffic

Currently also investigating sFlow - hopefully more fine grained

Exercise

Now lets do the exercise

Suricata Netflow 10min

which is number **36** in the exercise PDF.

Zeek JSON

As shown with Suricata, it is also possible to insert Bro output into Elasticsearch.

One example is shown in the blog posts: `https://logz.io/blog/bro-elk-part-1/`

Commercial Support

You can and should use updated rulesets for Suricata.

I Recommend the Emerging Threats ET Pro ruleset, which is about USD 900 per year per sensor. So two sites with Suricata running in both, 2x USD 900

Summary

We started from a basic Ubuntu/Debian server, and using Zeek and Suricata we now know more about our network.

We can collect logs about network traffic based on generic netflow, specific types DNS/TLS/protocols, and traffic matching advanced rulesets.

We know it is possible to create dashboards and visualizing the data.

What are the next steps?

Questions?

Henrik Lund Kramshøj hlk@zencurity.com @kramse 🐦

Need help with infrastructure security or pentesting, ask me!

You are always welcome to send me questions later via email