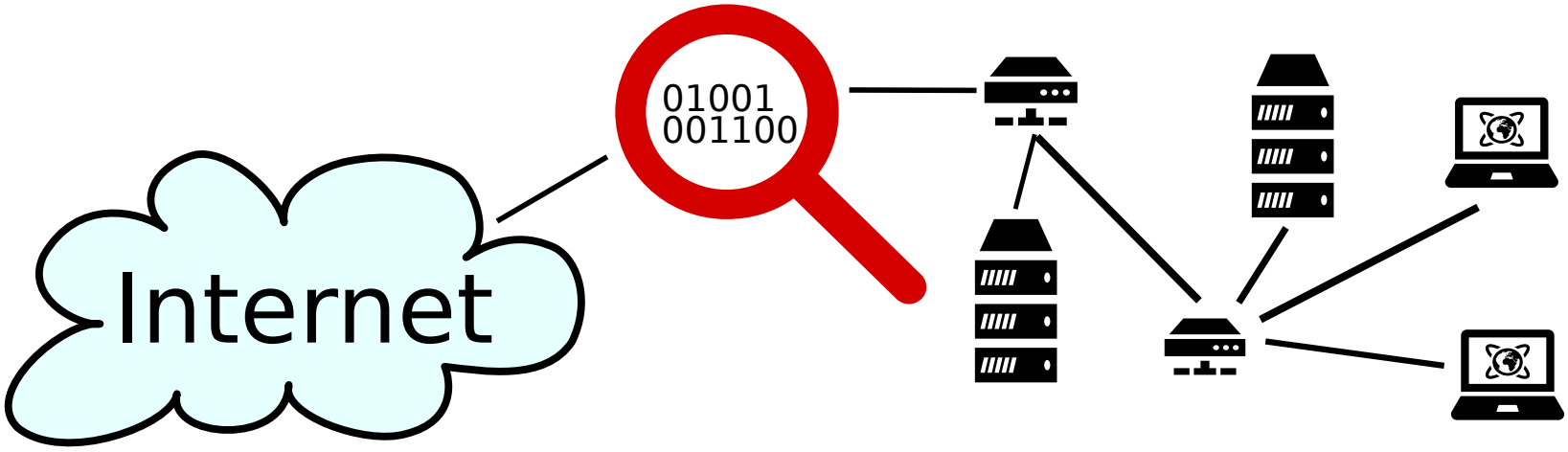
Intrusion Detection in High-Speed Networks: From Packets to Flows and Back

Falko Dressler

Network Monitoring

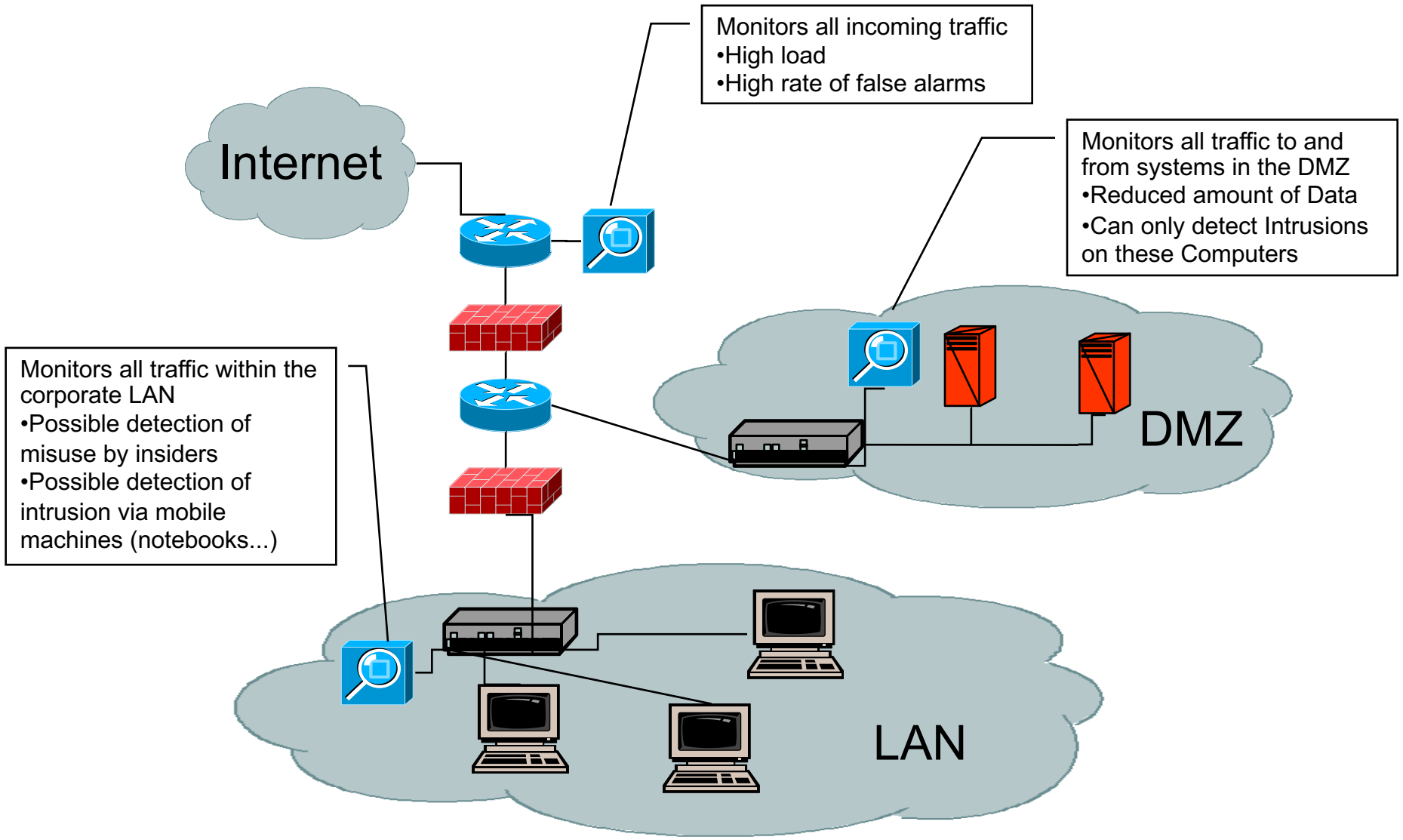
Network Monitoring

- Observing network traffic for:
  - Analysis (performance, statistics, troubleshooting, accounting)
  - Intrusion detection
  - Attack prevention
  - ...



Intrusion Detection

Placement of a Network Intrusion Detection System



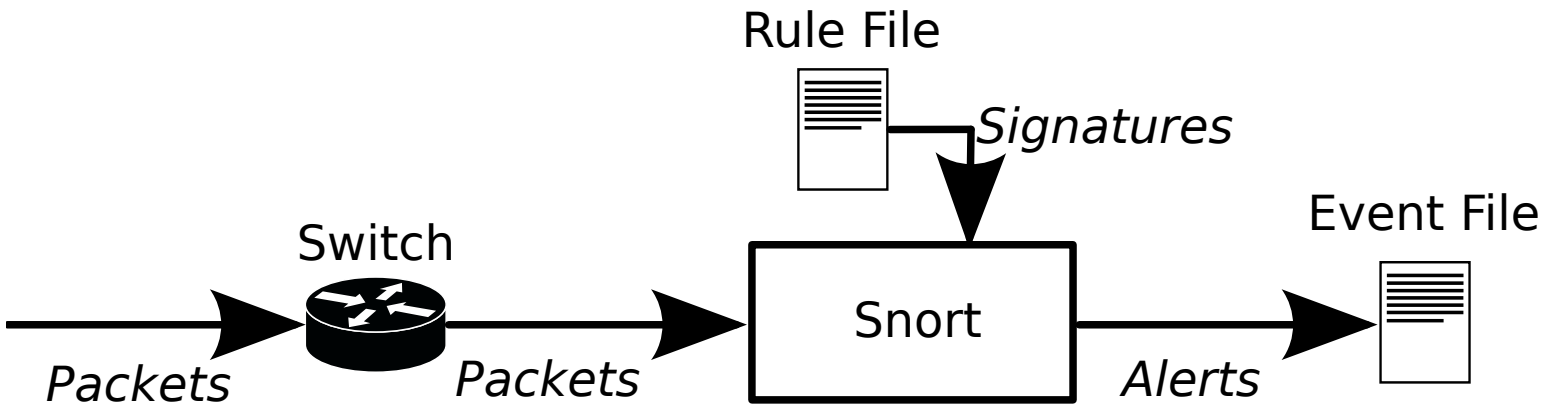
Network Intrusion Detection Systems (IDS)

- Analyze network traffic for malicious activity

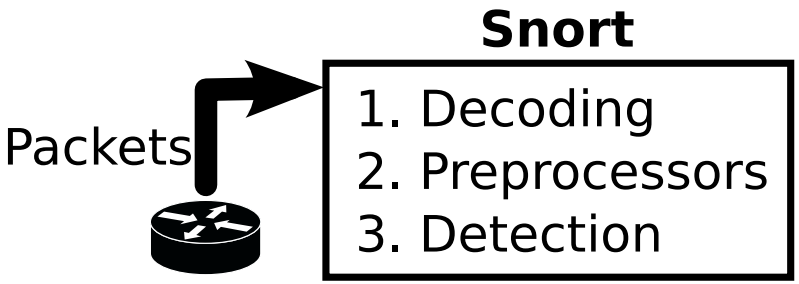
- Anomaly-based IDS
  - Have a model of ’normal’ traffic
  - Detect and alert deviations from ’normal’ traffic
  - + all sorts of attacks
  - − higher false positive rate

- Signature-based IDS
  - Have rule-set of known attacks and incidents
  - If packet/stream satisfies rules alarm is triggered
  - + low false positive rate
  - − no novel attacks
  - → Example: Snort

Signature-based IDS: Snort



- Signature analysis is very performance hungry:
  1. Decoding of packets
  2. Preprocessing data
  3. Detection phase



- **Snort with 5000 rules can handle 130k pkts/s**

Signature-based IDS: Snort

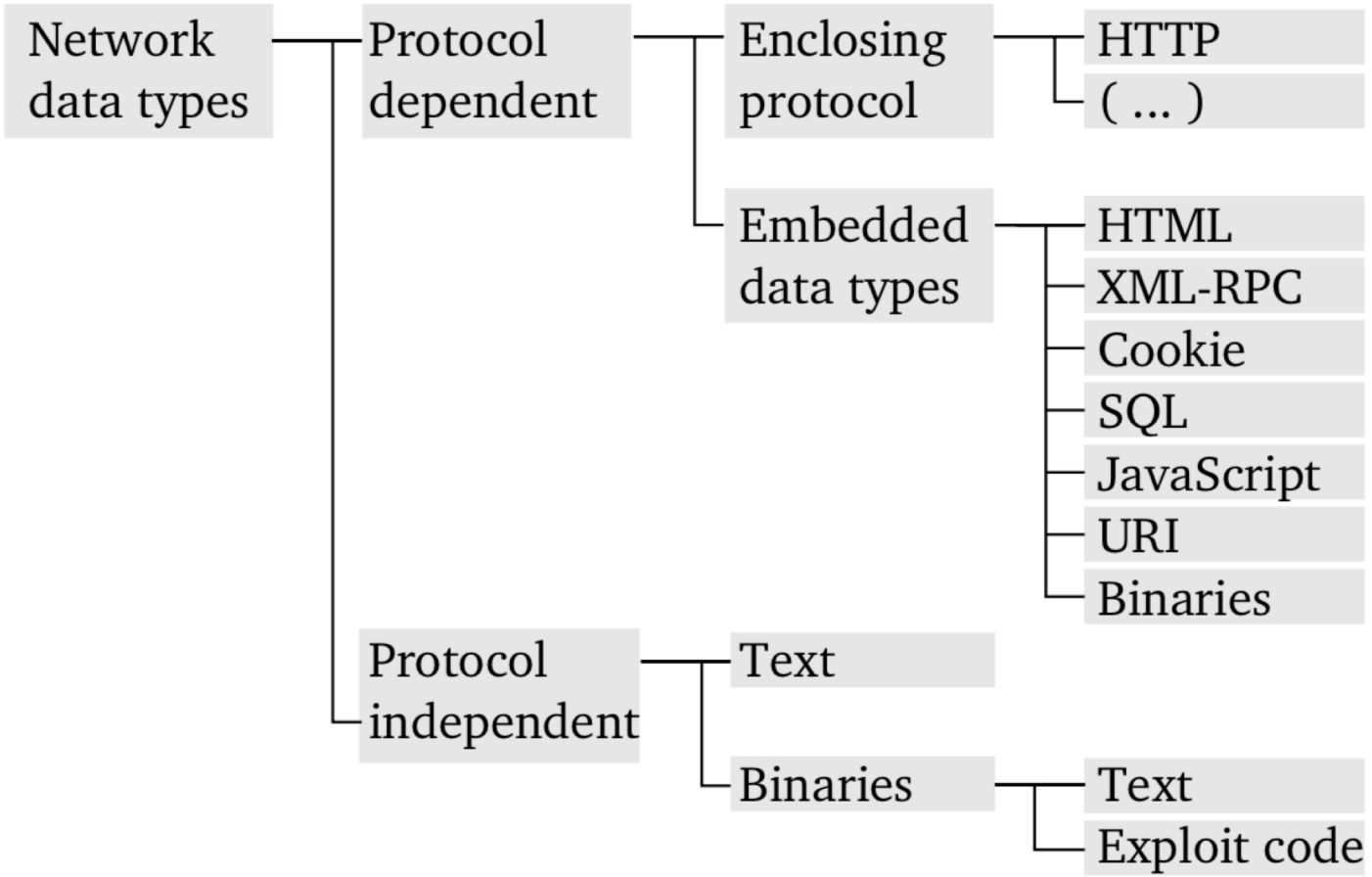
- Mainly signature based, each intrusion needs a predefined rule

```
alert tcp $HOME_NET any -> any 9996 \
    (msg:"Sasser ftp script to transfer up.exe"; \
    content:"|5F75702E657865|"; depth:250; flags:A+; \
    classtype: misc-activity; sid:1000000; rev:3)
```

- Three step processing of captured information (capturing is done by libpcap):
  - Preprocessing (normalized and reassembled packets)
  - Detection Engine works on the data and decides what action should be taken
  - Action is taken (log, alert, pass)

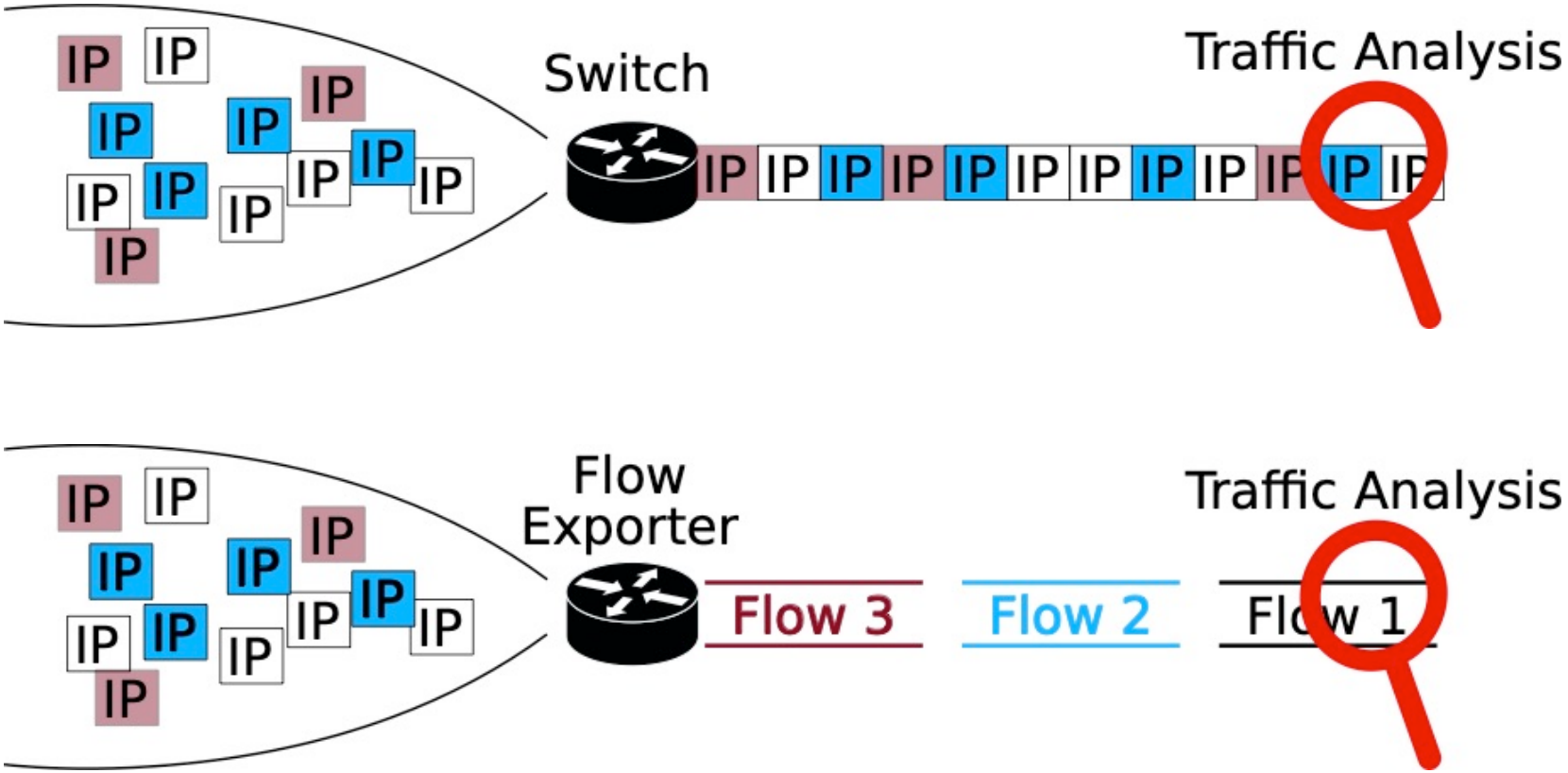
Signature-based IDS: Snort

- Rule processing



From Packets to Flows

From Packets to Flows



Flow-based Traffic Analysis

- Flows are “condensed” network traffic data
- Packets with same properties go into same Flow
- IPFIX supported by most industry grade switches
- Flow fields are configurable

IPFIX: IP Flow Information Export

- Example IPFIX Flow Record:

```
+--- Ipfix Data Record (id=999)
'- sourceIPv4Address                :10.0.2.15
'- destinationIPv4Address           :93.184.216.34
'- sourceTransportPort              :50488
'- destinationTransportPort         :80
'- packetTotalCount                 :13
'- octetDeltaCount                  :2304
+---
```

Flow-based IDS

Signature-based Intrusion Detection on IPFIX Flows

- Snort rule:

```
alert tcp any any -> any any
(msg:"Example Alert";
content:"GET"; http_method;
content:"/evil.jpg"; http_uri;
sid:1234567; rev:0;)
```

- IPFIX flow:

```
+--- Ipfix Data Record (id=999)
'- sourceIPv4Address
'- destinationIPv4Address
'- sourceTransportPort
'- destinationTransportPort
'- packetTotalCount
'- octetDeltaCount
+---
```

No app. layer information in IPFIX flows

IPFIX and HTTP

- IPFIX Flow Record w/ HTTP Data:

```
+--- Ipfix Data Record (id=999)
'- sourceIPv4Address                :10.0.2.15
'- destinationIPv4Address           :93.1[illegible]34
'- sourceTransportPort              :50488
'- destinationTransportPort         :80
'- httpRequestMethod                :'GET'
'- httpRequestTarget                :'/images/logo.png'
'- httpMessageVersion               :'HTTP/1.0'
'- httpRequestHost                  :'example.com'
+---
```

IPFIX HTTP fields now standardized with IANA

Signature-based Intrusion Detection on IPFIX Flows

- Snort rule:

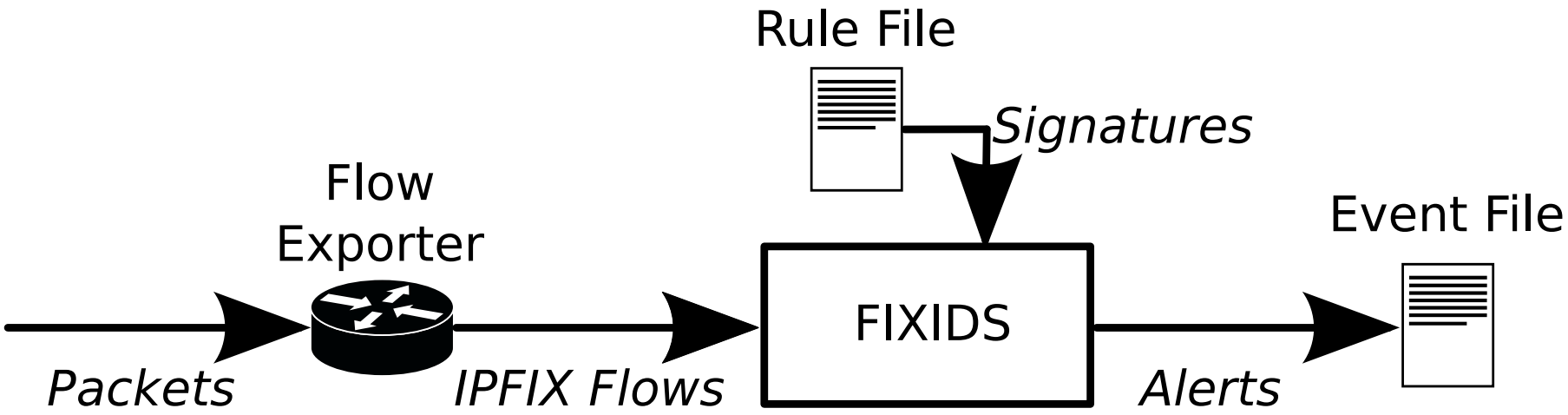
```
alert tcp any any -> any any
(msg:"Example Alert";
content:"GET"; http_method;
content:"/evil.jpg"; http_uri;
sid:1234567; rev:0;)
```

- IPFIX flow:

```
+--- Ipfix Data Record (id=999)
+- ...
'- sourceIPv4Address
'- httpRequestMethod    :'GET'
'- httpRequestTarget    :’/evil.jpg'
'- httpMessageVersion   :'HTTP/1.1'
'- httpRequestHost      :bad.com’
+---
```

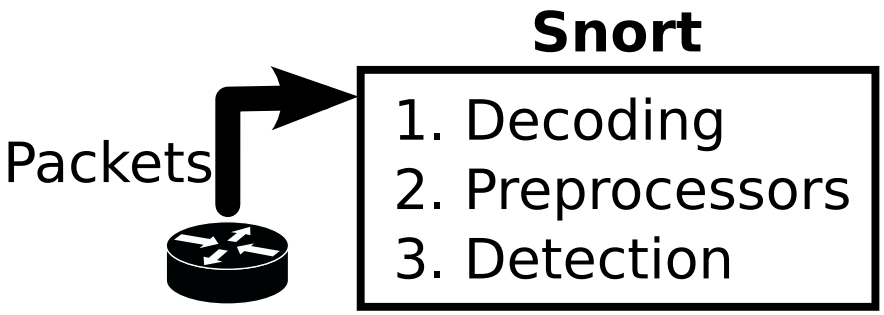
FIXIDS

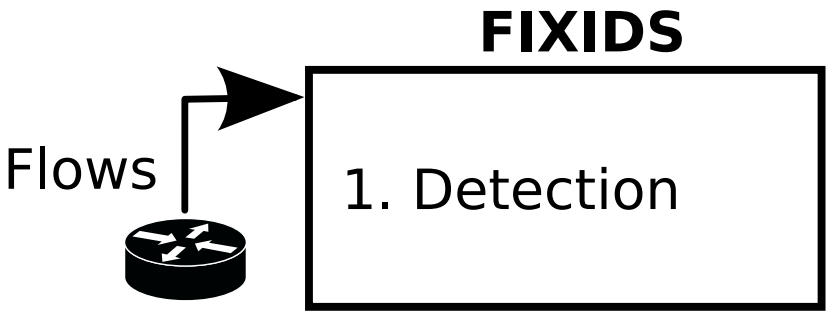
- IPFIX-based signature-based intrusion detection system
  - Signature-based Intrusion Detection (using Snort signatures)
  - on IPFIX flows (using standardized HTTP IPFIX fields)



Traditional Signature IDS (Snort) vs. FIXIDS

- Snort receives packets from a switch
- FIXIDS receives IPFIX flows from a flow exporting device (e.g., switch)



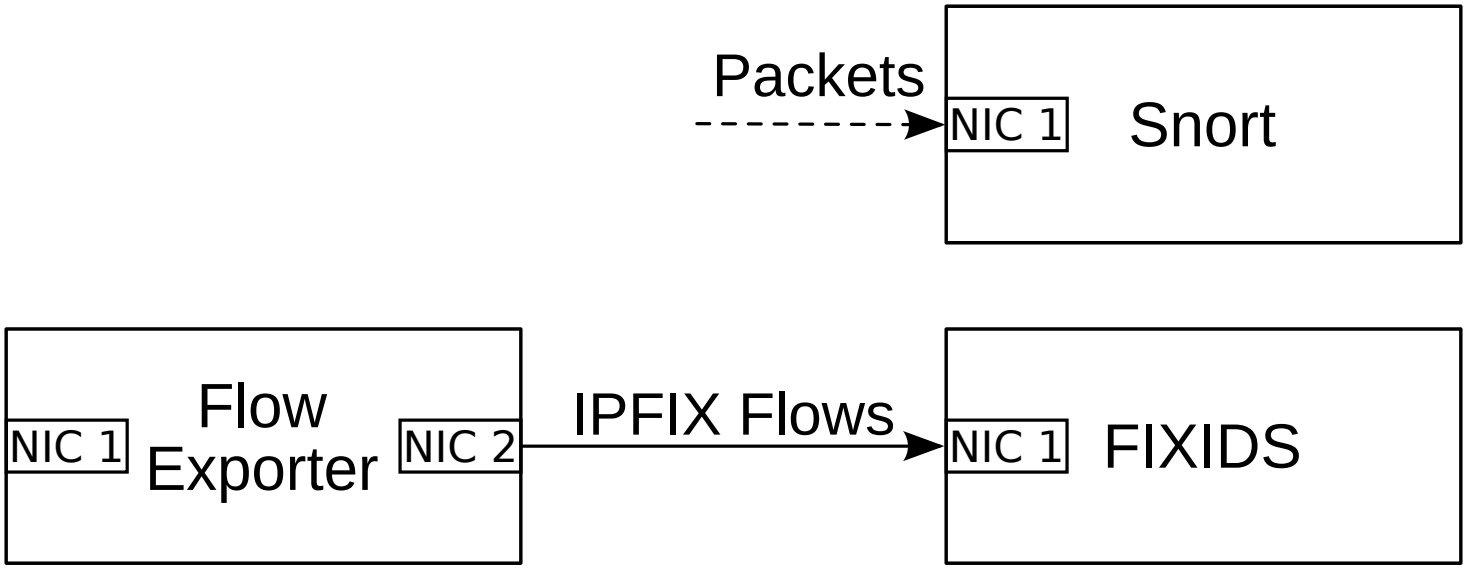


FIXIDS has to handle less than 0.5% of the data volume of Snort

Performance

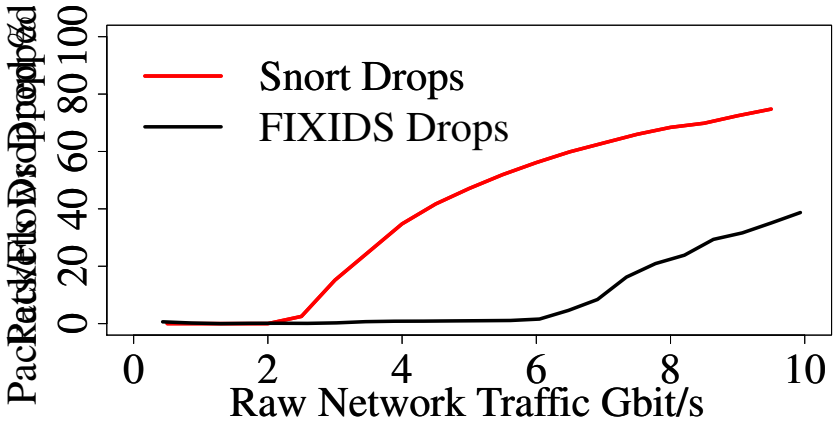
Evaluation: Experiment Setup

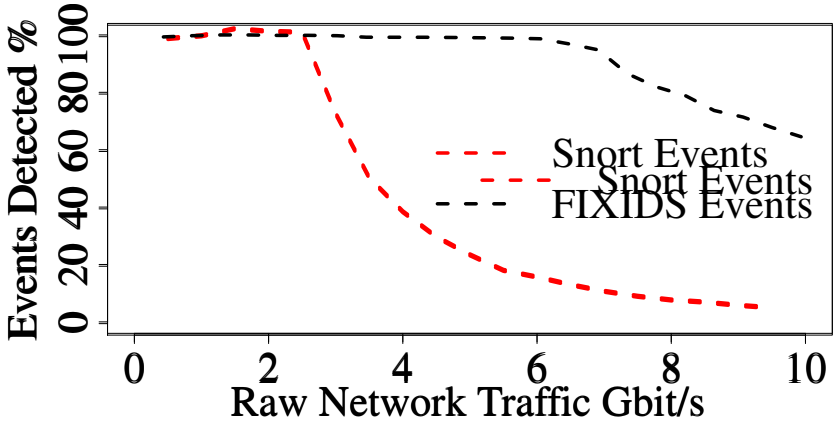
- Compare results of Snort and FIXIDS analyzing the same traffic, using the same signatures
- Replayed with increasing speed



Evaluation: Results

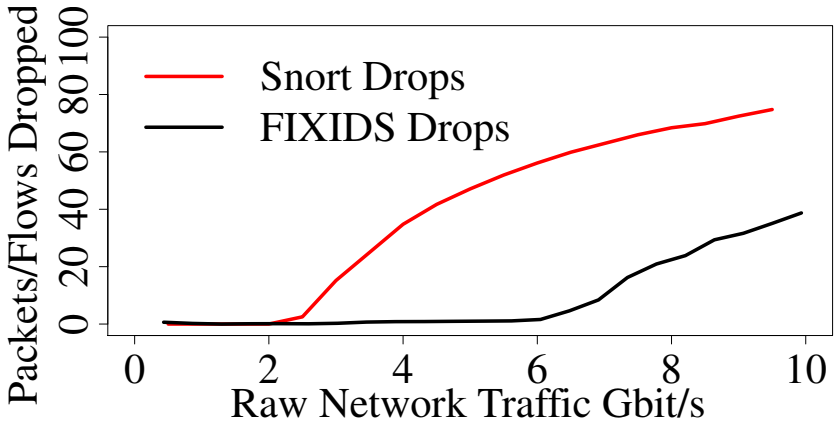
- Snort vs. FIXIDS: Same traffic, same signatures
- Replayed with increasing speed

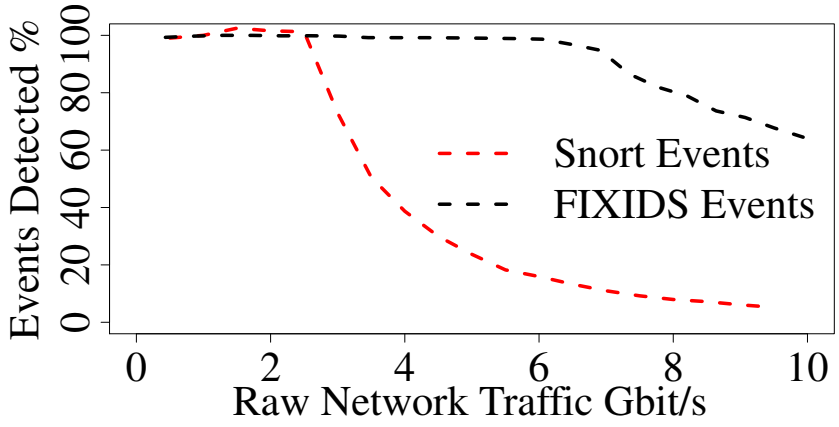




Evaluation: Results

- Snort:
  - 2 Gbit/s (136k Pkts/s): 0% Drops
  - 9.5 Gbit/s: >70% Drops
- FIXIDS:
  - 6 Gbit/s (14000 flows/s): 0% Drops
  - 9.5 Gbit/s (22000 flows/s): 40% Drops





Testing IDS

How to test a NIDS?

- Real traffic?
  - hard to get
  - public traces: old, no payload
  - contains only very few attacks
- Manually creating attack traffic?
  - time intensive
  - cumbersome
- In general, traces do not contain enough **unique** attacks

GENESIDS

- Generating Events for Signature-based Intrusion Detection Systems

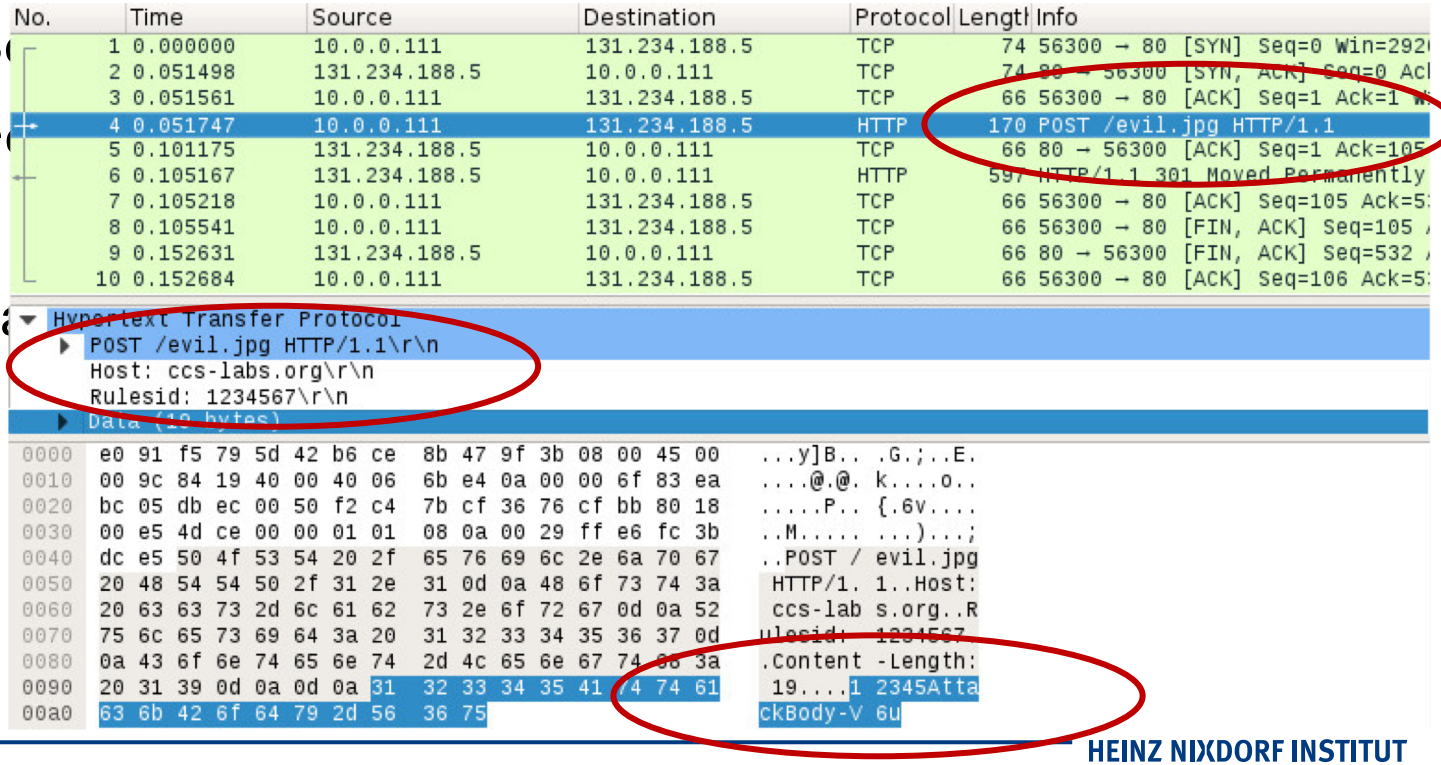
- INPUT: Set of attack descriptions
  - Snort syntax
  - HTTP attacks

- OUTPUT: Stateful network traffic containing attack patterns
  - One flow per attack
  - Annotated with an attack ID

Rule example

```
alert tcp any any -> any any (
msg:"This is an example rule";
content:"POST"; http_method;
uricontent:"|2F|evil.jpg";
pcre:"/AttackB
sid:1234567; r
```

- genesids -f ex



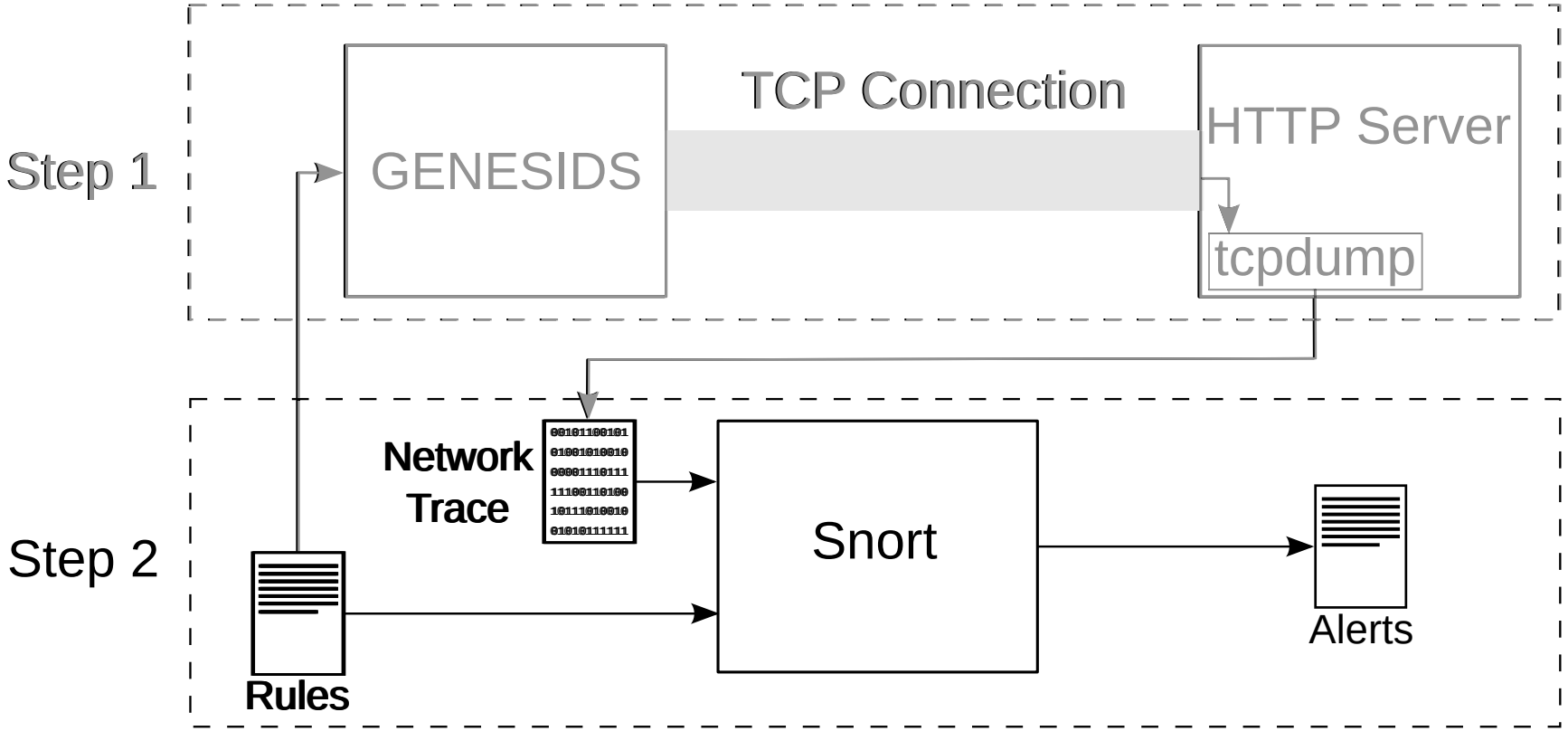
GENESIDS Evaluation: Goals & Rules

- Ability to generate a variety of different attacks
- Generated attacks trigger expected event

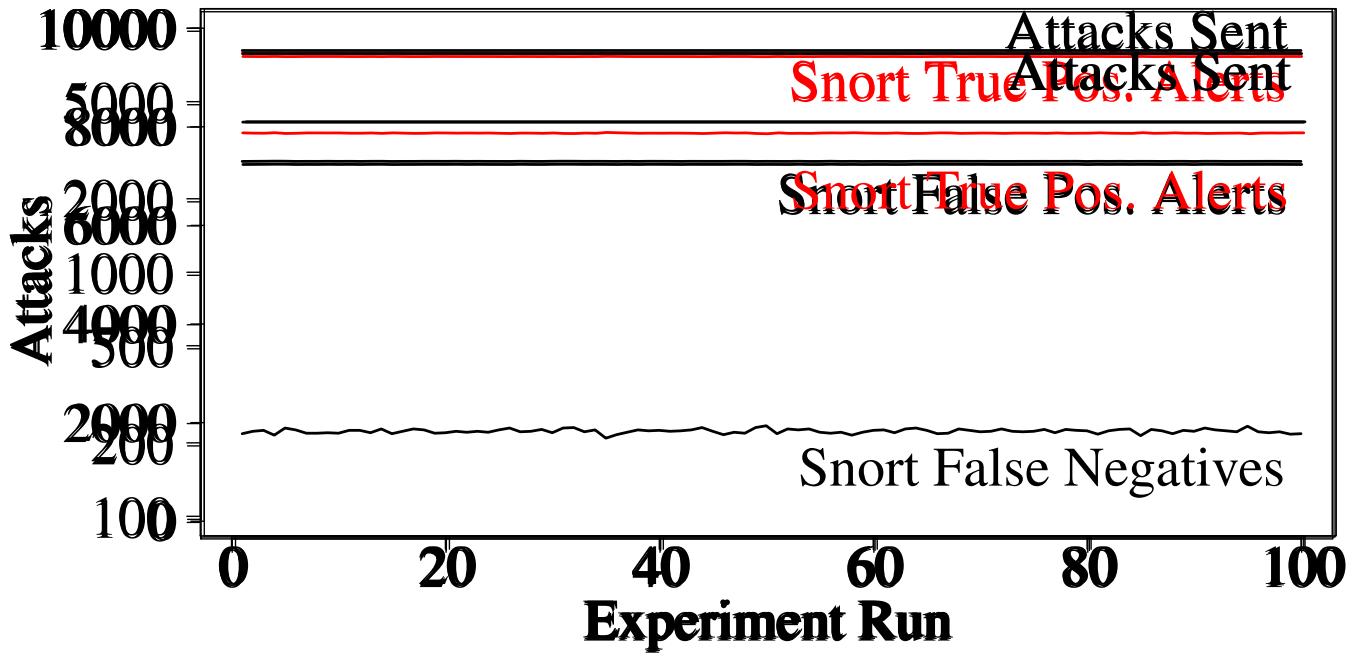
- All supported Snort rules from:
  - Snort.org subscriber rule-set
  - Snort.org community rule-set
  - Emerging Threats rule-set

- TOTAL 8101 different rules

GENESIDS Evaluation Steps



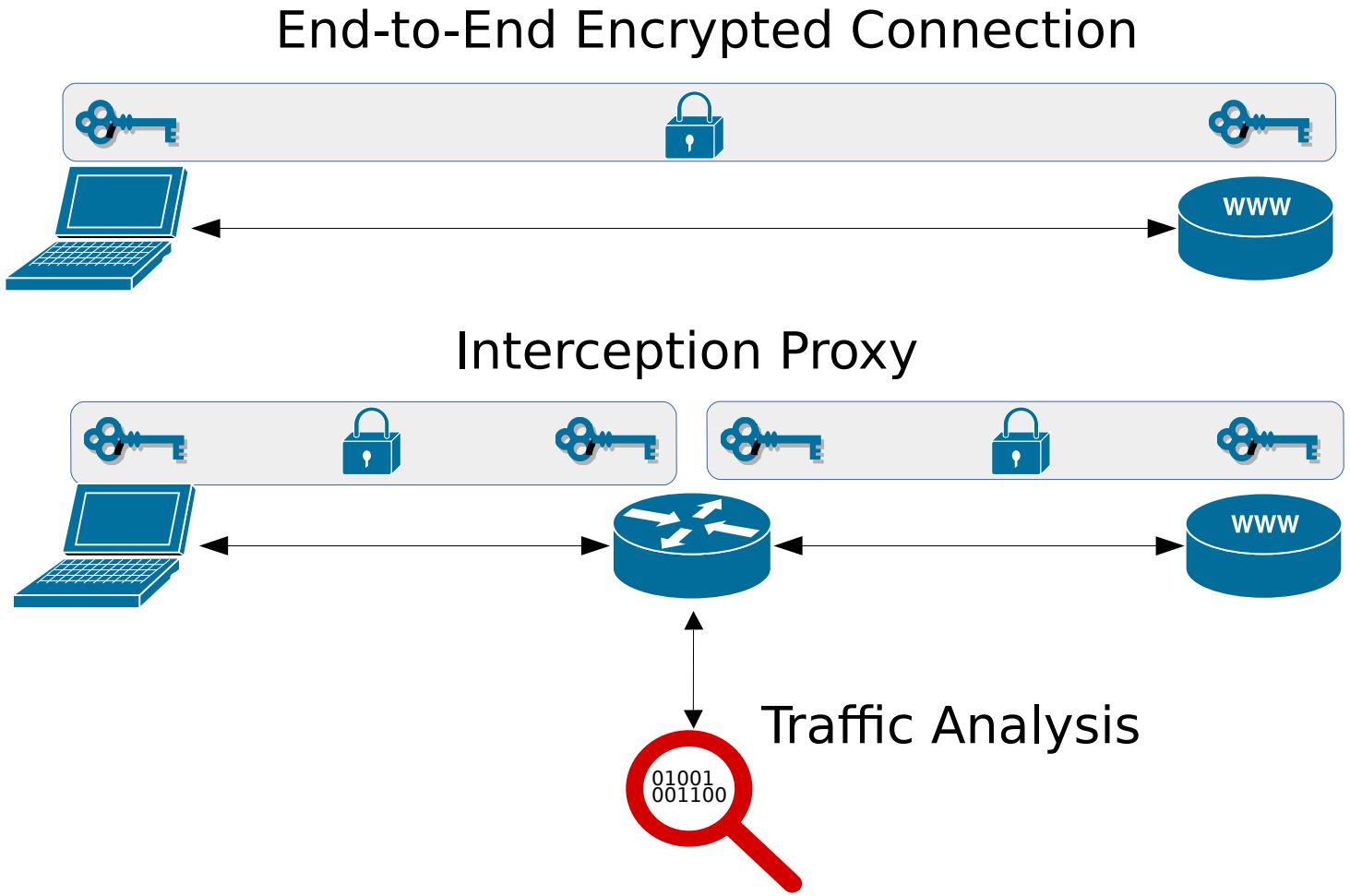
Evaluation Results: Generated Attacks



- GENESIDS: 8101 attacks generated (out of 8101 rules)
- Snort: 7877 (avg) true positive alerts triggered (out of 8101)
- Snort: 2847 (avg) false positive alerts triggered (62% triggered by 3 rules)
- Snort: 223 (avg) false negatives (generated attacks that did not trigger the corresponding alert)

Open Challenges

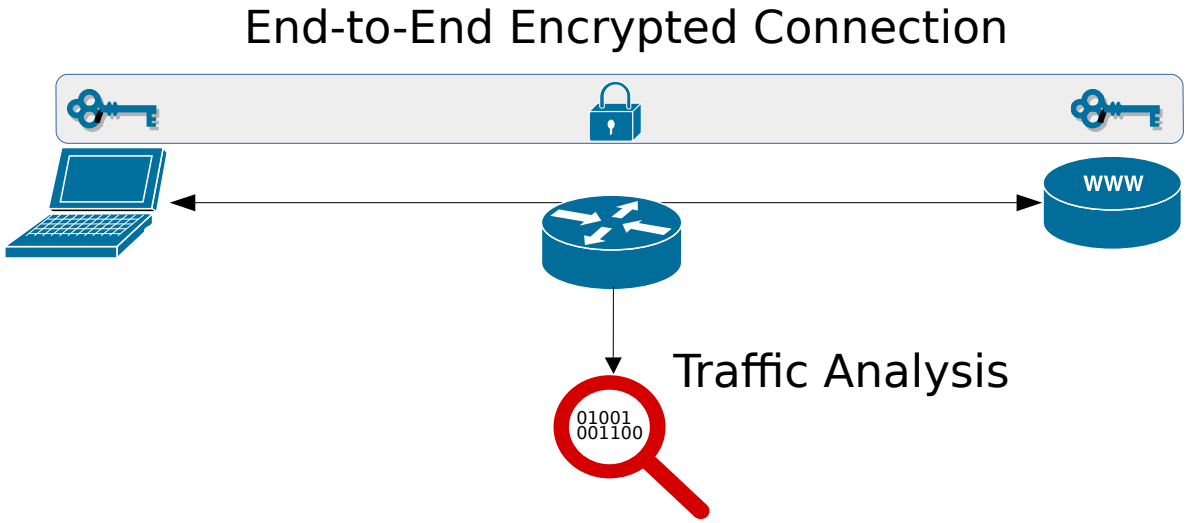
Network Monitoring on Encrypted Traffic



Very performance intensive

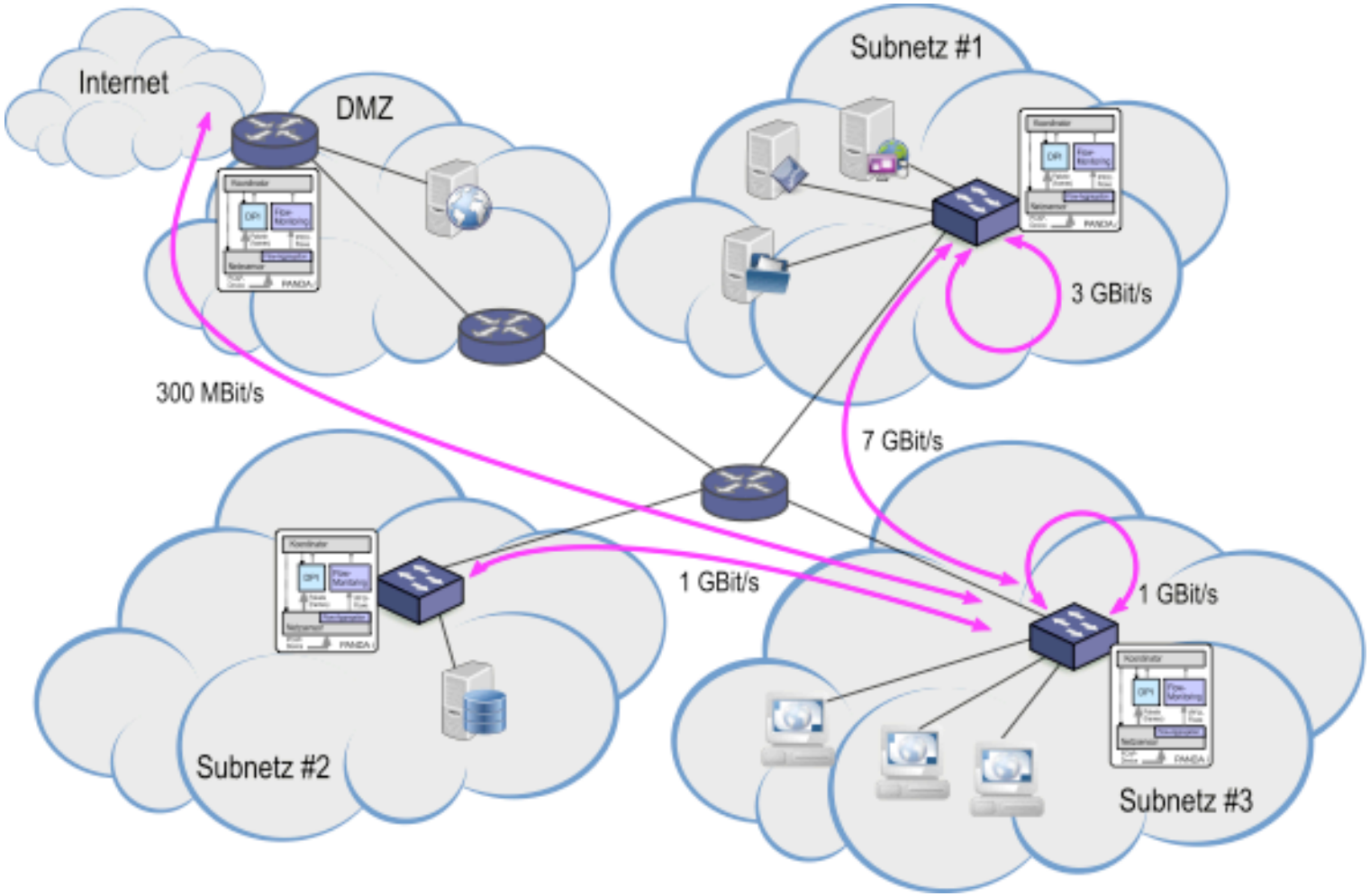
Passive Monitoring on Encrypted Traffic

- Using statistical properties and machine learning
- Only categorization possible (e.g., application)



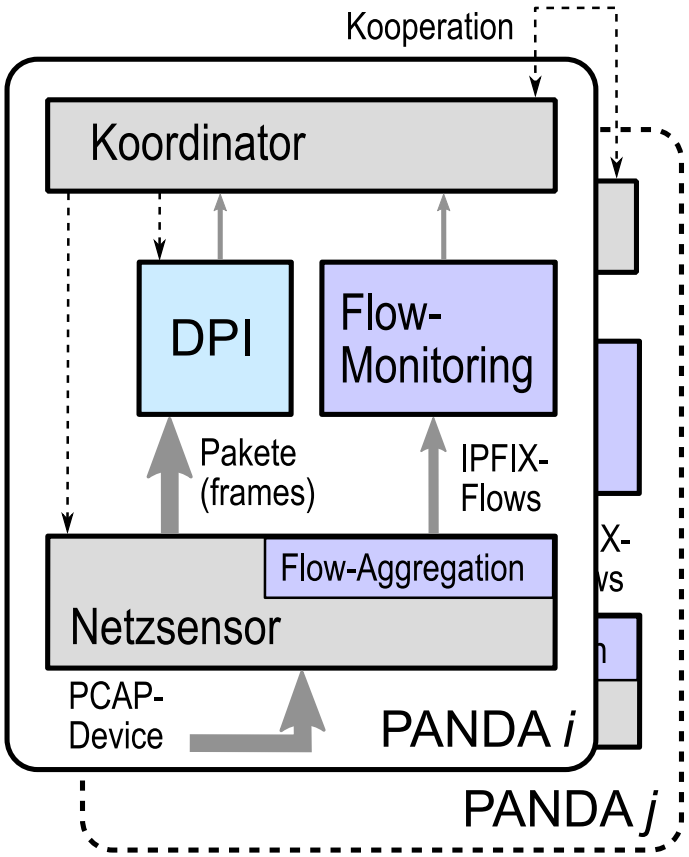
- How to passively detect malware (communication) in encrypted traffic?
- More general: How to foster strong encryption without sacrificing intrusion detection accuracy?

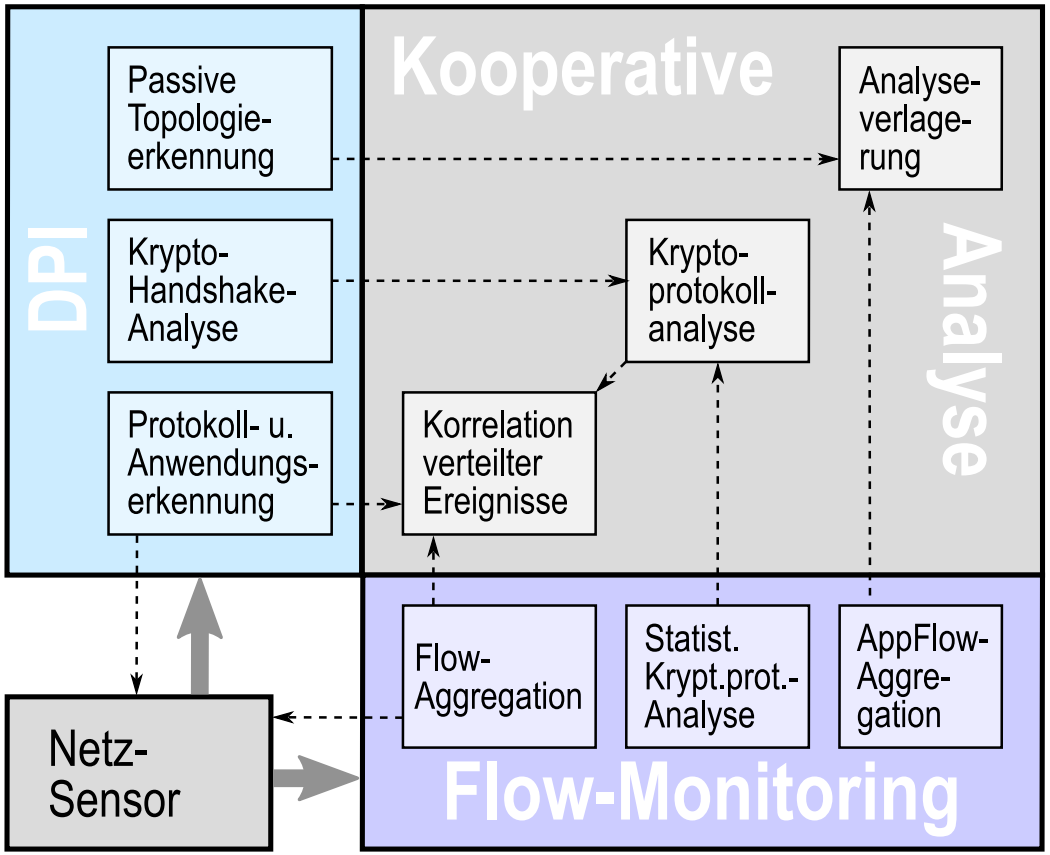
Where to Look for Attacks



DFG Project PANDA

PANDA – Precise Attack Detection for Network Domains by Application Classification





Conclusion

Conclusion

- Network Monitoring and IDS
  - Fundamental parts of every modern security solution
  - Flexible packet-based analysis is just too slow
- Flow-based approaches
  - Now standardized by IETF and IANA
  - FIXIDS builds directly upon this
- Unsolved so far
  - Evaluation of encrypted traffic
  - Optimal placement of probes in larger networks
- **… as can be seen, there are many open challenges and questions for another decade of interesting research ☺**

We are hiring! PhD positions available in Paderborn