Welcome to

2. Hello world of Security Data Analysis

KEA Kompetence SIEM and Log Analysis

Henrik Kramselund he/him han/ham hlk@zencurity.com @kramse

Slides are available as PDF, kramse@Github

2-hello-world-security-analysis.tex in the repo security-courses

Goals for today

Todays goals:

- A Hello World example
- Try to go from raw data into Zeek, into Elasticsearch
- Investigating a few data types along the way

Photo by Thomas Galler on Unsplash

Plan for today

Subjects

- Data types: IP addresses, DNS, Netflow
- Note: not all details about these types are explained today!

Exercise theme: Low level, high value

- Data types: IP addresses and reputation lists
- Zeek as an example of a powerful multi-tool

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.

Reading Summary

DDS 3. Learning the "Hello World"of Security Data Analysis

DDS 4. Performing Exploratory Security Data Analysis

Do exercises if you feel like it, notice how small valuable programs can be

Reading Summary, continued

> This chapter takes the “Hello World” concept and expands it to a walk-through of a self-contained, introductory security data analysis use case that you will be able to follow along with, execute, and take concepts from as you start to perform your own analyses. There are parallel examples in Python and R to provide a somewhat agnostic view of the similarities, strengths, and differences between both languages in a real-life data analysis context. If you’re not familiar with one or both of those languages, you should read Chapter 2 and at least skim some of the external resources referenced there.

Source: DDS 3. Learning the ”Hello World”of Security Data Analysis

- Get data from HTTP – can be scheduled
- Process
- Present it - graphs and later dashboards
- We dont need to run all examples to see the benefit
- Lets try some of the scripts from this chapter

Reading Summary, continued

This chapter focuses on exploring IP addresses by starting with further analyses on the AlienVault IP Reputation database first seen in Chapter 3.

IP addresses—along with domain names and routing concepts—are the building blocks of the Internet. They are defined in RFC 791, the "Internet Protocol / DARPA Internet Program / Protocol Specification" (http://tools.ietf.org/html/rfc791), which has an elegant and succinct way of describing them:

> A name *indicates what we seek*. An address *indicates where it is*. A route *indicates how to get there*.

Source DDS 4. Performing Exploratory Security Data Analysis

1. Downloading (if necessary) new data
2. Parsing/munging and converting the new data into a data frame
3. Validating the contents and structure of the new data
4. Extracting or computing relevant information from the new data source
5. Creating one or more new columns in the existing data frame
6. Running new analyses

What happens today?

Think like a blue team member find hacker traffic

Get basic tools running

Improve situation

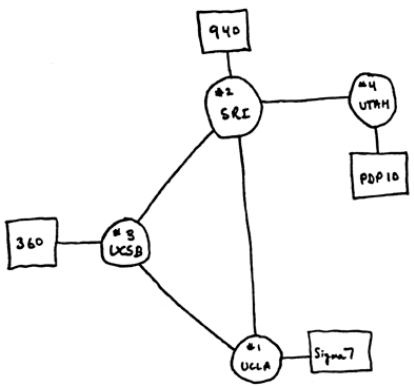
- See where the data end up
- What kind of data and metadata can we extract
- How can we collect and make use of it
- Databases and web interfaces, examples shown
- Consider what your deployment could be

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

IP: Internet historically

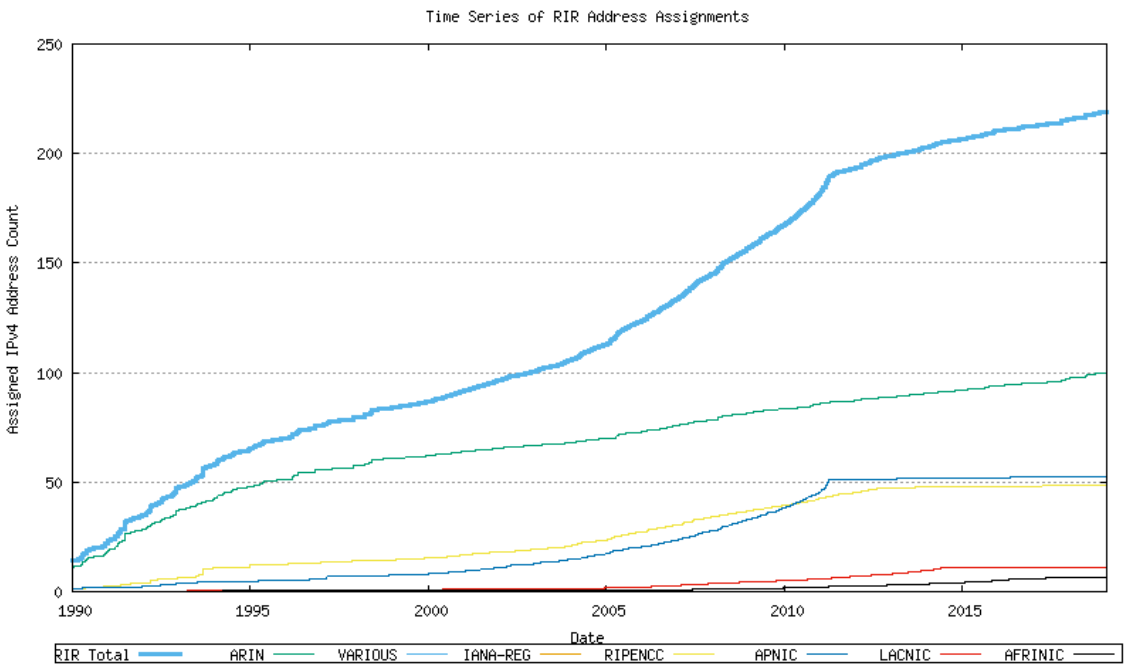
1961 L. Kleinrock, MIT packet-switching teori
1962 J. C. R. Licklider, MIT - notes
1964 Paul Baran: On Distributed Communications
1969 ARPANET start 4 nodes
1971 14 nodes
1973 Work on Internet Protocols IP started
1973 Email is about 75% af ARPANET traffik
1974 TCP/IP: Cerf/Kahn: A protocol for Packet Network Interconnection
1983 EUUG $\rightarrow$ DKUUG/DIKU connection
1988 ca. 60.000 systemer på Internet The Morris Worm rammer ca. 10%
2002 Ialt ca. 130 millioner på Internet

Internet historically set - anno 1969



- Node 1: University of California Los Angeles
- Node 2: Stanford Research Institute
- Node 3: University of California Santa Barbara
- Node 4: University of Utah

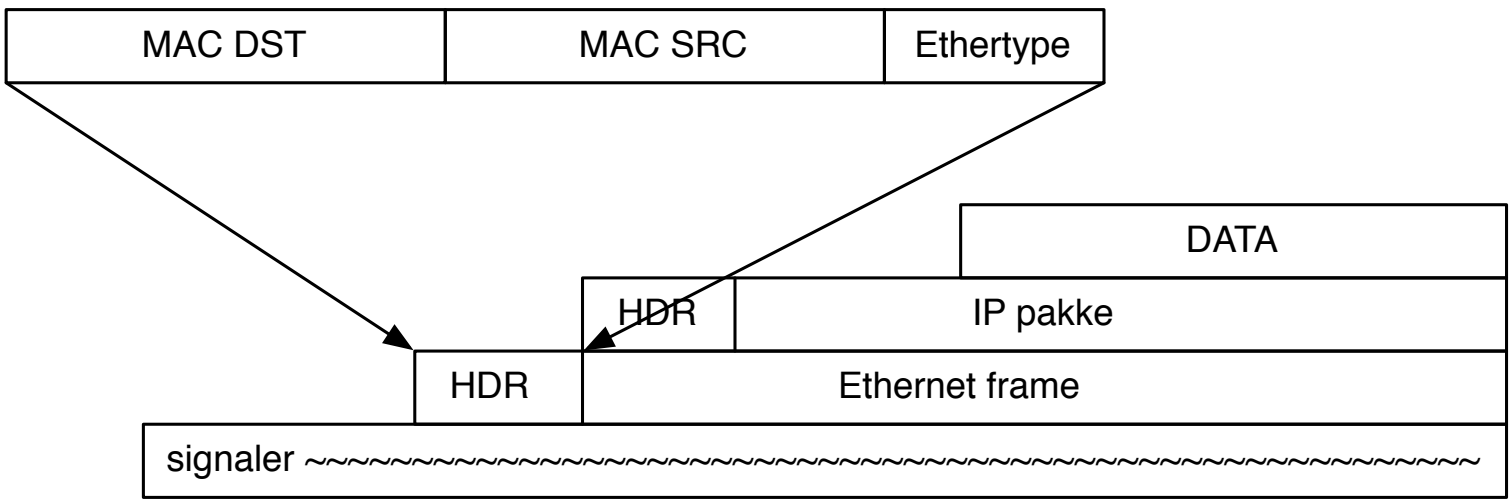
What are Internet hosts



Cumulative RIR address assignments, per RIR

Source: IPv4 Address Report - 28-Jan-2019 http://www.potaroo.net/tools/ipv4/

Packets across the wire or wireless



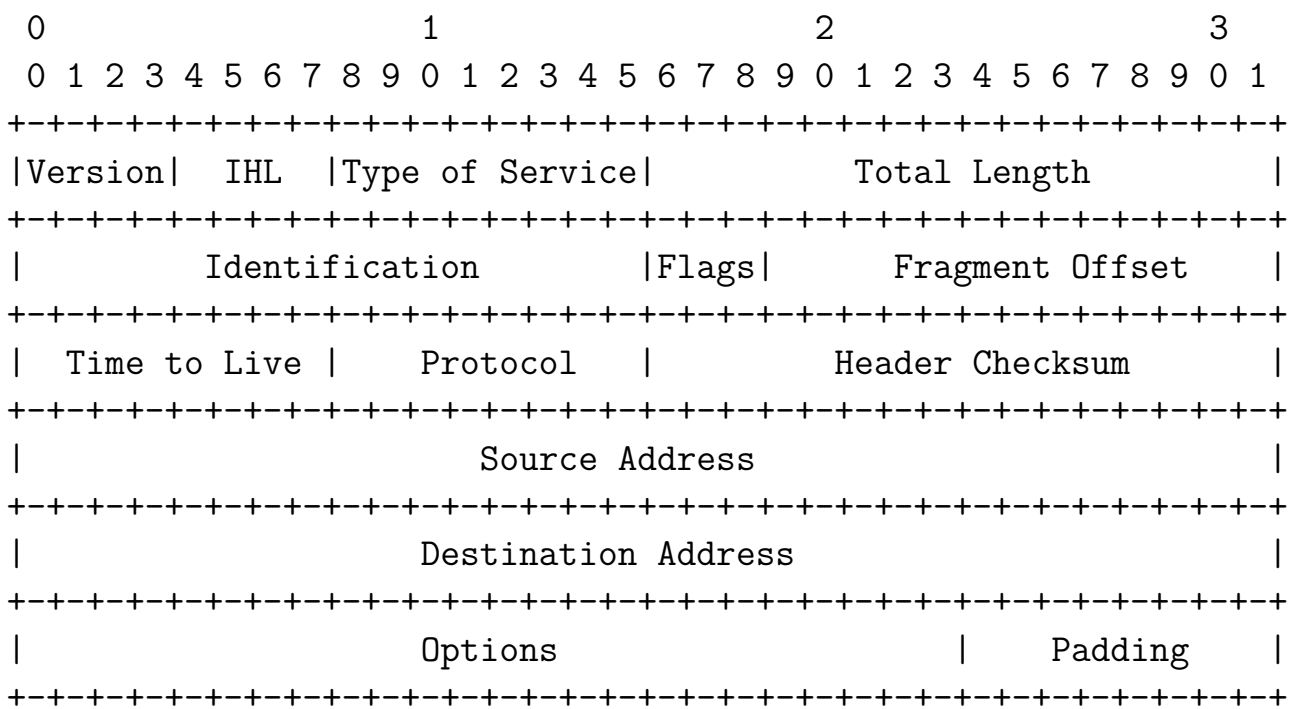
Looking at data as a stream the packets are a pattern laid on top

Network technology defines the start and end of a frame, example Ethernet

From a lower level we receive a packet, example 1500-bytes from Ethernet driver

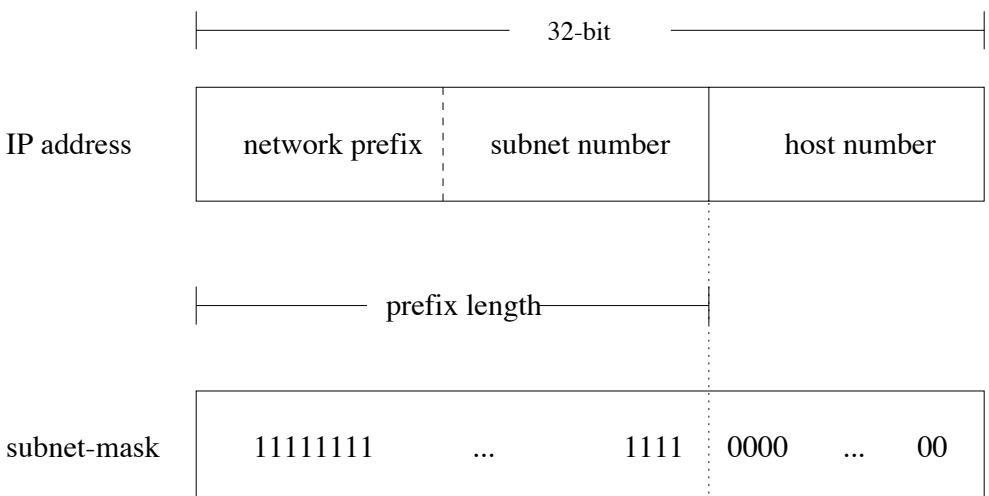
Operating system masks a lot of complexity

IPv4 packets - header - RFC-791



Example Internet Datagram Header

Common Address Space



Internet is defined by the address space, one

Based on 32-bit addresses, example 10.0.0.1

IPv4 address

```
hlk@bigfoot:hlk$ ipconvert.pl 127.0.0.1
Adressen er: 127.0.0.1
Adressen er: 2130706433
hlk@bigfoot:hlk$ ping 2130706433
PING 2130706433 (127.0.0.1): 56 data bytes
64 bytes from 127.0.0.1: icmp_seq=0 ttl=64 time=0.135 ms
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.144 ms
```

IP-adresser typically written as decimal numbers with dots

dot notation: 10.1.2.3

IP-adresser as bits

```
IP-adresse: 127.0.0.1
Heltal: 2130706433
Binary: 11111110000000000000000000000001
```

IP-address converted to bits

Computers use bits

Internet ABC

Previously we used classes: A, B, C, D og E

This proved to be a bit inflexible:

- A-klasse has 16 million hosts
- B-klasse about 65.000 hosts
- C-klasse only 250 hosts

Most people asked for B-klasser - starting to run out!

D-klasse used for multicast

E-klasse reserved

See http://en.wikipedia.org/wiki/Classful_network

Stop saying C, say /24

CIDR Classless Inter-Domain Routing

Classful routing		Classless routing CIDR	
4 class C networks	Inherent subnet mask	Supernet	Subnet mask
192.0.8.0	255.255.255.0	192.0.8.0	255.255.252.0
192.0.9.0	255.255.255.0		252d=11111100b
192.0.10.0	255.255.255.0		
192.0.11.0	255.255.255.0	Base network/prefix	
		192.10.8.0/22	

Subnet mask originally inferred by the class

Started to allocate multiple C-class networks - save remaining B-class
Resulted in routing table explosion

A subnet mask today is a row of 1-bit

10.0.0.0/24 means the network 10.0.0.0 with subnet mask 255.255.255.0

Supernet, supernetting

RFC-1918 Private Networks

Der findes et antal adresserum som alle må benytte frit:

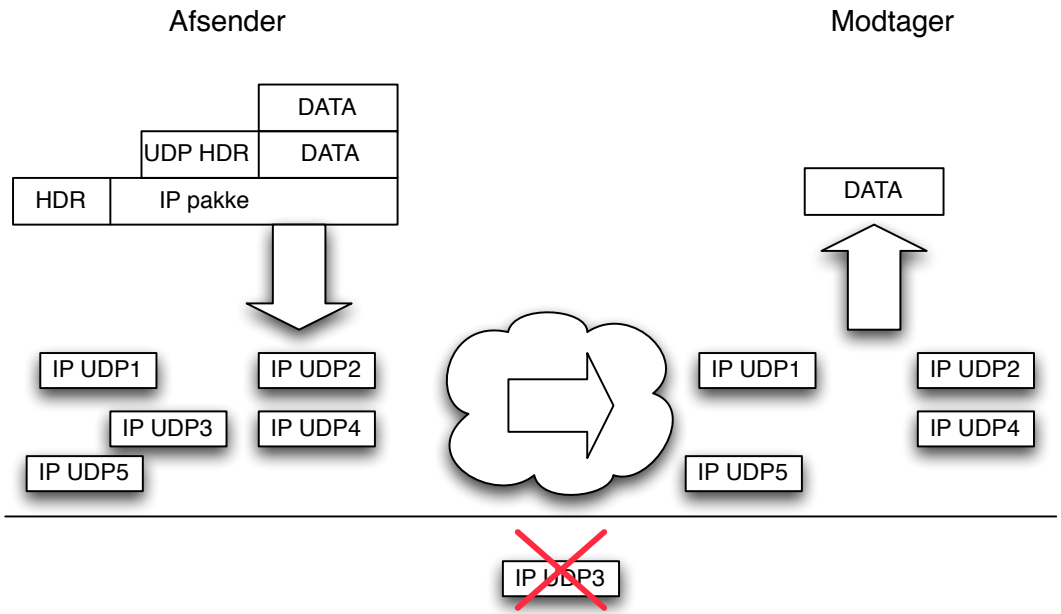
- 10.0.0.0 - 10.255.255.255 (10/8 prefix)
- 172.16.0.0 - 172.31.255.255 (172.16/12 prefix)
- 192.168.0.0 - 192.168.255.255 (192.168/16 prefix)

Address Allocation for Private Internets RFC-1918 adresserne!

NB: man må ikke sende pakker ud på internet med disse som afsender, giver ikke mening

```
The blocks 192.0.2.0/24 (TEST-NET-1), 198.51.100.0/24 (TEST-NET-2),
and 203.0.113.0/24 (TEST-NET-3) are provided for use in
documentation.

169.254.0.0/16 has been ear-marked as the IP range to use for end node
auto-configuration when a DHCP server may not be found
```

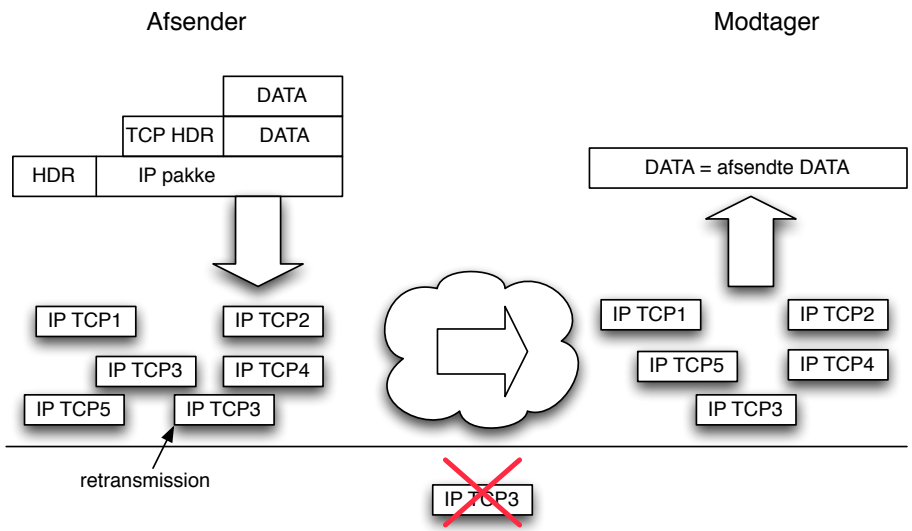
UDP User Datagram Protocol



Connection-less RFC-768, *connection-less*

Used for Domain Name Service (DNS)

TCP Transmission Control Protocol

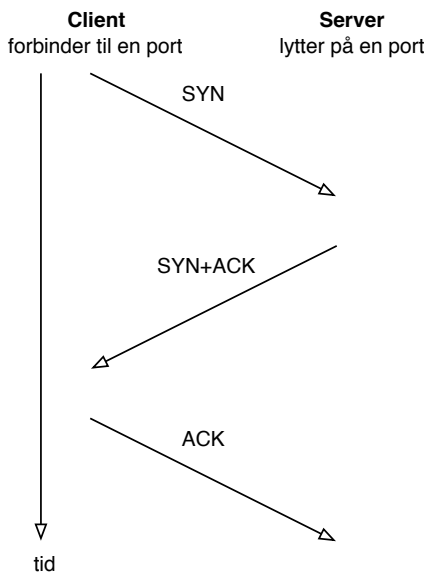


Connection oriented RFC-791 September 1981, *connection-oriented*

Either data delivered in correct order, no data missing, checksum or an error is reported

Used for HTTP and others

TCP three way handshake



- Session setup is used in some protocols
- Other protocols like HTTP/2 can perform request in the first packet

Well-known port numbers

IANA maintains a list of magical numbers in TCP/IP

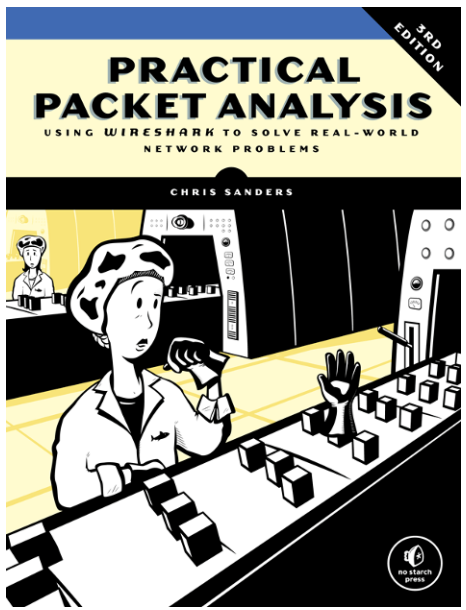
Lists of protocl numbers, port numers etc.

A few notable examples:

- Port 25/tcp Simple Mail Transfer Protocol (SMTP)
- Port 53/udp and 53/tcp Domain Name System (DNS)
- Port 80/tcp Hyper Text Transfer Protocol (HTTP)
- Port 443/tcp HTTP over TLS/SSL (HTTPS)

Source: http://www.iana.org

Book: Practical Packet Analysis (PPA)



Practical Packet Analysis, Using Wireshark to Solve Real-World Network Problems by Chris Sanders, 3rd Edition April 2017, 368 pp. ISBN-13: 978-1-59327-802-1

https://nostarch.com/packetanalysis3

Investigate examples from the internet

Simple Example

Let's begin with an example. Assume we want to create a simple `ping` measurement from `50` probes anywhere in the world to `ripe.net`.

Here's how to do this in cURL:

Terminal

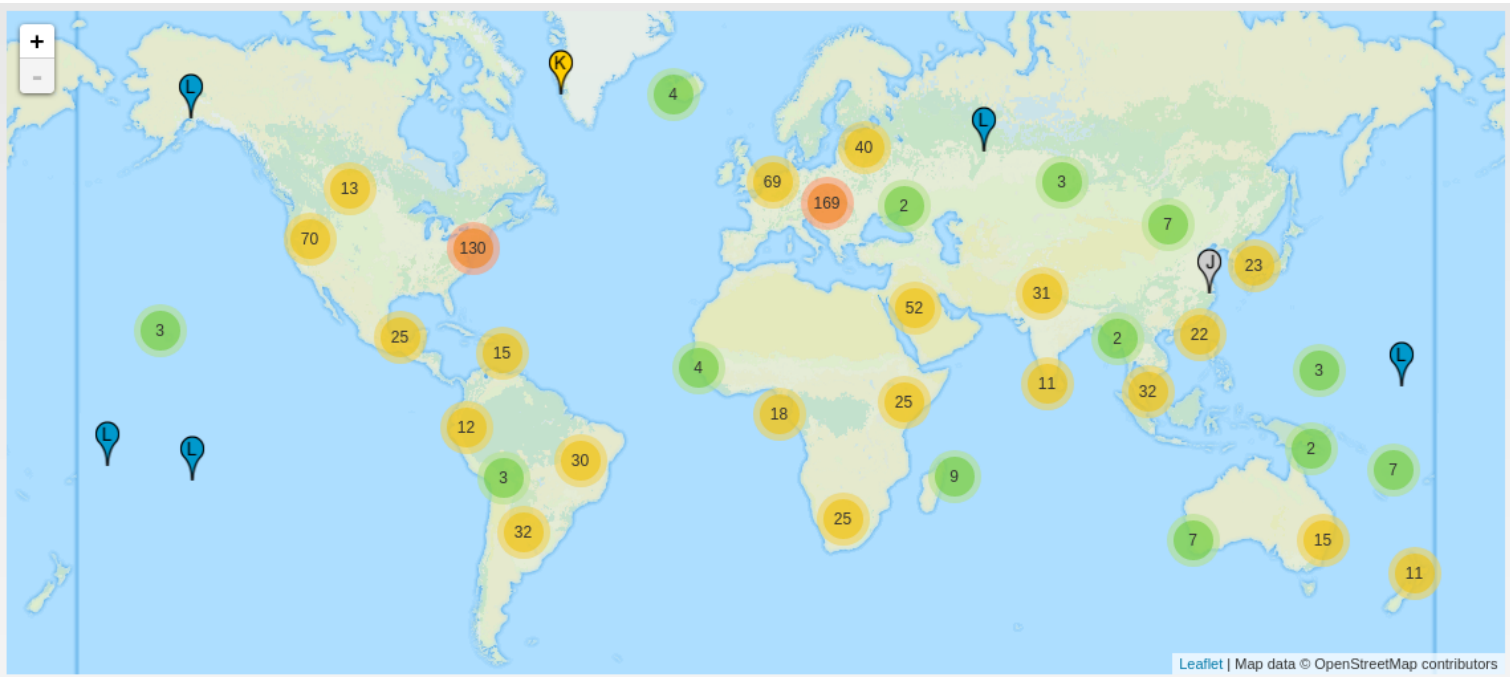
```
~$ curl -H "Content-Type: application/json" -H "Accept: application/json" -X POST -d '{
    "definitions": [ { "target": "ripe.net", "description": "My First Measurement",
    "type": "ping", "af": 4 } ],
    "probes": [ { "requested": 50, "type": "area", "value": "WW" } ] }' https://atlas.ripe.net/api/v1/meas
urement/?key=YOUR_API_KEY
```

- Which web services do you use? Can we find examples of XML and JSON web services
- Look into the services provided by DBC, what languages, formats, services - look at their Github account DBCDK
- We will use internet data from RIPE NCC Atlas service, if nothing else:

  https://ripe-atlas-tools.readthedocs.io/en/latest/

  https://atlas.ripe.net/docs/api/v2/manual/pdf/ripe_atlas_api_V2_manual.pdf
- If using the command line to set the API key: `ripe-atlas configure --set authorisation.create=e7fd3981-9592-481e-8bcd-f50d17e65de8`

DNS root servers



As of 2019-01-29, the root server system consists of 933 instances operated by the 12 independent root server operators.

http://root-servers.org/

DNS is important

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
```

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.

Other tools and references

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

Other tools and references:

- Packetbeat https://www.elastic.co/products/beats/packetbeat
- http://securityblog.switch.ch/2013/01/22/using-packetq/
- http://jpmens.net/2013/05/27/server-agnostic-logging-of-dns-queries-responses/

Indicators of Compromise and Signatures

An IOC is any piece of information that can be used to objectively describe a network intrusion, expressed in a platform-independent manner. This could include a simple indicator such as the IP address of a command and control (C2) ser ver or a complex set of behaviors that indicate that a mail server is being used as a malicious SMTP relay.

When an IOC is taken and used in a platform-specific language or format, such as a Snort Rule or a Zeek-formatted file, it becomes part of a signature. A sign ature can contain one or more IOCs.

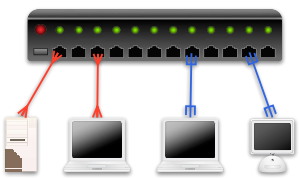
Source: Applied Network Security Monitoring Collection, Detection, and Analysis, 2014 Chris Sanders

Reading Summary, False Positives

- True Positive (TP). An alert that has correctly identified a specific activity. If a signature was designed to detect a certain type of malware, and an alert is generated when that malware is launched on a system, this would be a true positive, which is what we strive for with every deployed signature.Indicators of Compromise and Signatures
- False Positive (FP). An alert has incorrectly identified a specific activity. If a signature was designed to detect a specific type of malware, and an alert is generated for an instance in which that malware was not present, this would be a false positive.
- True Negative (TN). An alert has correctly not been generated when a specific activity has not occurred. If a signature was designed to detect a certain type of malware, and no alert is generated without that malware being launched, then this is a true negative, which is also desirable. This is difficult, if not impossible, to quantify in terms of NSM detection.
- False Negative (FN). An alert has incorrectly not been generated when a specific activity has occurred.

Source: Applied Network Security Monitoring Collection, Detection, and Analysis, 2014 Chris Sanders

Reputation-Based Detection



- The most basic form of intrusion detection is reputation-based detection
- Similar concept to block lists for SMTP spam relays
- I often recommend https://github.com/stamparm/maltrail as a source of lists
- Other sources are lists like RIPE NCC delegated, which IP prefixes are handed out in different countries
  https://ftp.ripe.net/pub/stats/ripencc/delegated-ripencc-extended-latest
  `ripencc|DK|ipv4|185.129.60.0|1024|20151130|allocated|`
- Tool often mentioned are Argus and SiLK https://tools.netsa.cert.org/silk/
  If we end up having time today, or another day, we should look into this tool chain also!
- Old and mature tools have been proven to work

IP reputation

Zeek documentation Intel framework
https://docs.zeek.org/en/stable/frameworks/intel.html

Suricata reputation support
https://suricata.readthedocs.io/en/latest/reputation/index.html

Exercise

Now lets do the exercise

Data types: IP addresses – 15min

which is number **15** in the exercise PDF.

Exercise

Now lets do the exercise

Data types: IP reputation – 15min

which is number **16** in the exercise PDF.

Low level, high value

- I will introduce some of my favourite tools, and they are low-level

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
- NFSen is an old but free application http://nfsen.sourceforge.net/
- Currently also investigating sFlow - hopefully more fine grained
- sFlow, short for "sampled flow", is an industry standard for packet export at Layer 2 of the OSI model, https://en.wikipedia.org/wiki/SFlow

Netflow is often from routers, we dont have any here

Also look into Elastiflow! https://github.com/robcowart/elastiflow

Collect Network Evidence from the network

Network Flows

Cisco standard NetFlow version 5 defines a flow as a unidirectional sequence of packets that all share the following 7 values:

- Ingress interface (SNMP ifIndex)
- IP protocol, Source IP address and Destination IP address
- Source port for UDP or TCP, 0 for other protocols
- Destination port for UDP or TCP, type and code for ICMP, or 0 for other protocols
- IP Type of Service

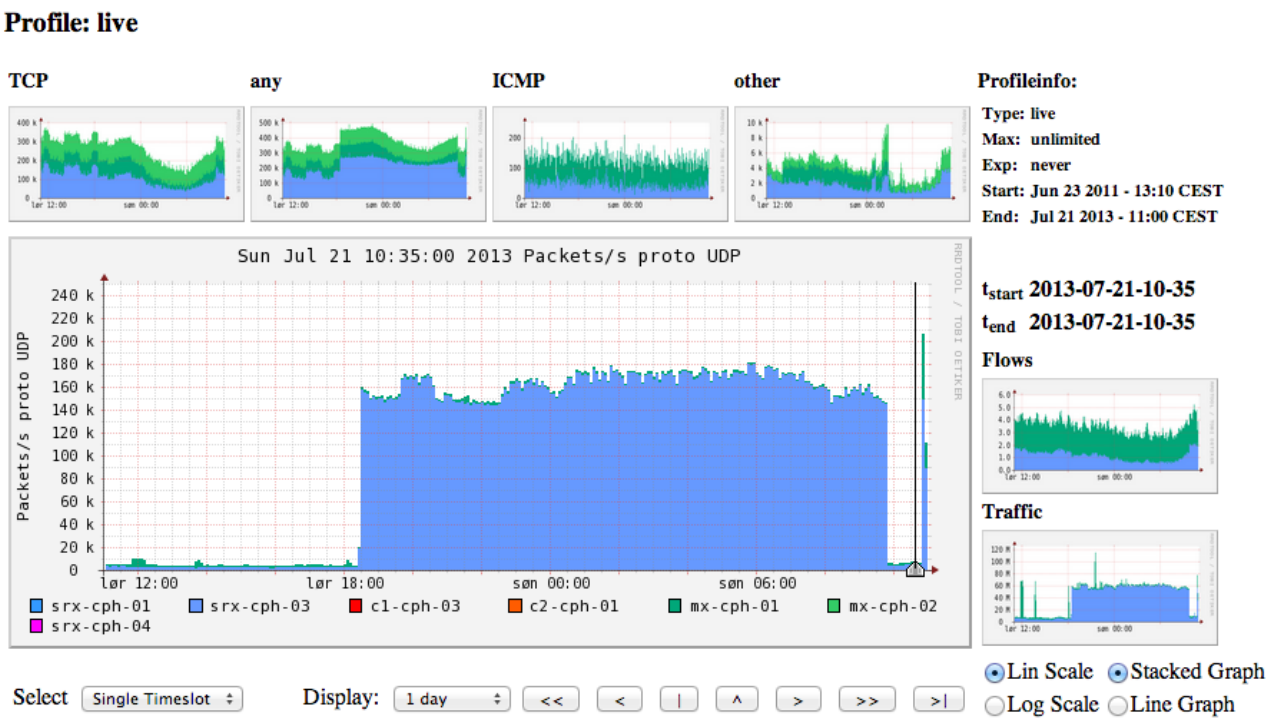
today Netflow version 9 or IPFIX

Source:

https://en.wikipedia.org/wiki/NetFlow

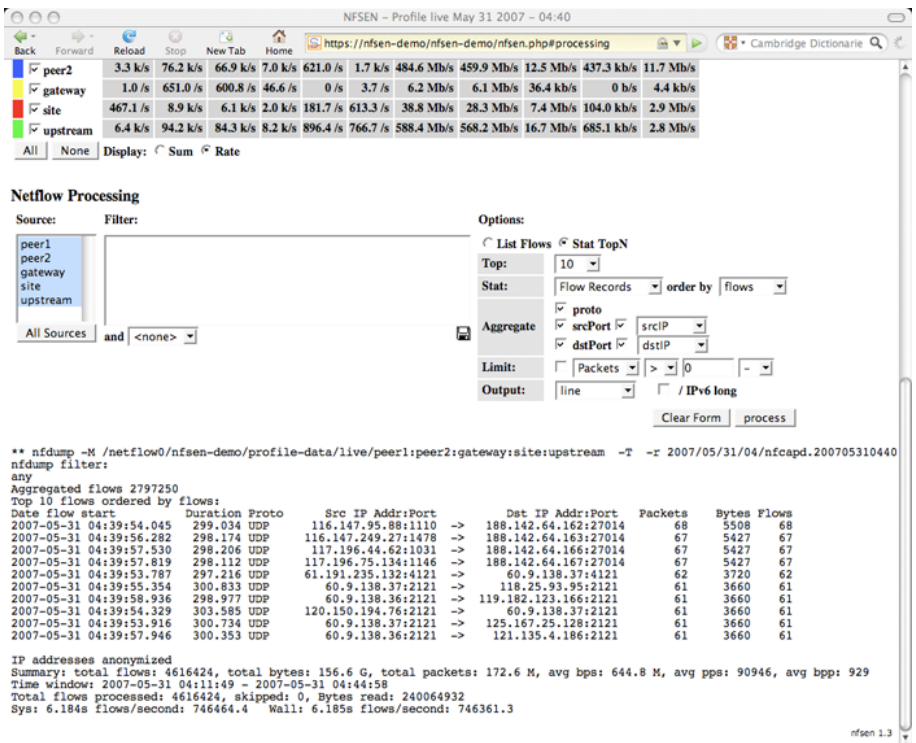
https://en.wikipedia.org/wiki/IP_Flow_Information_Export

Netflow NFSen



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

Netflow processing from the web interface



Bringing the power of the command line forward

The Zeek Network Security Monitor

The Zeek Network Security Monitor is not a single tool, more of a powerful network analysis framework. Note: the project was renamed from Bro to Zeek in Oct 2018

Source https://www.zeek.org/

Zeek scripts

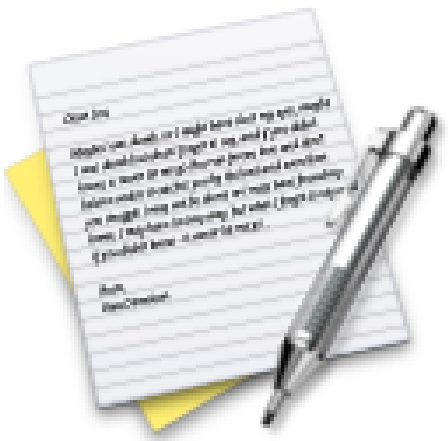
```
global dns_A_reply_count=0;
global dns_AAAA_reply_count=0;
...
event dns_A_reply(c: connection, msg: dns_msg, ans: dns_answer, a: addr)
        {
        ++dns_A_reply_count;
        }

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.zeek.org/sphinx-git/script-reference/scripts.html

Exercise



Now lets do the exercise

Zeek on the web 10min

which is number **17** in the exercise PDF.

Get Started with Zeek

- To run in “base” mode:
  `zeek -r traffic.pcap`
- To run in a “near zeekctl” mode:
  `zeek -r traffic.pcap local`
- To add extra scripts:
  `zeek -r traffic.pcap myscript.zeek`

Zeek demo: Run

// Use the deploy command to initialize and start zeek first

```
debian:~ root# zeekctl

Welcome to ZeekControl 1.5
Type "help" for help.

[ZeekControl] > install
creating policy directories ...
installing site policies ...
generating standalone-layout.zeek ...
generating local-networks.zeek ...
generating zeekctl-config.zeek ...
generating zeekctl-config.sh ...
...
debian:etc root# grep eth0 node.cfg
interface=eth0
```

Our Zeek node.cfg is in/opt/zeek/etc

Zeek demo: Run Zeek

```
[ZeekControl] > start
... starting zeek
// Exit using ctrl-d and then look at logs
debian:zeek root# cd /opt/zeek/logs/current
debian:zeek root# pwd
/opt/zeek/logs/current
debian:current root# tail -f dns.log
```

More examples at:
https://www.zeek.org/sphinx/script-reference/log-files.html

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 the OISF and its supporting vendors.

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

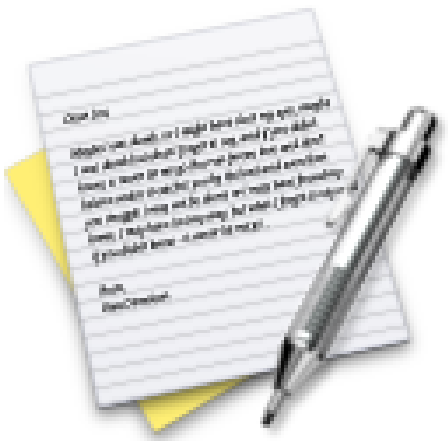
I often use Suricata and Zeek together

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

Exercise

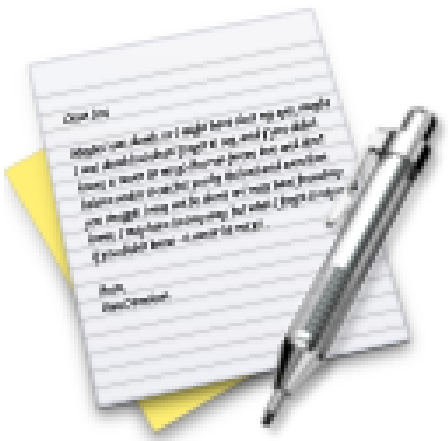


Now lets do the exercise

Zeek on the web 10min

which is number 17 in the exercise PDF.

Exercise

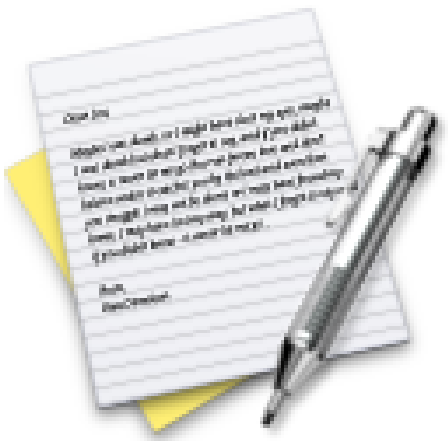


Now lets do the exercise

Zeek DNS capturing domain names – 15min

which is number **18** in the exercise PDF.

Exercise



Now lets do the exercise

Zeek TLS capturing certificates – 15min

which is number **19** in the exercise PDF.

Zeek and Elasticsearch, Filebeat to the rescue

- Lets try to get data from Zeek into Elasticsearch
- Can Zeek produce something nice
- Can Elasticsearch parse something nice
- I think JSON is nice.
- Lucky us: 06 AUGUST 2020ENGINEERING *Collecting and analyzing Zeek data with Elastic Security* by Michael Young
  https://www.elastic.co/blog/collecting-and-analyzing-zeek-data-with-elastic-security
- PS it talks about GeoIP enrichment too!

Zeek packages

```
zkg install
```

- Zeek can be expanded and a lot of packages exist
- https://packages.zeek.org/ 153 packages!
- Lets look at some example scripts like some from CVEs

More Exercises?

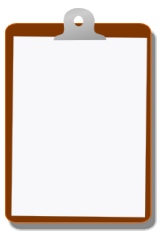
We probably ran out of time, but I wanted to show
https://github.com/crits/crits/wiki which has a large list of CRITs objects

Security Onion

Security Onion is a Linux distro for IDS (Intrusion Detection) and NSM (Network Security Monitoring).
http://securityonion.net

Nice starting point for researching dashboards/network packets

For Next Time



Think about the subjects from this time, write down questions

Check the plan for chapters to read in the books

Visit web sites and download papers if needed

Retry the exercises to get more confident using the tools