Advanced Network Security (2019-2020)

Preventing and detecting network attacks

Harald Vranken

About me

Open University & Radboud University

Office: Mercator I, room 2.16 (Friday)

Email: harald.vranken@ou.nl

Skype: harald.vranken

Web: www.cs.ru.nl/staff/harald.vranken
www.open.ou.nl/hvr

Agenda

- Network attacks
- Intrusion detection systems
- Network flows
- Security application of networks flows

Introduction

- Central theme of this course: availability

- Contents of this course (see course website http://www.cs.ru.nl/~jhh/ans.html):
  - Fault tolerance of distributed systems (*Jaap-Henk Hoepman*)
  - Security in networks and applications (*Harald Vranken*)
    - Preventing and detecting network attacks (Feb. 10)
    - Economics/governance of network security (Feb. 17)
    - Wifi security (March 16)
    - Routing security: BGP and future internet architecture (May 4, *Joeri de Ruiter*)
    - Botnets (May 25)
    - Mobile telephony security (June 8, *Fabian van den Broek*)

Introduction

Network attacks

- Attack through the network: network provides means to launch attack
  - DDoS attack against internet-banking webserver
  - sending phishing emails
- Attack on the network: network itself is target of attack
  - DDoS attack to overload network components (routers)
  - BGP hijacking
- Combination
  - DDoS attack by Mirai botnet against Dyn's DNS name servers, Oct. 2016

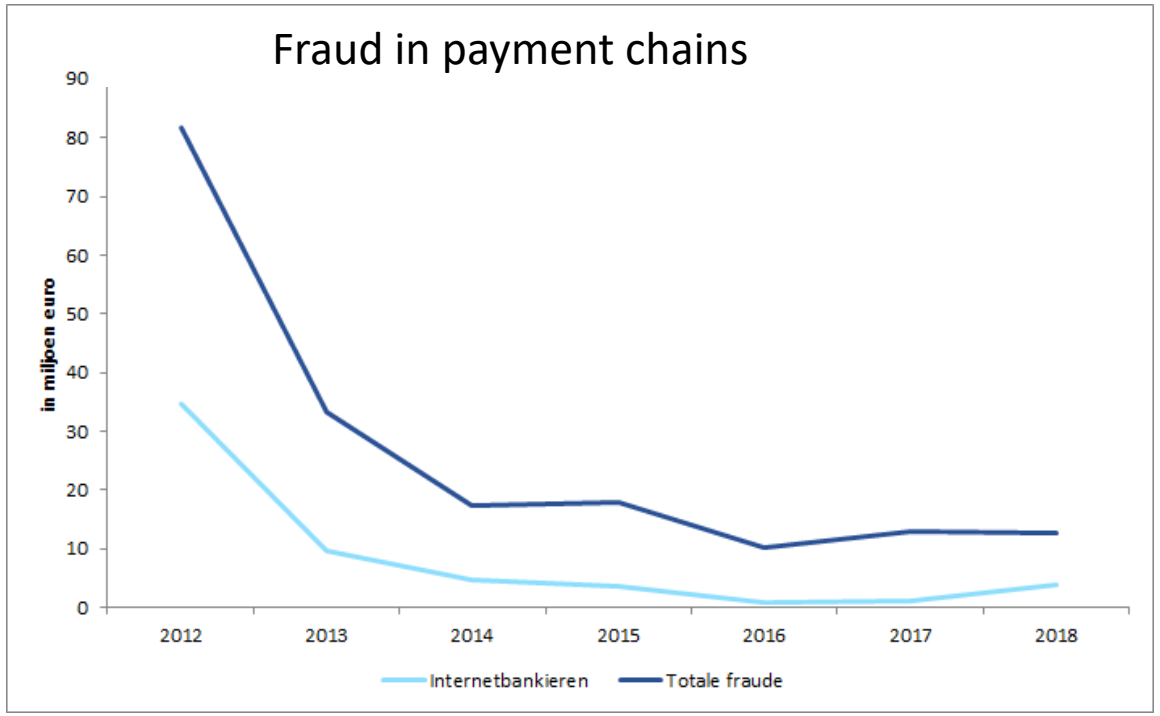
Introduction

- Dealing with network attacks
  - Prevention
  - Detection
  - Response

- Prevention would be best, but not always possible nor ‘waterproof’
  - “The only truly secure system is one that is powered off, cast in a block of concrete and sealed in a lead-lined room with armed guards - and even then I have my doubts.” (*Gene Spafford, 1989*)

Internet banking fraud in the Netherlands

- Prevention works!
  - Fraud mainly due to malware and phishing
  - Banks monitor transactions and can detect and prevent fraud faster
  - Campaigns on ‘veilig bankieren’ by internet, radio and TV made customers aware of methods applied by criminals, and what banks never ask
  - Block sites that that are mentioned in phishing mails



Source: NVB and Betaalvereniging Nederland

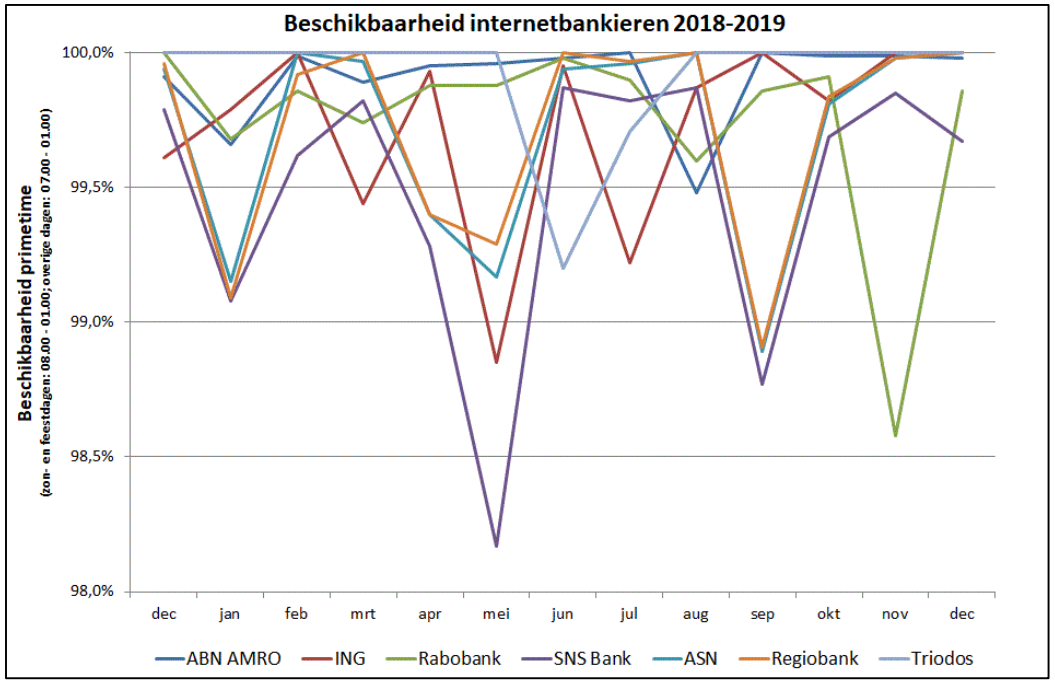
- But, the battle has not been won yet: fraud mainly due to phishing is increasing:
  - 1.05 M€ in 2017
  - 3.81 M€ in 2018
  - 3.08 M€ in first half of 2019 (phishing via mobile service, like SMS, WhatsApp and Messenger)

Availability of payment chains in the Netherlands

Source: NVB and Betaalvereniging Nederland

- Formal/legal requirements on availability
  - Dutch National Bank required 99.88% (2018) for chip-and-pin and contactless
  - Dutch law dictates that online banking services may not be interrupted for more than two hours at a time

Availability (%)	2016	2017	2018	2019
Chip-and-pin and contactless	99.88	99.88	99.89	99.89
Internet banking (via websites)	99.79	99.83	99.72	99.78
Mobile banking (via apps)	99.77	99.83	99.75	99.81



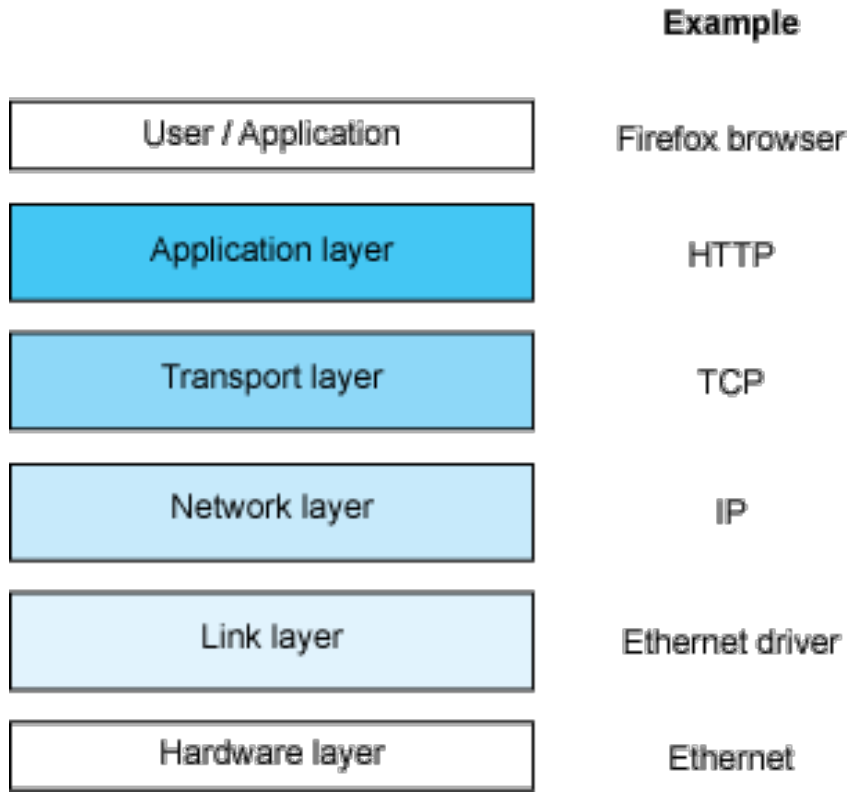
- Monitoring
  - Currence monitors real-time availability of iDEAL (leading Dutch online payment method) www.ideal.nl/en/latest-news/keyfigures/ideal-availability/
  - Dutch Payments Association monitors availability of internet banking and mobile banking www.betaalvereniging.nl/en/payment-products-services/availability-mobile-and-internet-banking/

Operational security

- Provide service to genuine users
- Keep attackers and malicious users out
- How can we achieve this (with technology)?
  - Prevention: for example by separating and limiting network traffic
  - Detection: for example by monitoring and inspecting network traffic

Network stack

- Prevention and detection can be applied on different layers of the network stack
  - Physical/link layer
  - Network/transport layer
  - Application layer



Prevention on the physical layer

- Use physically separated network
- For example:
  - Alliander's wireless CDMA network
    (for connecting to 'smart meters',
    and 'smart grid' to make network intelligent)
  - Fiber-optic cables in power grid
    (between high-voltage substations)
- Not enough as your only defense
  - Remember Stuxnet!

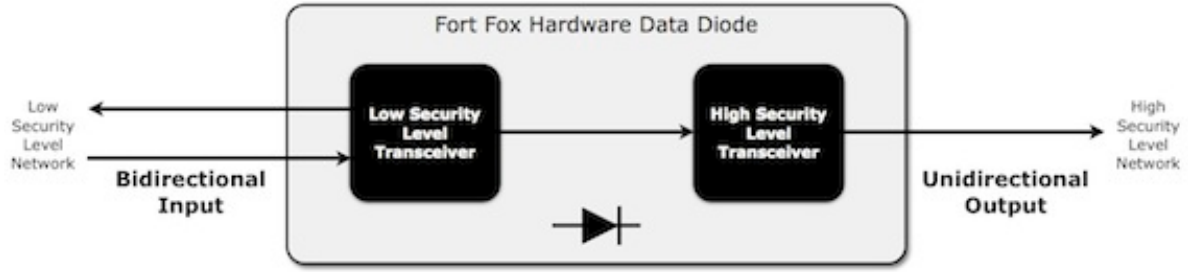
Source: www.youtube.com/watch?v=KzvaShAyK64

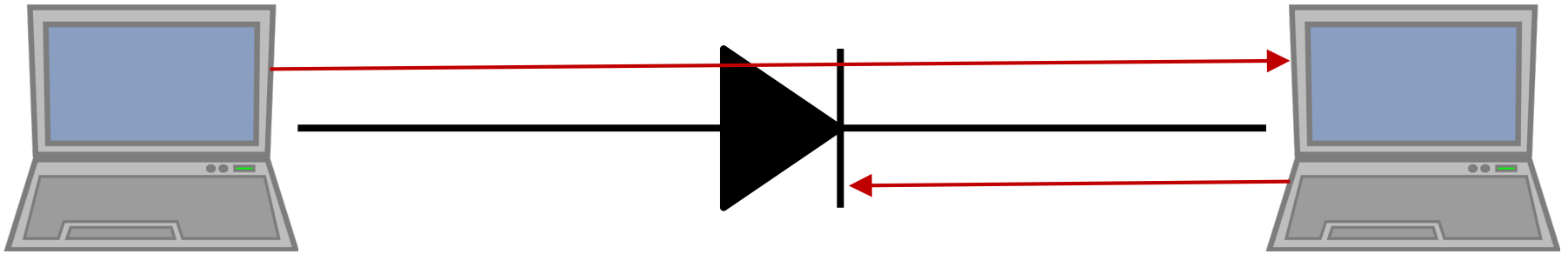
Prevention on the physical layer

- Use multiple networks with different security policies
  - E.g. used in the military
- Air gapping: physically separate networks
- How do you get information from one network to the other?
  - For example, using a data diode

Data diode

- Data allowed to only go in one direction
- Can be physically enforced
  - For example, by using optical signals
- No reliable data transfer!





Prevention on the physical layer

- Also unintended ways: covert channels
- Example: hacked surveillance/security cameras
  https://youtu.be/om5fNqKjj2M
  - Exfiltration: malware can control infrared LEDs in cameras, and leak info to attacker at a distance
  - Infiltration: remote attacker can send infrared light pulses, which are observed by cameras

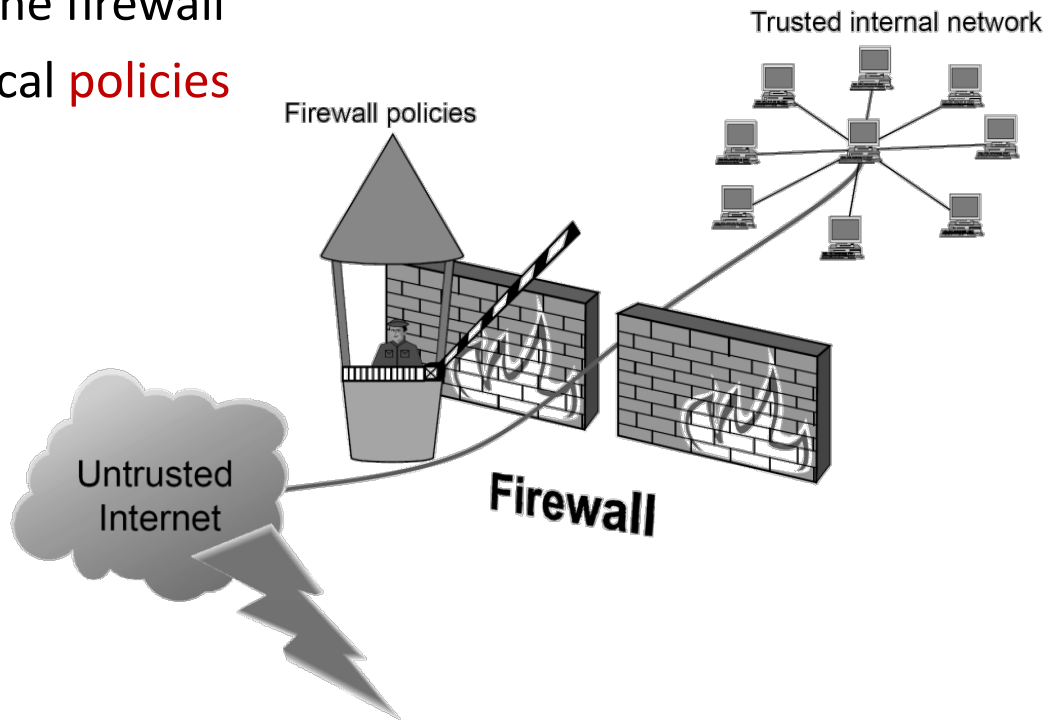
aIR-Jumper: Covert air-gap exfiltration/infiltration via security cameras & infrared (IR)
M Guri, D Bykhovsky
Computers & Security 2019, 82, 15-29

Prevention on the network layer

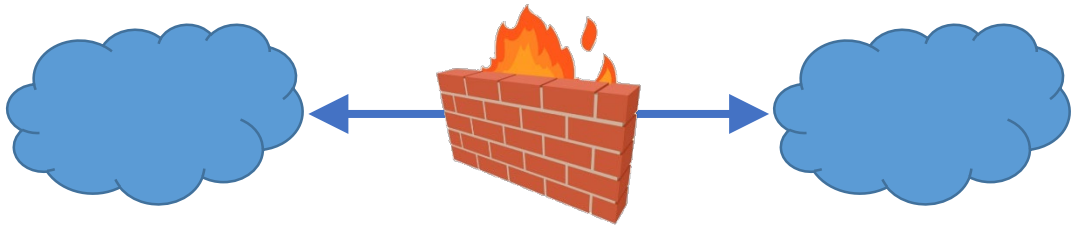
- Segmentation of your network
- For example, different (virtual) networks for employees/students and visitors
- Apply different policies for different networks

Firewalls

- Conceptually separates two networks
- Access control between outside world and internal resources
- Three goals
  - All traffic between inside and outside passes the firewall
  - Only authorized traffic is allowed, following local policies
  - The firewall itself is immune to penetration
- Different types of firewalls
  - Traditional packet filters
  - Stateful filters
  - Application gateways



Firewalls

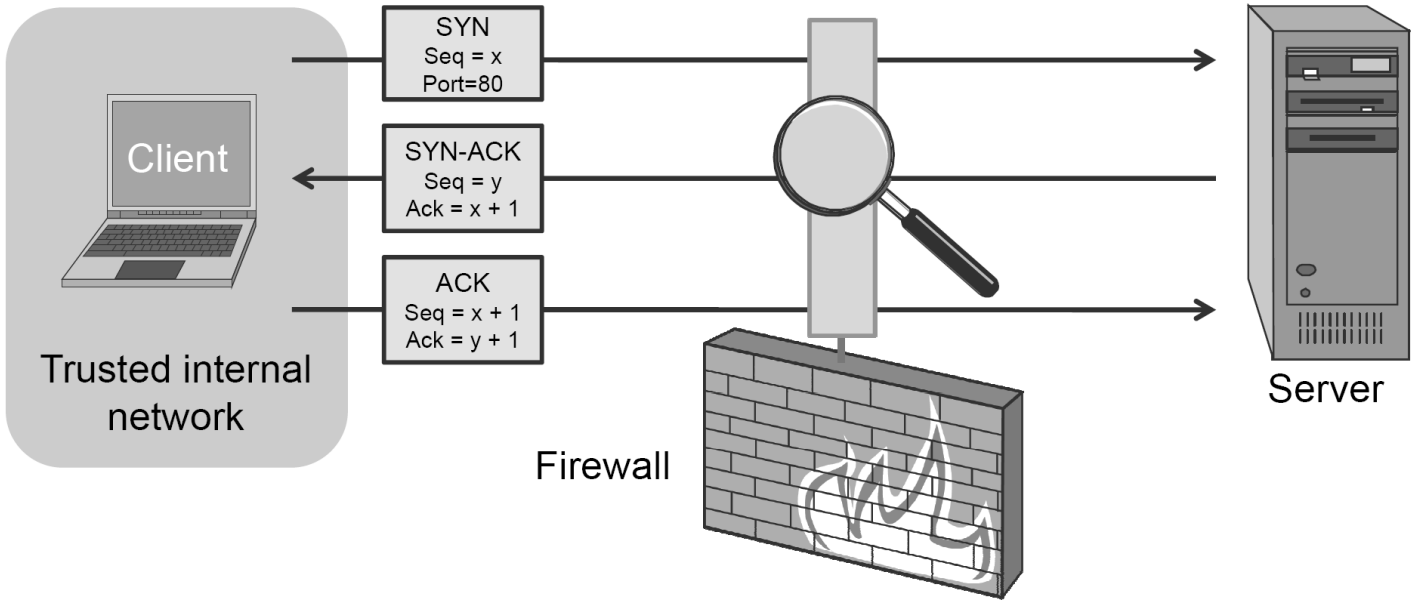


- Packet filter: control packets travelling between two networks
  - accepted: permitted through the firewall
  - dropped: not allowed through with no indication of failure
  - rejected: not allowed through, attempt to inform source that packet was rejected

- Packets allowed or dropped based on policies
  - Protocol type (e.g. TCP, UDP, etc)
  - TCP or UDP source and/or destination port number
  - IP source and/or destination address
  - TCP flags
  - Direction (incoming or outgoing)
  - Interface

Stateless firewalls

- Example: allow packets to/from port 80



Allow outbound SYN packets, destination port=80
Allow inbound SYN-ACK packets, source port=80

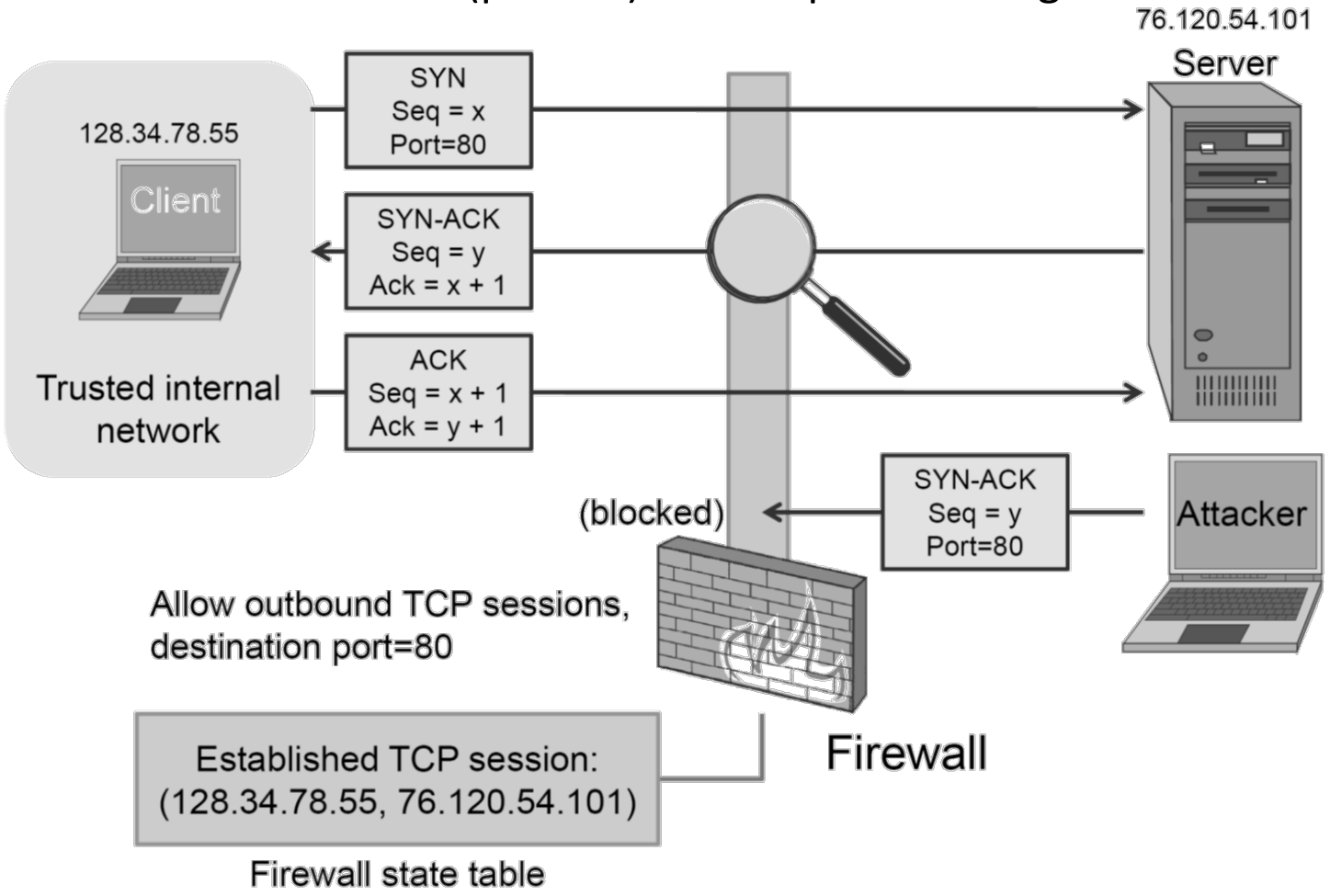
Firewalls

- Stateless packet filters
  - Look at one packet at a time
  - Very efficient
  - Does not know whether packets belong to an existing TCP connection

- Stateful packet filters
  - Track TCP connections
  - Connection table containing source and destination address, source and destination port
  - Observe three-way handshake (SYN, SYN/ACK and ACK) and closing of connection (FIN)
  - Can be used, for example, to define policies to only allow outgoing TCP connections

Stateful firewalls

- Example: allow TCP web sessions (port 80) with request coming from inside the trusted network

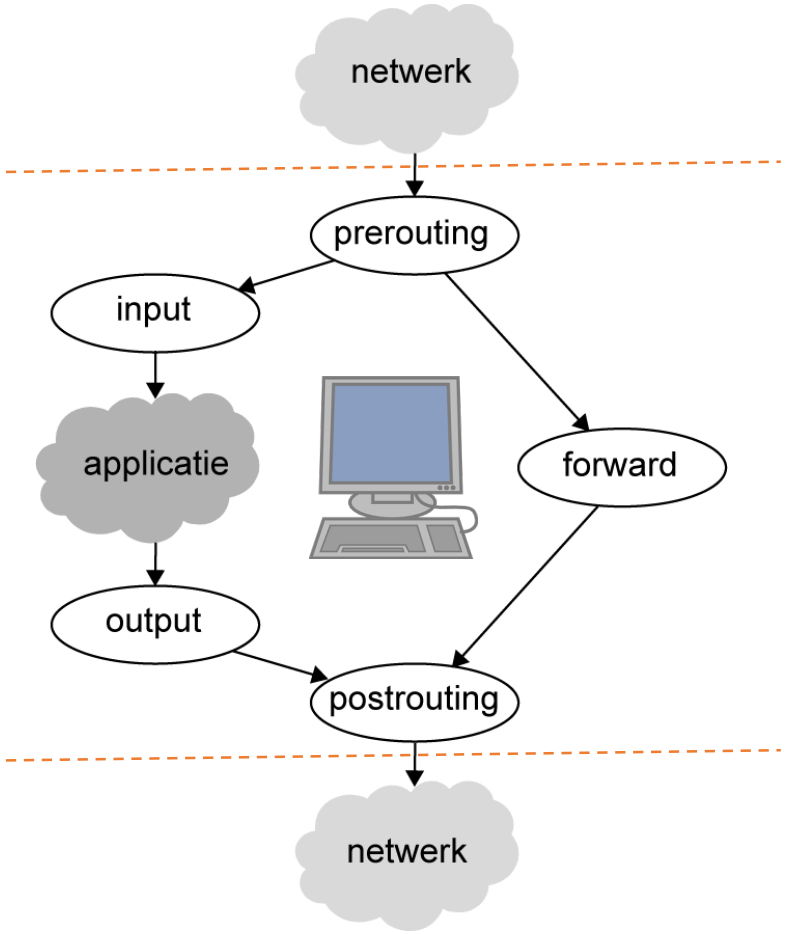


Firewall with iptables

Example

- Firewall implemented with iptables
  on host with networkinterface eth0 and IP-adres 130.0.0.10
- Configuration

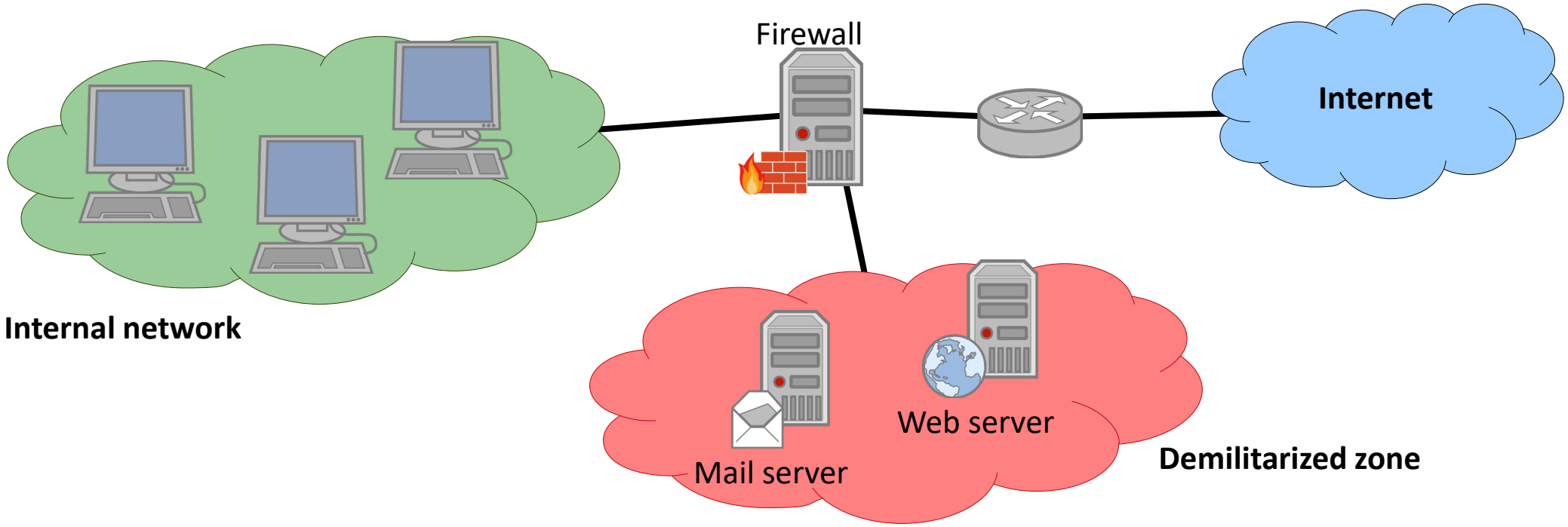
```
Chain INPUT (policy DROP)
Num      target   prot     in       out      source      destination
1        ACCEPT   all      eth0     *        120.0.0.0/8 130.0.0.10

Chain OUTPUT (policy DROP)
num      target   prot     in       out      source      destination
1        ACCEPT   all      *        eth0     130.0.0.10  0.0.0.0/0
```



- Can an application on this host do IP address spoofing?

Demilitarized zone (DMZ)

- Put services that should be accessible from the outside world in a separate network
- If a service gets compromised, the attacker does not yet have access to the internal network
- Different firewall rules for internal network and DMZ



Cryptography

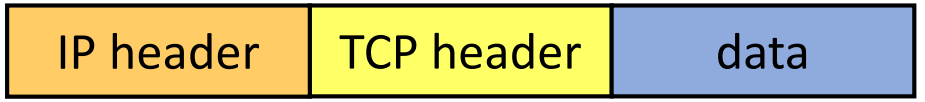
- Can be used to secure different layers of the network stack
- Link layer
  - Wi-Fi
- Network layer
  - VPN (Virtual Private Network)
  - For example, IPsec and OpenVPN
- Transport layer
  - TLS (Transport Layer Security)
- Application layer
  - PGP or S/MIME for email

IPsec

- TCP/IP
  - Application layer

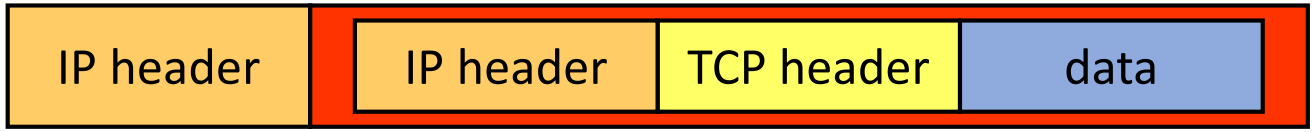
  - Transport layer (segment)

  - Network layer (datagram)



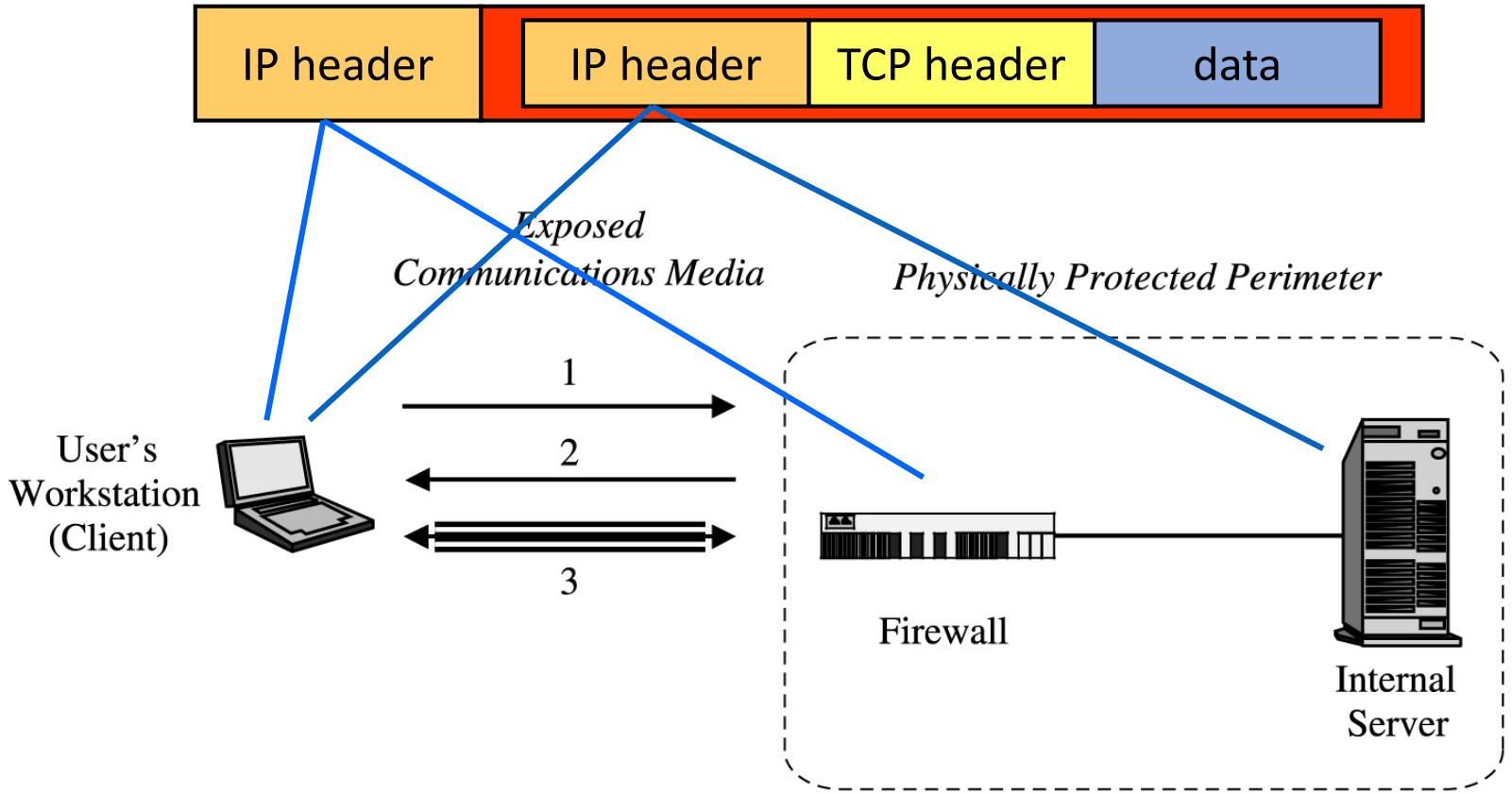
- IPsec
  - Transport mode
    - Payload data is segment from transport layer

  - Tunnel mode
    - Payload data is datagram from network layer (*IP in IP*)



VPN with IPsec

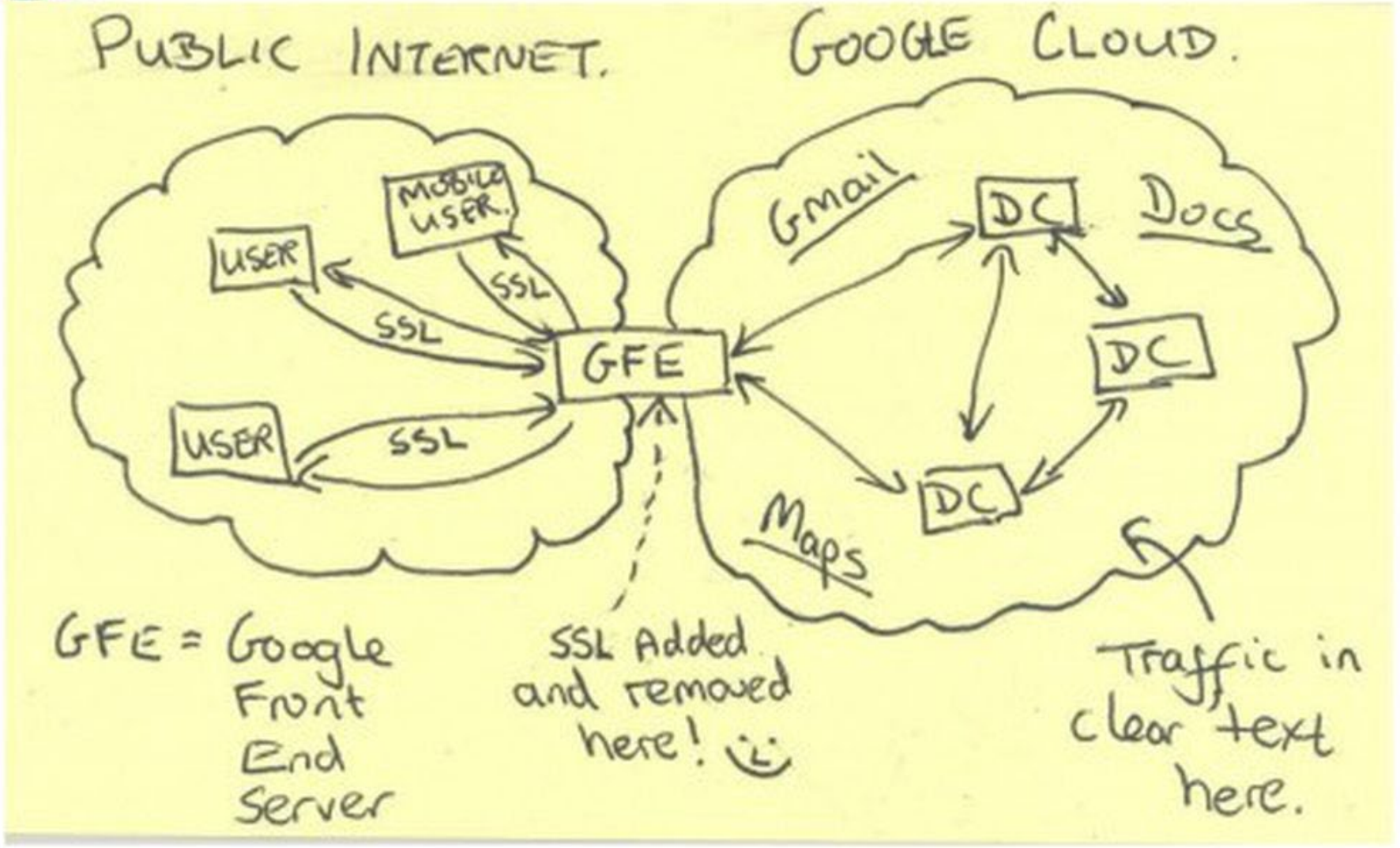
- IPsec in tunnel mode



Cryptography

- Many potential problems
  - Cryptographic algorithms need to be secure
  - Security protocol needs to be secure
  - Both need to be implemented correctly
  - Both need to be configured correctly

Current Efforts - Google



Edward Snowden: NSA secretly broke into main communications links that connect Yahoo and Google data centers around the world.

Network Segmentation

- Of course organisations have divided their networks into segments, right?

  – Maastricht University (2019): infected by ransomware
    - Windows domain: administrator account also used for administration of ‘regular’ servers
    - UM network segmented in V-LANs (with rather open connections)

  One of the recommendations: improve network network segmentation

  fox_it_rapport_reactie_universiteit_maastricht.pdf

Agenda

- Preventing attacks
- Intrusion detection systems
- Network flows
- Security application of networks flows

Intrusion Detection/Prevention

- Intrusions detection/prevention can take place on the end-points or in the network
- Inspect network traffic to determine whether malicious activity is taking place
  - Packet contents (Deep Packet Inspection or DPI)
  - Packet headers (metadata)
- Passive
  - Intrusion detection systems (IDS)
  - Example actions: generate logs and alerts
- Active
  - Intrusion prevention systems (IPS)
  - Example actions: kill network connections and ban IP addresses

Intrusion Detection/Prevention

- Signature-based or rule-based
  - Can detect known attacks by looking for signatures
  - Rules encode signature for a specific attack
  - Most widely deployed
  - Well-known open source IDS/IPS system is Snort (https://snort.org/)
    - Example: ARP-poisoning attack

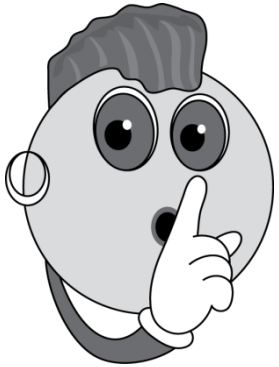
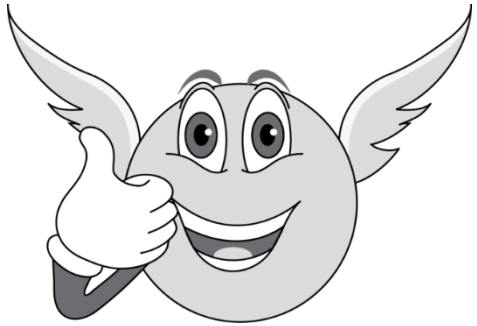
Intrusion Detection/Prevention

- Anomaly based
  - Try detect suspicious behaviour
  - Can detect unknown attacks
  - Needs to know/learn about normal traffic
  - Can have a high false-positive rate (why is this an issue?)

- Statistical
  - Build profile (statistical representation) of typical ways that user acts or host is used
  - Determine thresholds for anomalous behaviors

Intrusion detection systems

- Intrusion detected (positive) or not (negative)

	Intrusion Attack	No Intrusion Attack
Detected	 True Positive	 False Positive
Not detected	 False Negative	 True Negative

Base-rate fallacy

- Difficult to create IDS with both high true-positive rate and low false-positive rate
- Effectiveness of IDS can be reduced if IDS accuracy and number of intrusions are relatively small
  - Statistical error known as the base-rate fallacy
  - Example
    - IDS is 99% accurate (having a 1% chance of false positives or false negatives)
    - 1,000,000 benign events and 100 malicious events
    - Of the 100 malicious events: 99 true positives, 1 false negative
    - Of the 1,000,000 benign events: 990,000 true negatives, 10,000 false positives
    - Hence, there will be 10,099 positives reported, of which 10,000 are false alarms: 99% false alarms!
  - Precision: TP/(TP+FP) = 99/(99+10,000) = 0.01 (measure of exactness or quality)
  - Recall: TP/(TP+FN) = 99/(99+1) = 0.99 (measure of completeness or quantity)

Alternative detection methods

- Honeypots
  - Vulnerable systems specifically set up to attract/detect attackers in your network

- Canary tokens
  - Specially prepared data that trigger an alert when, for example, accessed
  - Can be a URL, directory on a file server, an email address, etc.

Agenda

- Preventing attacks
- Intrusion detection systems
- Network flows
- Security application of networks flows

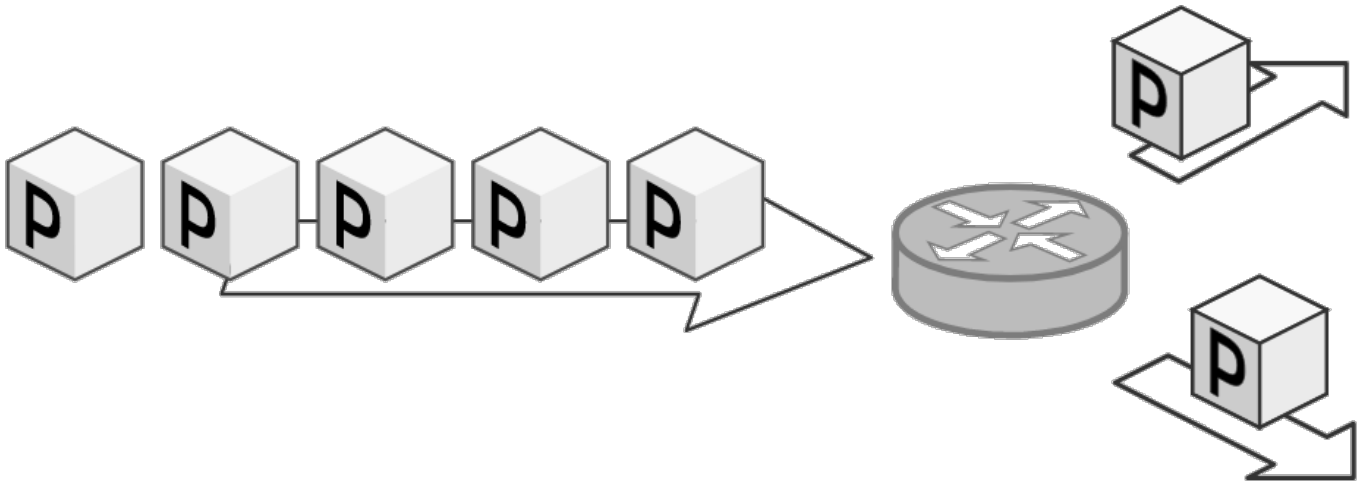
Network traffic

- Analysing network data: at what level?
  - bits, bytes, protocol header (field), packets, …
- Analysis of packet contents is
  - Expensive
  - Privacy sensitive
  - Often not even possible (due to encryption)
  - Unfeasible for huge amounts of data

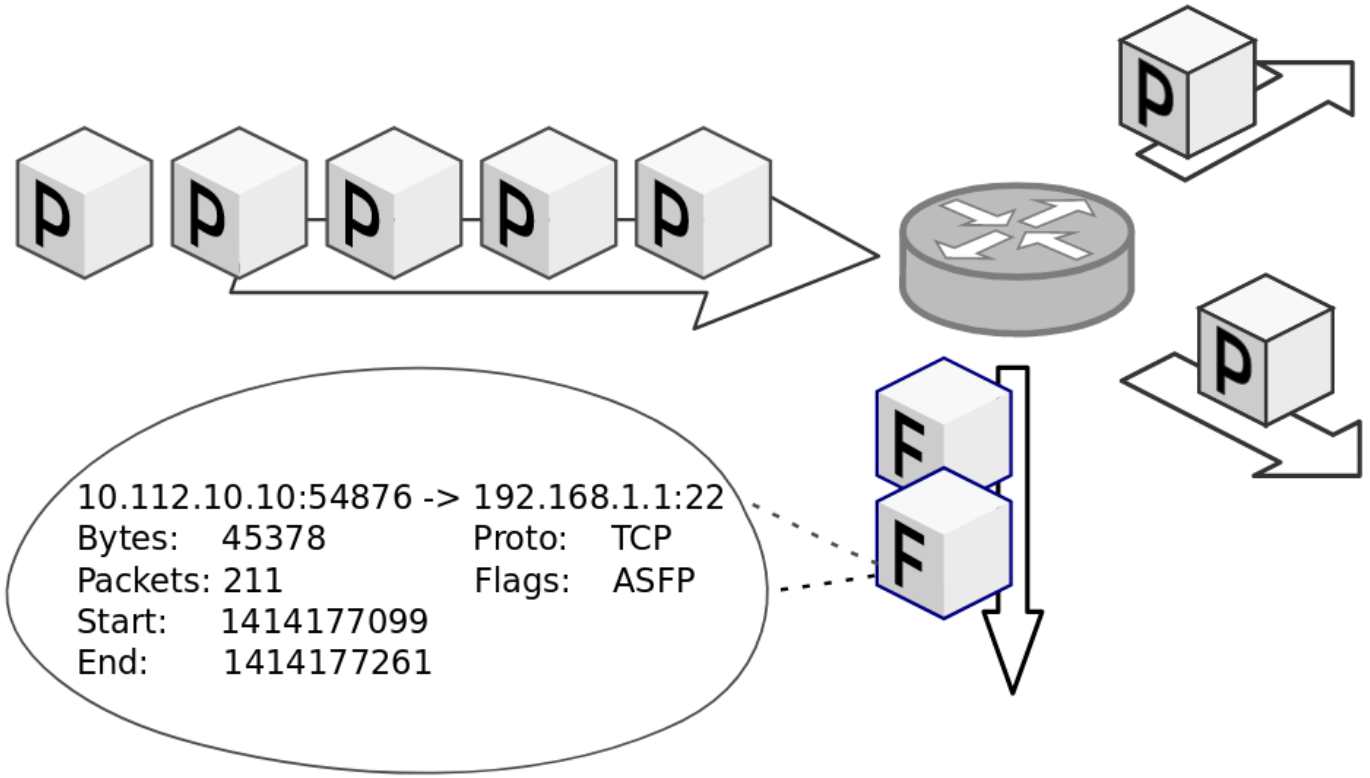
Network flow data

- Limit collection and analysis to flows
- "A flow is defined as a set of IP packets passing an observation point in the network during a certain time interval. All packets belonging to a particular flow have a set of common properties." (RFC 5101)
- Examples of common properties are
  - Source and destination IP addresses
  - Source and destination port numbers
  - Protocol type
- Metadata: who talks to whom, how much, and when
- Even this is not a trivial task when you have links that process gigabits of data per second!

Network flow data



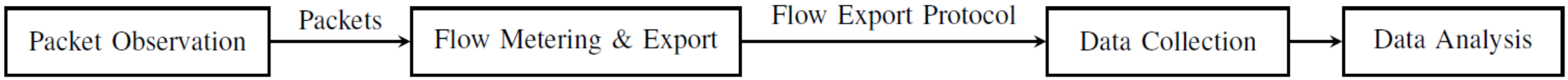
Network flow data



Network flow data

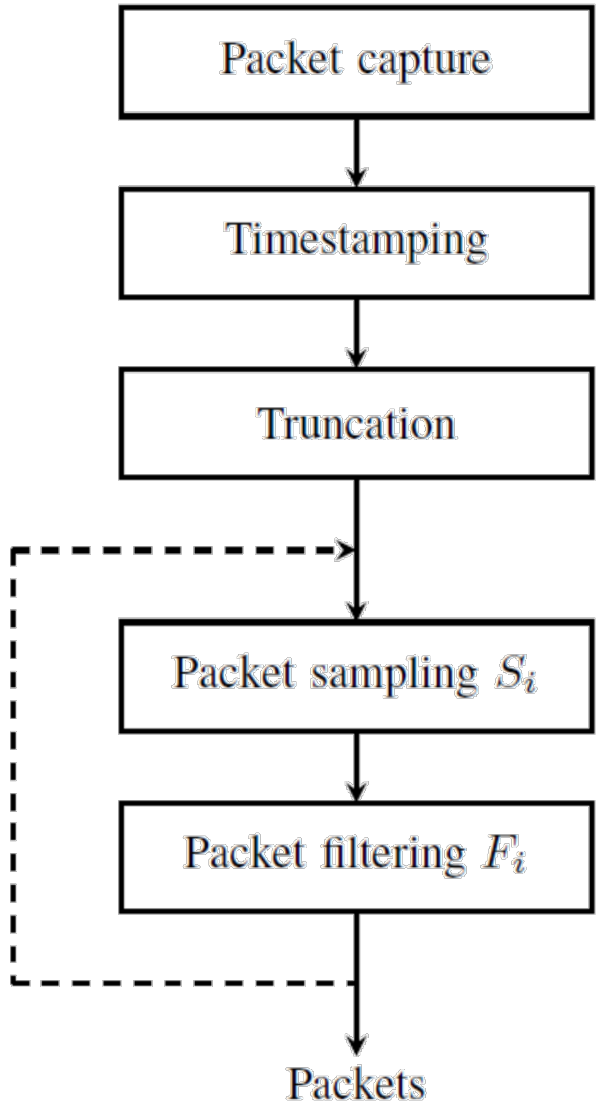
- Aggregate data over specified time periods
  - Miss outliers in the data
- Much less data generated
  - UTwente dataset: 2.1 TB packet capture results in 820 MB of flow data (for IPFIX) (0.04%)

Flow monitoring architecture



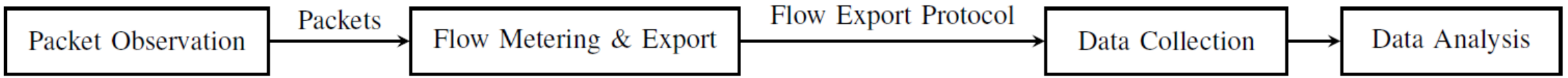
Packet observation

- Read packets from the network link and a add timestamp
  - Can be done inside packet forwarding devices or using dedicated devices
  - Eg. using a prisma to “copy” the network data in a fibre link
- Optionally truncate, sample, and filter to reduce load on subsequent steps
- Packet truncation to select only bytes of interest
  - Eg. select packet header fields, ignore packet payloads
- Packet sampling to select subset of packets subset
  - Still being able to estimate properties of full packet stream
  - Random sampling
  - Systematic sampling (e.g. every n-th packet)
- Packet filtering to remove all packets that are not of interest
  - Eg. IP addresses, port numbers, protocol type
- Resulting packets are forwarded to next stage



Flow metering and export

- Packets aggregated into flow records
  - Defined by flow key,
    for example (source IP, dest. IP, source port, dest. port, protocol type)
  - Data collected typically includes: start and end time, number of packets, number of bytes exchanged
- Flow records kept in flow cache
- Flows exported when it expires, for example:
  - Active timeout
  - Idle timeout
  - Resource constraints
  - Natural expiration



Flow metering and export

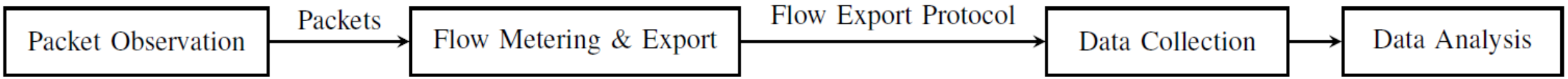
- Flow records combined for export
  - For example, using IPFIX or NetFlow messages
- Exported to collector using transport protocol
  - TCP
    - Pro: optimised and can be secured using TLS
    - Con: no graceful degradation
  - UDP
    - Pro: easy to implement and low overhead
    - Con: only best-effort delivery and no congestion control (potential DoS)
  - SCTP (Stream Control Transmission Protocol)
    - Pro: reliable transmission with congestion control and graceful degradation
    - Con: support lagging and protocol number might be unknown

Data collection

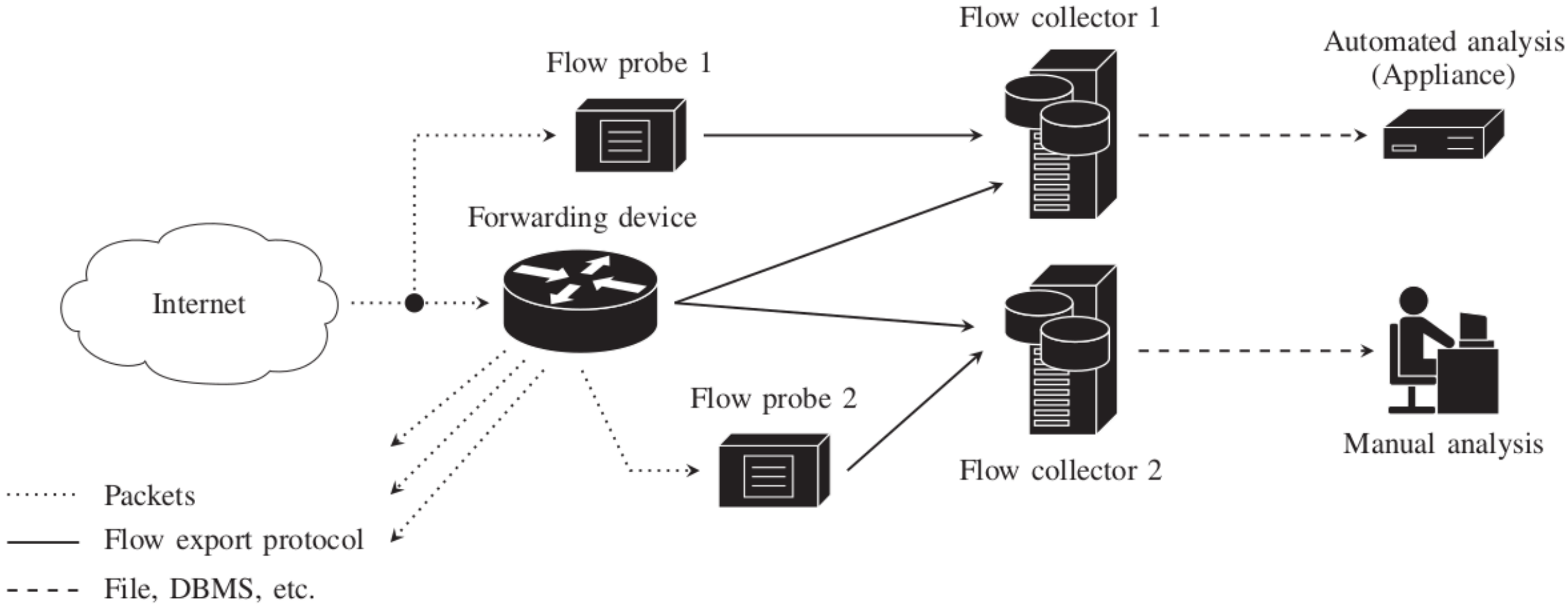
- Flow collectors can combine flows from different flow exporters
- Stores the flows for actual analysis
- This data can be highly privacy sensitive
  - IP address is considered to be personal data

Data analysis

- What can we (ie. network operators) do with this data?
  - Flow analysis and reporting
  - Performance monitoring
  - Intrusion detection



Flow monitoring architecture



Source: Flow Monitoring Explained: From Packet Capture to Data Analysis With NetFlow and IPFIX, by Rick Hofstede, Pavel Čeleda, Brian Trammell, Idilio Drago, Ramin Sadre, Anna Sperotto, Aiko Pras

Agenda

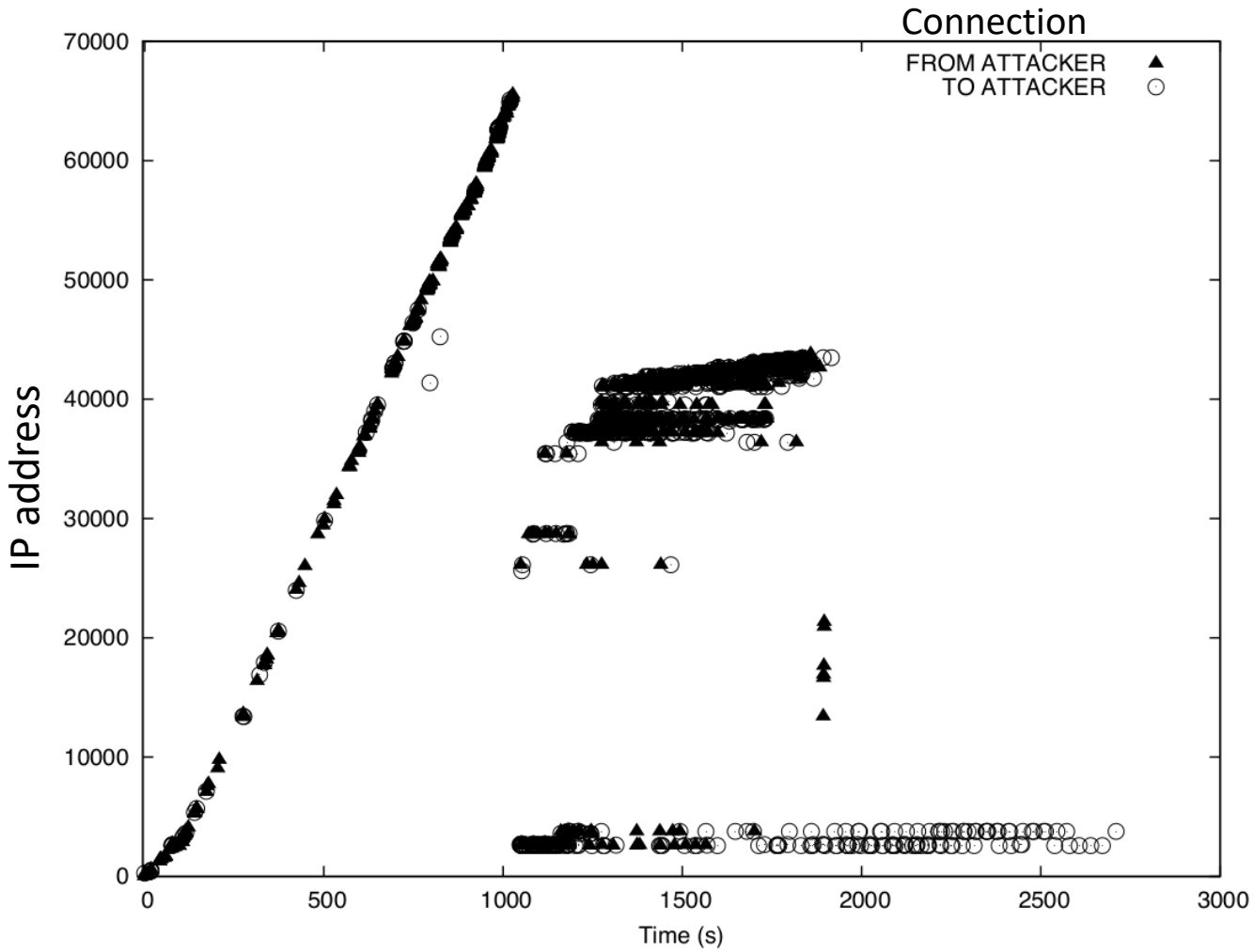
- Preventing attacks
- Intrusion detection systems
- Network flows
- Security application of networks flows

Intrusion detection using netflow data

- Specific types of attacks can be observed directly in network flows, such as
  - (D)DoS attacks
  - Network scans
  - Worm spreading
  - Botnet communication
- Detect internal hosts communicating with blacklisted hosts

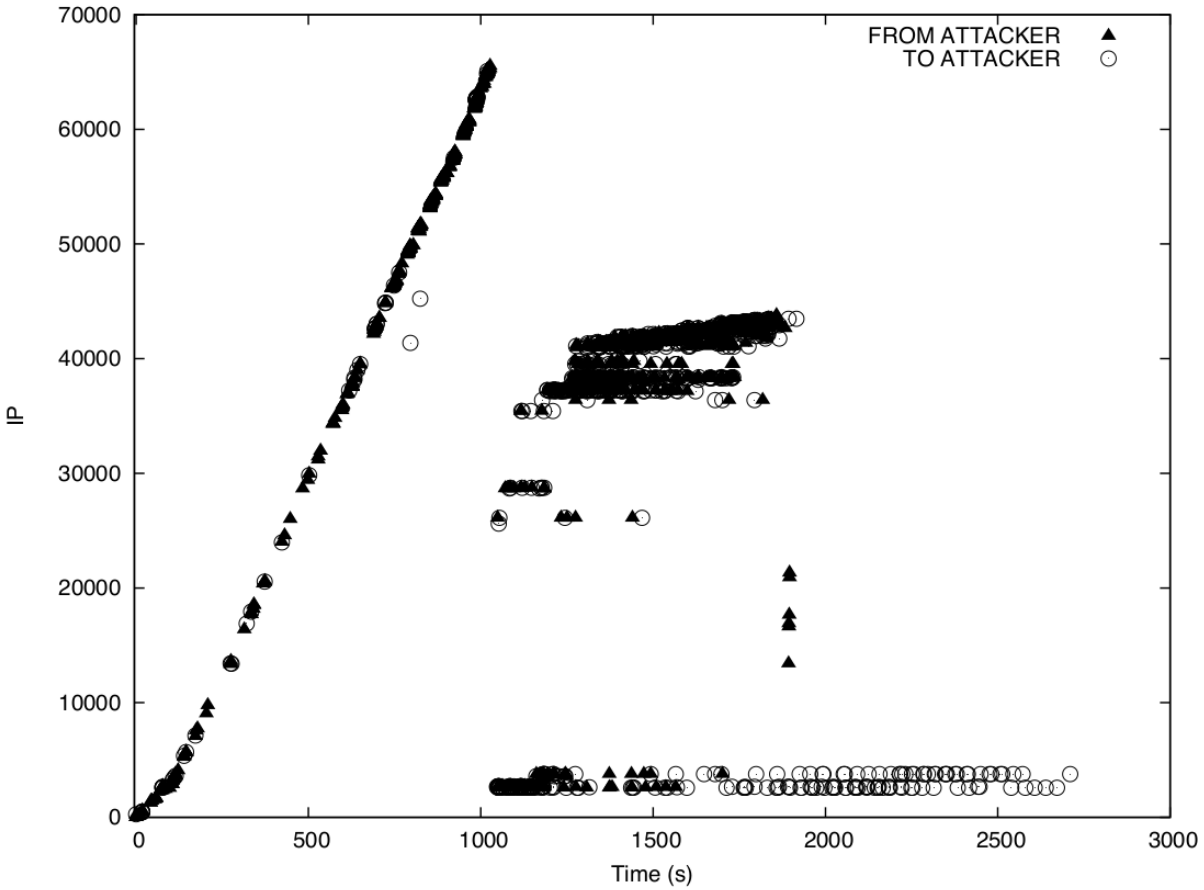
Example: SSH attacks

- Very common attacks targeting poorly configured SSH servers
- Attack consists of three phases
  - Scan phase: scan for SSH servers
  - Brute-force phase: perform dictionary attack on discovered servers
  - Compromise phase: log in to and use compromised servers
- Every phase has very specific flow characteristics
- Not all phases can be detected at host level

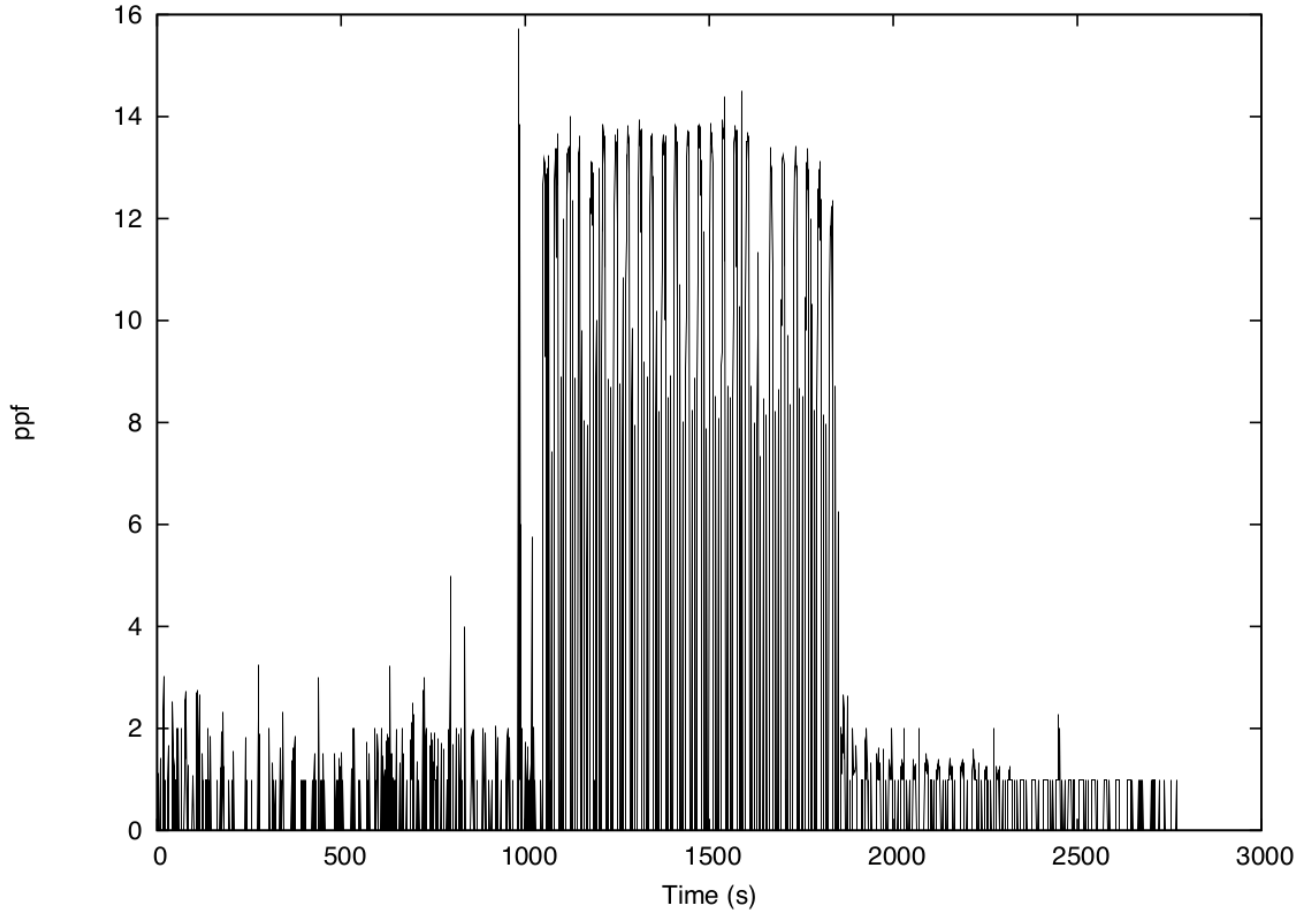


SSH attacks

IP addresses vs. time

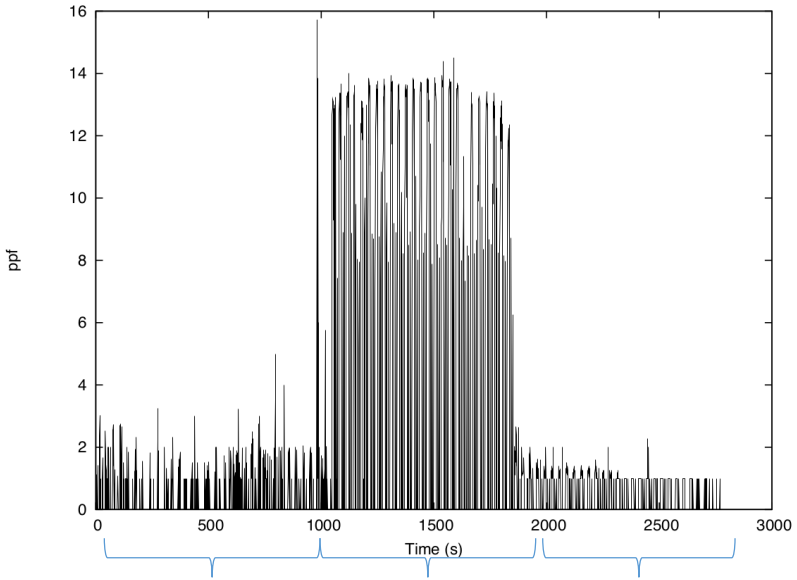


Packets per flow (ppf) vs. time



Source: Hidden Markov Model Modeling of SSH Brute-Force Attacks, by Anna Sperotto, Ramin Sadre, Pieter-Tjerk de Boer, and Aiko Pras

SSHCure



- Detect SSH attacks using netflow data
  - Plugin for the open source NfSen
  - Try to detect differences and changes between phases
- Scan phase
  - Packets-per-flow: very low
  - Minimum number of flow records/s: fairly high (many hosts scanned)
- Brute-force phase
  - Packets-per-flow: traffic needed for three failed SSH logins
  - Minimum number of flow records/s: high (many login attempts)
- Die-off phase
  - Change in behaviour from brute-force phase which might indicate compromise

SSHCure

Dashboard

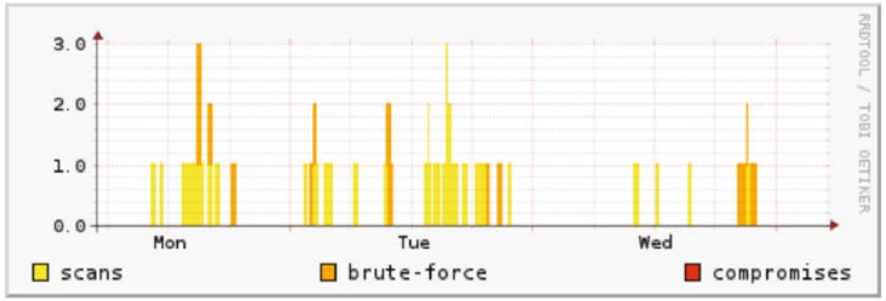
SSHCure
Keep your SSHells SSHafe!

UNIVERSITY OF TWENTE.

Dashboard — Help About License

Time range: 3 days **from** Mon. Mar 19, 2012 0 — **Mon. Mar 19, 2012 - Thu. Mar 22, 2012**

Attacks



Date	Ongoing	Phases	Attacker	Targets
Wed. Mar 21, 2012 11:59			195.3.152.159	3301
Mon. Mar 19, 2012 14:33			14.132.97.93	3204
Mon. Mar 19, 2012 14:38			126.164.235.252	1012
Mon. Mar 19, 2012 15:38			126.164.235.252	880
Mon. Mar 19, 2012 18:00			126.164.230.151	647
Tue. Mar 20, 2012 20:17			126.164.224.205	586
Tue. Mar 20, 2012 19:09			126.164.224.205	572
Tue. Mar 20, 2012 09:13			11.26.202.65	160
Wed. Mar 21, 2012 20:09			178.144.120.41	41

Top attackers

Attacker	Attacks	Targets (distinct)	Targets (total)
126.164.224.205	6	1908	4492
126.164.230.151	4	1757	3580
126.164.235.252	3	3030	3938
178.144.120.41	3	3420	3468
22.175.168.68	3	83	90
126.164.228.143	2	2751	4340
195.3.152.159	1	3301	3301
14.132.97.93	1	3204	3204
22.147.218.10	1	3111	3111
14.34.127.2	1	2609	2609

Top targets

Target	Attackers (distinct)	Attacks	Compromises
126.164.227.237	16	17	17
126.164.235.64	16	17	17
126.164.239.170	15	20	20
126.164.239.56	15	16	16
126.164.229.30	14	15	15
126.164.229.136	14	15	15
126.164.229.80	14	15	15
126.164.228.26	14	15	15
126.164.228.223	14	15	15
126.164.232.139	14	15	15

Source: SSHCure: A Flow-Based SSH Intrusion Detection System,
by Laurens Hellemons, Luuk Hendriks, Rick Hofstede, Anna Sperotto, Ramin Sadre and Aiko Pras

SSHCure

Attack details

SSHCure
Keep your SSHells SSHafe!

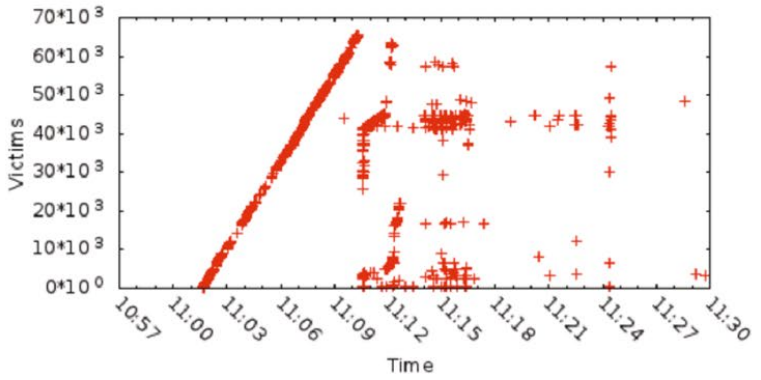
UNIVERSITY OF TWENTE.

Dashboard » *Attack details*

Help About License

Basic information

Attacker	65.202.103.64
Start time	July 13, 2008 11:01
End time	July 13, 2008 11:29
Phases	
Total flows	
Total packets	
Total bytes	



Targets (14221)

Target	Phases
126.164.35.13	
126.164.35.107	
126.164.35.100	
126.164.35.117	
126.164.35.10	
126.164.35.150	
126.164.35.133	
126.164.35.166	
126.164.35.19	
126.164.35.229	
126.164.35.231	
126.164.35.237	
126.164.35.73	
126.164.35.109	
126.164.41.1	
126.164.41.37	
126.164.41.45	
126.164.41.46	

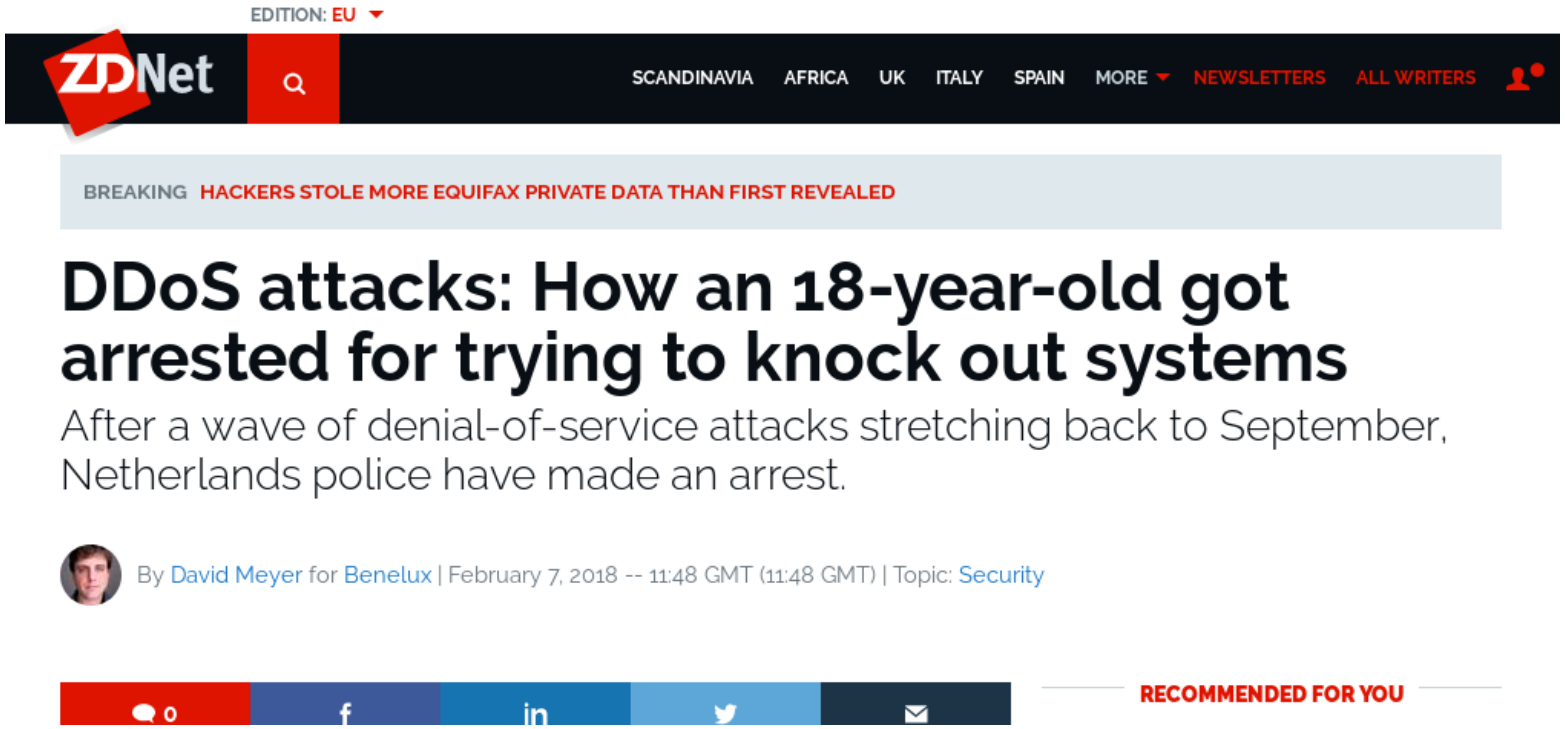
Flows

Start	End	Duration	Flags	D	Packets	Bytes
11:01:39	11:01:39	0.128	.....	←	3	187
11:01:39	11:01:39	0.320	.....	→	5	256
11:10:37	11:10:40	3.584	.....	←	16	3143
11:10:37	11:10:41	3.776	.....	→	12	1168
11:10:39	11:10:44	5.248	.....	←	14	2583
11:10:39	11:10:45	5.440	.....	→	13	1228
11:10:44	11:10:47	2.368	.....	→	12	1152
11:10:44	11:10:47	2.176	.....	←	13	2523
11:10:47	11:10:56	9.152	.....	→	13	1356
11:10:47	11:10:55	8.960	.....	←	13	1927
11:10:55	11:10:57	1.984	.....	←	13	1938
11:10:55	11:10:58	2.176	.....	→	12	1152
11:11:01	11:11:01	0.512	.....	←	5	396
11:11:01	11:11:01	0.640	.....	→	6	516
11:11:01	11:11:03	1.536	.....	→	12	1152
11:11:01	11:11:03	1.408	.....	←	12	1863

Source: SSHCure: A Flow-Based SSH Intrusion Detection System
by Laurens Hellemons, Luuk Hendriks, Rick Hofstede, Anna Sperotto, Ramin Sadre and Aiko Pras

Example: (D)DoS attacks

- (Distributed) denial of service
  - Aim is to reduce offered services
- Typically by overloading targets
- Examples
  - UDP flooding
  - TCP SYN flooding
- Very easy to perform

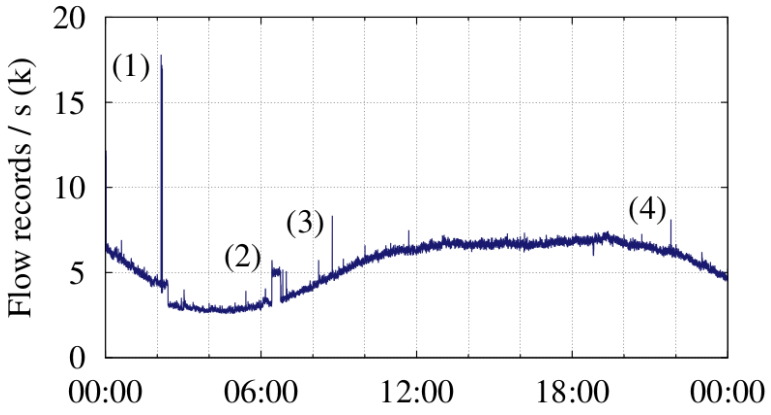


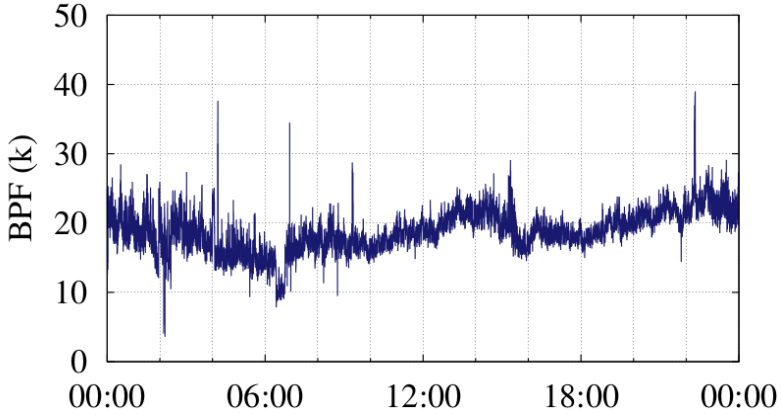
DDoS detection and mitigation

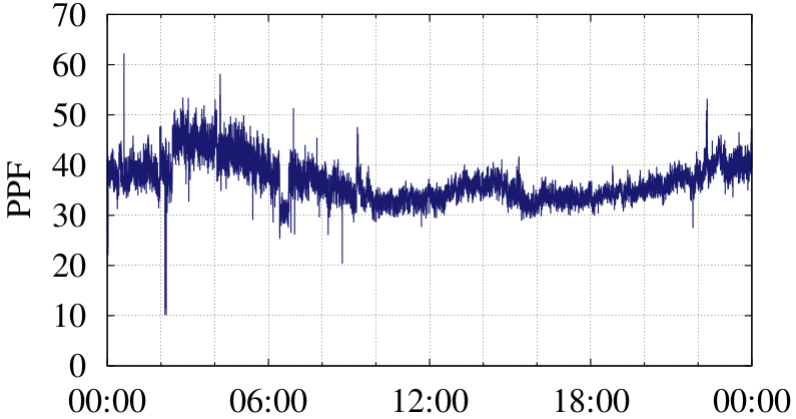
- DDoS attacks result in many different flows
- Potential problems
  - Flow collector might overload
  - Delay introduced by flow metering and collection process
- Move detection closer to the source
  - Quick detection and response
- How can we detect a DoS attack?
  - Sudden increase in network traffic
  - Also occurs at the beginning of a working day...

Traffic measures

- Flow records creations per second
- Average number of bytes per flow
- Average number of packets per flow
- Average flow duration





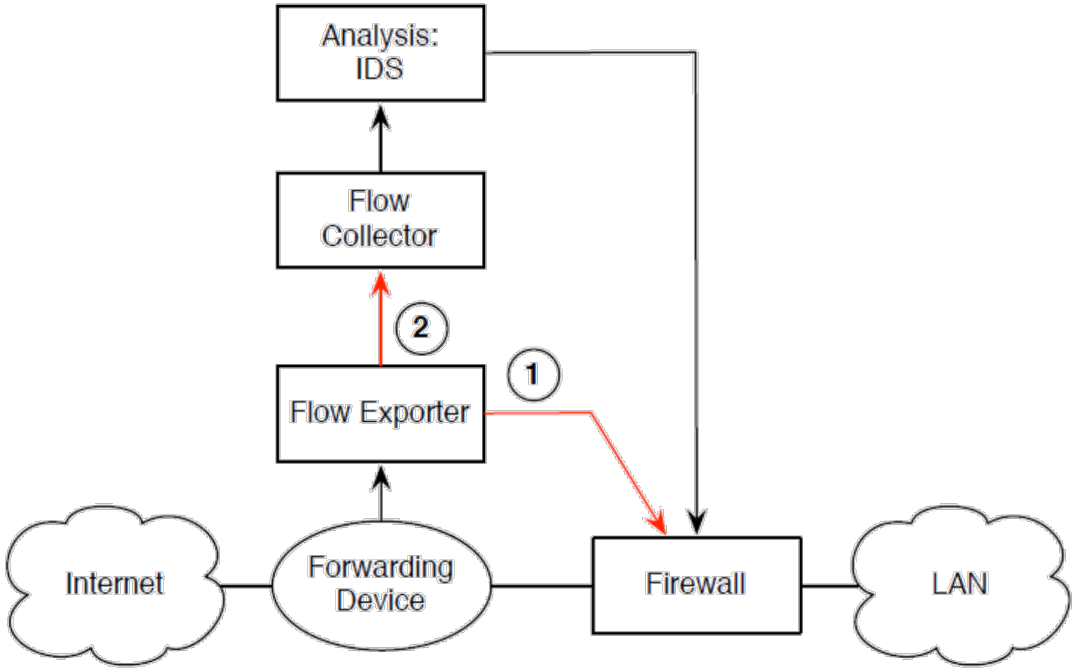


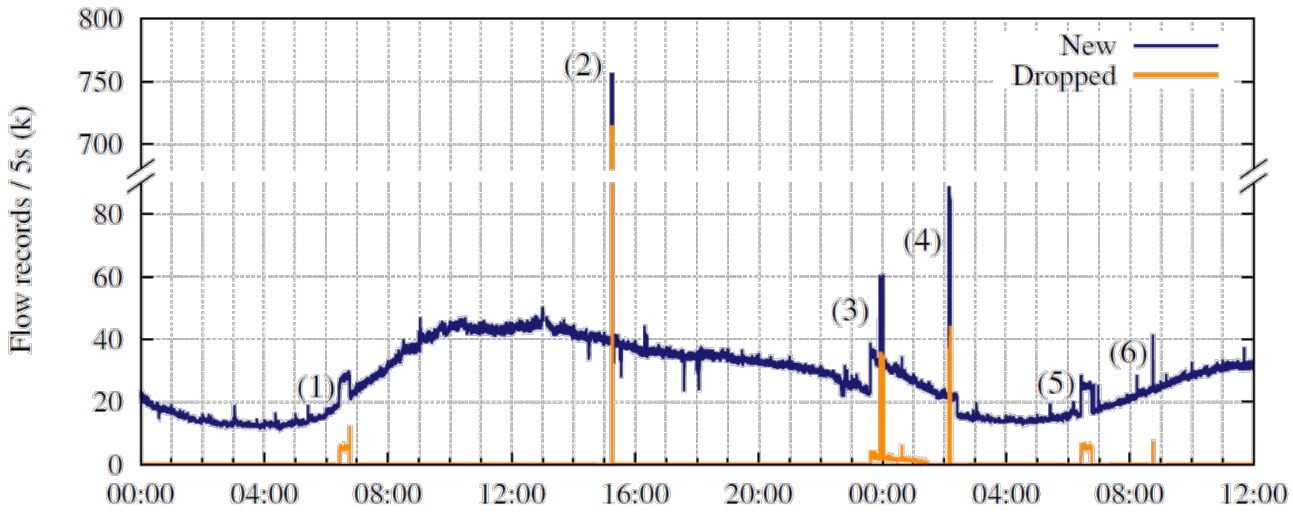
Measurements on CESNET network

Source: Towards Real-Time Intrusion Detection for NetFlow and IPFIX, by Rick Hofstede, Václav Bartoš, Anna Sperotto, Aiko Pras

DDoS detection and mitigation

- DDoS detection in flow metering process
  - Identify attack by counting number of flows per source IP address
  - DDoS attack if many flows per second (≥ 200) from same IP address that contain only few packets
- DDoS mitigation by adding IP address to blacklist
  - Add rules to firewall to block traffic from blacklisted IP addresses (1)
  - Filter flows from blacklisted IP addresses to reduce stream of flow records (2)
- Once attack is over, remove filters again
- What about IP spoofing?
  - Blacklist destination IP addresses

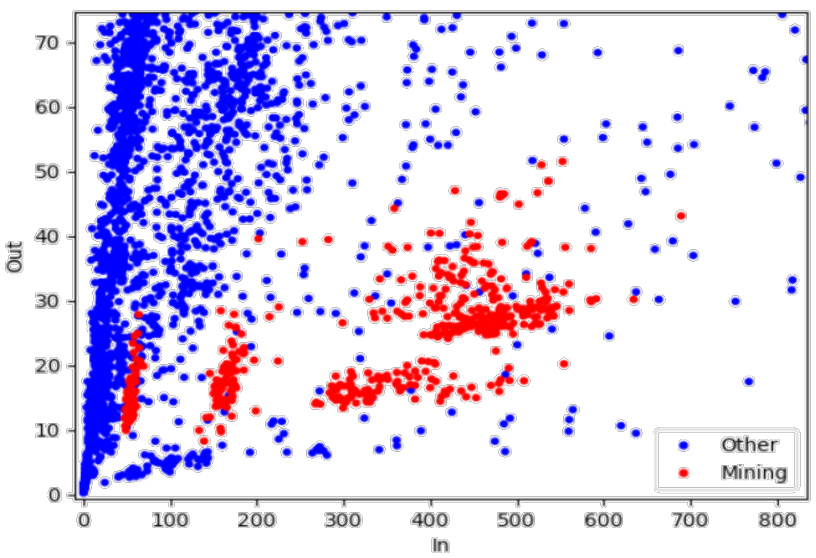




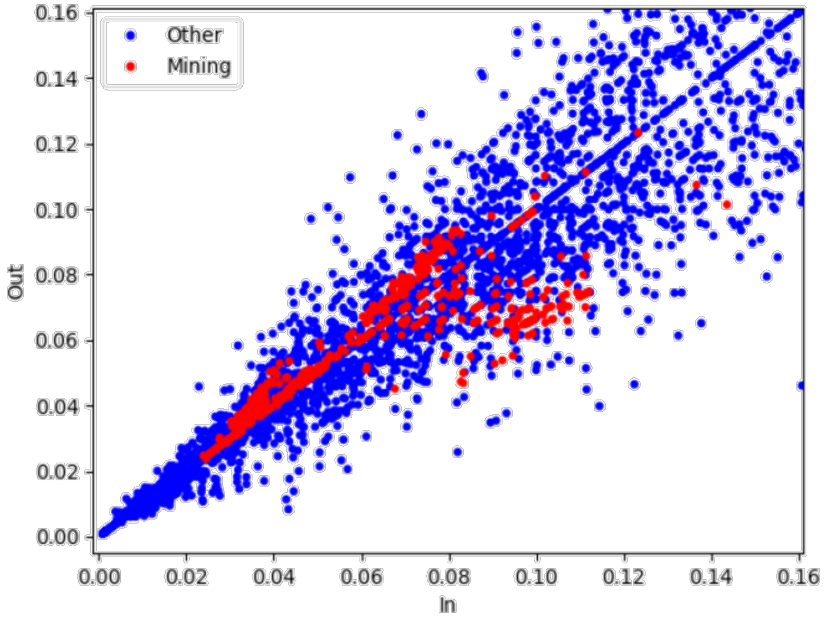
Source: Towards Real-Time Intrusion Detection for NetFlow and IPFIX, by Rick Hofstede, Václav Bartoš, Anna Sperotto, Aiko Pras

Detecting cryptocurrency miners

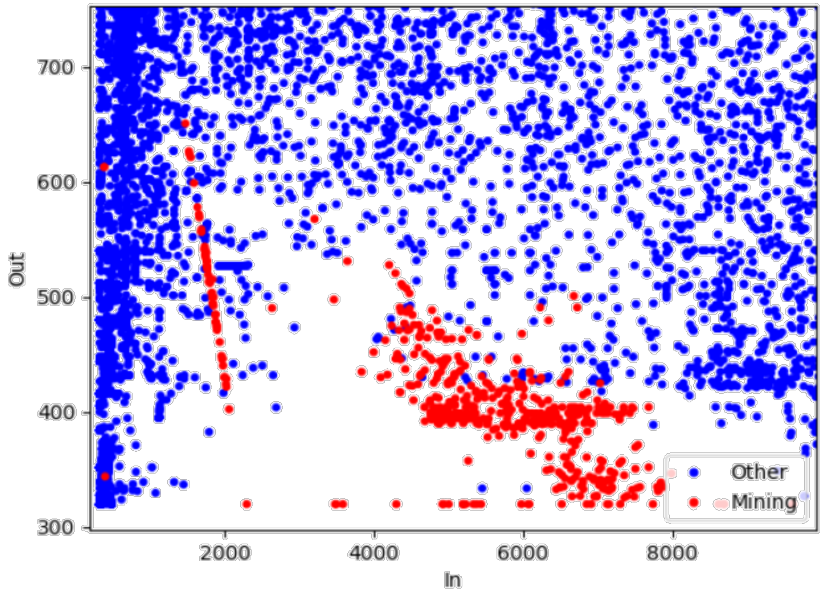
- Flows related to Stratum-protocol (between mining clients and servers in mining pools)



a) Bits/second



b) Packets/second



c) Bits/packet

Source: Detecting cryptocurrency miners with NetFlow/IPFIX network measurements, by Z. Muñoz, J. Suárez-Varela and P. Barlet-Ros, 2019 IEEE International Symposium on Measurements & Networking (M&N), 2019, pp. 1-6

Further reading

Read the following paper (also for the exam):

- *Flow Monitoring Explained: From Packet Capture to Data Analysis With NetFlow and IPFIX*
  Rick Hofstede, Pavel Čeleda, Brian Trammell, Idilio Drago, Ramin Sadre, Anna Sperotto, Aiko Pras
  IEEE Communications Surveys & Tutorials, Vol. 16, Issue 4, Fourthquarter 2014, p. 2037-2064.
- Note: read the following sections (the other sections are optional):
  - I. Introduction (up at A. Objective)
  - III. Flow monitoring architecture
  - IV. Packet observation
  - V. Flow metering & export (up to E. IPFIX Messages)
  - VII. Data analysis

Optional reading

On real-time intrusion detection and DDoS attack detection

- *Towards real-time intrusion detection for NetFlow and IPFIX*
  Rick Hofstede, Vaclav Bartos, Anna Sperotto, Aiko Pras
  Proceedings 9th International Conference on Network and Service Management (CNSM), 2013

On SSH attack detection

- *SSH Compromise Detection using NetFlow/IPFIX*
  Rick Hofstede, Luuk Hendriks, Anna Sperotto, Aiko Pras
  ACM SIGCOMM Computer Communication Review archive, Volume 44, Issue 5, Oct. 2014, p. 20-26
- *Hidden Markov Model Modeling of SSH Brute-Force Attacks*
  Anna Sperotto, Ramin Sadre, Pieter-Tjerk de Boer, and Aiko Pras
  Lecture Notes in Computer Science, vol. 5841, 2009, Springer, p. 164-176

On SSHCure

- *SSHCure: A Flow-Based SSH Intrusion Detection System*
  Laurens Hellemons, Luuk Hendriks, Rick Hofstede, Anna Sperotto, Ramin Sadre and Aiko Pras
  Lecture Notes in Computer Science, vol. 7279, 2012, Springer, p. 86-97