

9. Network Intrusion Detection Systems

Simona Buchovecká, Tomáš Čejka

Faculty of Information Technology, CTU in Prague

simona.buchovecka@fit.cvut.cz, tomas.cejka@fit.cvut.cz

April 19, 2021

Section 1

Introduction

IDS Intrusion Detection System

- focused on suspicious/malicious traffic detection
- host-based / network-based
- signature-based (recognition of bad patterns, malware)

ADS Anomaly Detection System

- statistical, machine learning, or other approach to detection
- anomalous traffic — deviation from normal, usually observed
- detection of known or unknown traffic
- “behavioral analysis”

IPS Intrusion Prevention System

- identify malicious activity, log information, report it, try to block/stop it
- packet discarding (/blackholing), connection resetting

Section 2

Particular Detection Systems

Existing Tools (mainly IDS)

Packet-based

- Snort
- Zeek (formerly Bro)
- Suricata
- ...

Flow-based

- Flowmon ADS / DDoS Defender
- Stream4Flow
- ntopng
- NfSen (batch processing)
- Analysis Pipeline (SiLK)
- NEMEA
- ...

Offline Processing

- Elasticsearch + Kibana
elastalert (<https://github.com/Yelp/elastalert>)
elastiflow (<https://github.com/robcowart/elastiflow>)
- Python + Pandas, matplotlib
- ...

IDS/IPS - Intrusion Detection / Prevention System

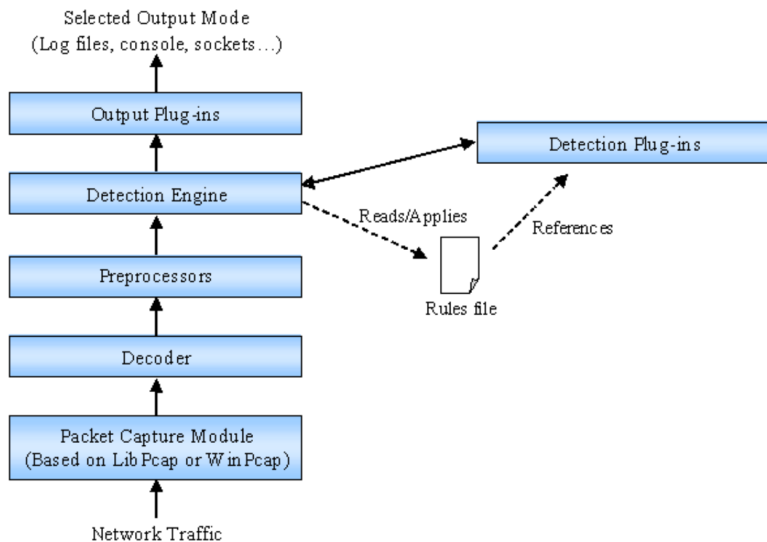
- Statefulness: Rule-based detection with thresholds to track the number of times a rule was triggered

Open source, developed by Cisco Systems (formerly SourceFire)

Combines:

- Signature, protocol, and anomaly-based inspection
- But usually it's packet-based
- Sniffer, packet logger, intrusion detection

Snort Structure & Plugins



Snort Components #1

Packet Decoder

- Takes packets from various interfaces

Preprocessors

- Arrange or modify packets
- E.g., convert unicode or hex characters in URL to text
- Reassemble fragmented IP packets
- Reassemble TCP segments
- Check for anomalies in packet headers (and issue alerts)
- Port scan processor

Snort Components #2

Detection Engine

- Detects intrusions in packets
- The detection engine speed is critical
- Uses Snort rules:
 - Rules arranged in a chain
 - If a rule is matched, a defined action is taken
 - Otherwise, the packet is dropped
 - *Actions*: Logging packets, issuing alerts, ...

General principles:

- The first rule that applies generates action/alert
- The highest priority applicable rule generates action/alert

Snort Components #3

Rules are applied to:

- IP Header
- Transport Layer Header (TCP, UDP, ICMP)
- Application layer header
- Packet payload (legal issues)

Detection engine performance depends on:

- Machine power
- Internal bus speed
- Network load
- Number of rules

Snort Components #4

Logging and Alerting System

- Logs in `/var/log/snort` by default

Output Plugins

- Log to `/var/log/snort/alerts` or other file, or via syslog facility
- Log to a database (MySQL, Oracle)
- Send e-mails, show web-based alerts
- Send SNMP traps
- Generate XML output
- Modify configuration of routers or firewalls
- Send SMB messages to MS Windows machines

Snort Rules #1

Intro: <https://resources.infosecinstitute.com/snort-rules-workshop-part-one/>

General form:

```
action proto src_ip src_port direction dst_ip  
dst_port (options)
```

Log 100 packet if ssh exploit is suspected

```
activate tcp any any -> 192.168.1.21 22  
(content:"/bin/sh"; activates:1; \  
msg:"Possible SSH buffer overflow"; )
```

```
dynamic tcp any any -> 192.168.1.21 22  
(activated_by:1; count:100;)
```

Snort Rules #2

Custom action:

```
ruletype redalert
{
    type alert
    output alert_syslog: LOG_AUTH LOG_ALERT
    output database: log, mysql, user=snort
    dbname=snort host=localhost
}
```

Snort Rules #3

Detect Nop:

```
alert tcp any any -> any any (msg:"Possible exploit"; \
    content:"|90|"; offset:40; depth:75; dsize: >6000;)
```

SYN & FIN Sent in one packet:

```
alert any any -> any any (flags: SF,12; \
    msg: "Possible SYN FIN scan";)
```

Note: Do not use escaped quotes

Basic information

- Transforms packets into events
- Events are processed using a script interpreter
- Turing complete Bro scripting language
- Zeek analyzers, in Zeek's event engine, perform application layer decoding, anomaly detection, signature matching, connection analysis

Resources:

<https://www.zeek.org/>

<https://en.wikipedia.org/wiki/Zeek>

Basic information

- IDS, IPS, Network Security Monitoring (NSM)
- TCP/IP engine (IPv6 support, tunnel decoding, tracking sessions, reassembling)
- protocol parsing (HTTP, SSL, TLS, SMB, ...)
- PCRE support
- Lua scripting

Rules format

action header options

Example:

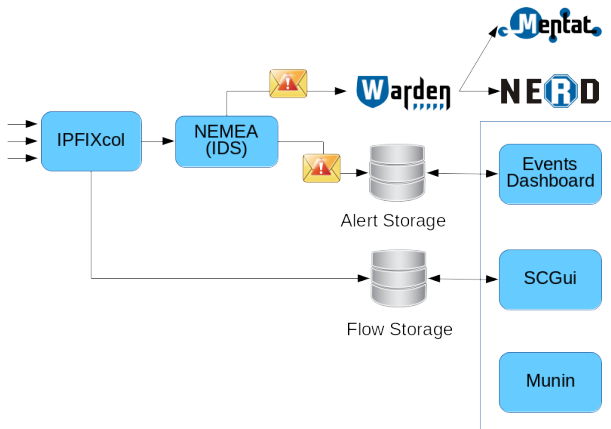
```
drop tcp $HOME_NET any -> $EXTERNAL_NET any
(msg:"ET TROJAN Likely Bot Nick in IRC (USA +..)";
flow:established,to_server; flowbits:isset,is_proto_irc;
content:"NICK "; pcre:"/NICK .*USA.*[0-9]{3,}/i";
reference:url,doc.emergingthreats.net/2008124;
classtype:trojan-activity; sid:2008124; rev:2;)
```



Resources:

<https://suricata-ids.org/>

- Modular, consisting of independent interconnected NEMEA modules
- Flow-based
- Stream-wise
- Application-aware (can work with L7-extend flow records)



NEMEA and other tools: screenshots of visualization (1/4)



NEMEA and other tools: screenshots of visualization (2/4)

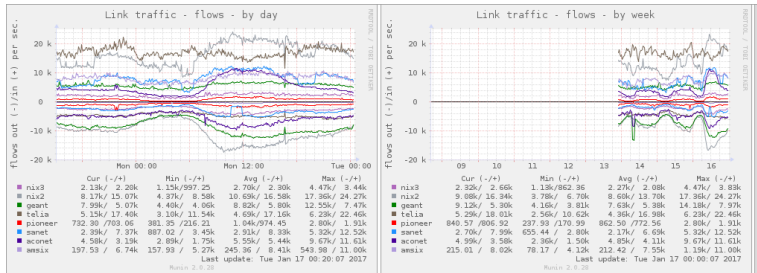
The screenshot displays the SecureCloud web interface. On the left is a dark sidebar with navigation links: Graph, Statistics, Database Query, Profiles, and User Control. The main area is titled 'Tab 1' and contains several sections:

- Sources:** A list with two items, 'ipset', each with a checked checkbox.
- Filter:** An empty text box with a 'Clear filter' button below it.
- Fast Options:** A panel with settings: 'Limit to: 20 records', 'Aggregate: srcip,dstip', 'Order by: flows', and 'Output: pretty'.
- Custom Options:** A panel with settings: 'add' (dropdown), 'direction' (dropdown), and 'No summary' (checkbox).
- Process request:** A button to execute the query.
- Query parameters:** A text box containing a command: `/usr/lib64/mpich/bin/mpxexec -n 2 /usr/lib64/mpich/bin/fdlatdump_mpic -f '*' -t 14038801000 -L 20 -s srcip,dstip -o flows --output-format=pretty --output-items=r,p`
- Query output:** A table with 10 columns: first, last, bytes, pkts, flows, srcip, dstip, duration, and bp. It contains 10 rows of data.

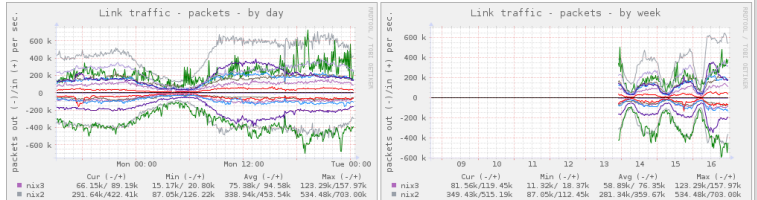
At the bottom left, a small box indicates 'Selected profile: 1410' and 'Translation: Jan 08 2017 14:10'.

first	last	bytes	pkts	flows	srcip	dstip	duration	bp
2017-01-08 13:05:37.120	2017-01-08 13:08:02.120	403.7 k	5.3 k	3.9 k	61.112.174.184	93.113.168.53	00:02:25.000	22
2017-01-08 13:05:40.120	2017-01-08 13:07:36.120	293.4 k	3.5 k	3.5 k	160.150.80.37	93.113.168.58	00:01:50.000	14
2017-01-08 13:05:10.372	2017-01-08 13:08:02.120	1.3 M	8.5 k	3.5 k	70.216.1.118	93.113.173.202	00:02:51.748	80
2017-01-08 13:05:15.877	2017-01-08 13:08:02.120	76.5 M	67.5 k	2.9 k	61.48.10.41	93.113.104.195	00:02:46.743	9
2017-01-08 13:05:32.120	2017-01-08 13:08:02.120	492.4 k	2.8 k	2.8 k	177.60.143.199	93.113.249.5	00:02:30.000	26
2017-01-08 13:05:09.201	2017-01-08 13:08:01.120	206.6 k	2.7 k	2.5 k	146.84.77.250	93.113.168.53	00:02:51.919	9
2017-01-08 13:05:37.120	2017-01-08 13:08:01.120	3.0 M	17.4 k	2.5 k	1500174af16a18:603:ffff:f100:ffff:dffe	dffa1476713f6a187a:61a7:b088:1c90:f1363d15	00:02:24.000	16
2017-01-08 13:05:36.249	2017-01-08 13:07:33.120	2.0 M	97.9 k	1.4 k	166.173.92.65	93.204.48.150	00:01:56.471	13

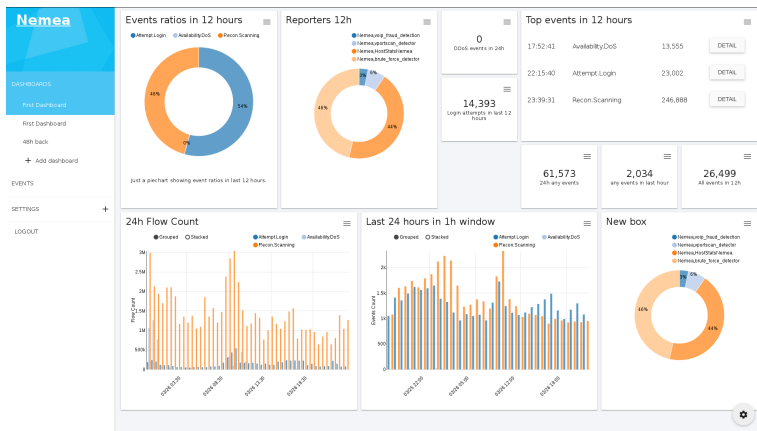
NEMEA and other tools: screenshots of visualization (3/4)



Link traffic - packets



NEMEA and other tools: screenshots of visualization (4/4)



- <http://nemea.liberouter.org/>
- <http://github.com/CESNET/LiST>
- <http://github.com/CESNET/SecurityCloudGUI>

FastNetMon

- Detects DDoS Attacks in 2 seconds
- Supports flow data, sFlow, port mirror/SPAN
- Supports BGP

```
FastNetMon v1.0
IPs ordered by: packets (use keys 'b'/'p'/'f' for change) and use 'q' for quit
Threshold is: 35000 pps and 1000 mbps total hosts: 13568

Incoming traffic      171015 pps    384 mbps  11973 flows
159.11.22.33          3309 pps    33.3 mbps   77 flows
159.11.22.33          3116 pps    34.8 mbps   2 flows
159.11.22.33          2567 pps    29.5 mbps   2 flows
159.11.22.33          2439 pps     1.8 mbps  76 flows
159.11.22.33          2364 pps     1.4 mbps  55 flows
159.11.22.33          2104 pps     1.5 mbps  19 flows
159.11.22.33          1938 pps     1.3 mbps  36 flows

Outgoing traffic      225121 pps   1905 mbps  17893 flows
159.11.22.33          3699 pps    39.9 mbps   83 flows
159.11.22.33          3557 pps    37.3 mbps  124 flows
159.11.22.33          2965 pps    32.8 mbps   98 flows
159.11.22.33          2645 pps    29.7 mbps   38 flows
159.11.22.33          2522 pps    26.1 mbps   65 flows
159.11.22.33          2474 pps    26.8 mbps   61 flows
159.11.22.33          2285 pps    18.9 mbps  194 flows

Internal traffic       0 pps      0 mbps

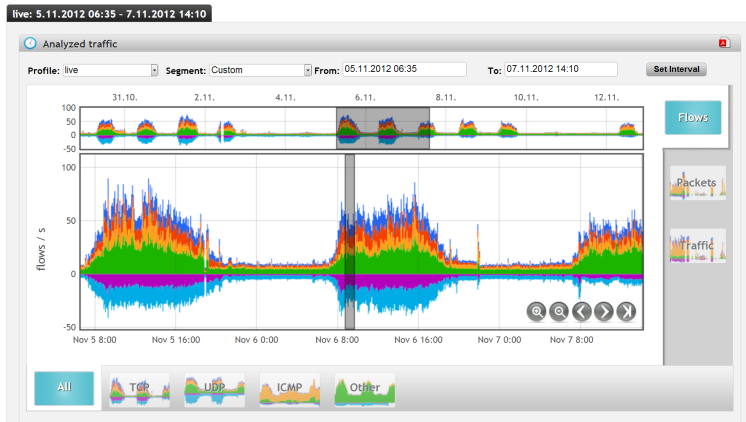
Other traffic          56 pps      0 mbps

Traffic calculated in: 0 sec 14670 microseconds
Packets received:     2308537
Packets dropped:       0
Packets dropped:      0.0 %
```

<https://fastnetmon.com>

Flowmon

Commercial



<https://www.flowmon.com/en/>

Section 3

Closing Words

There are many more. . .

Discussion?

What are Your experiences?

Questions?