Feedback experience with SNORT®

Christian Perez - Solange Gentil

CEA Cadarache

RMLL, Montpellier - July 2014

Definition

Snort®?

- Open-source NID(P)S project started in 1998 by Martin Roesch
- Now supported and developed by Sourcefire

Definition

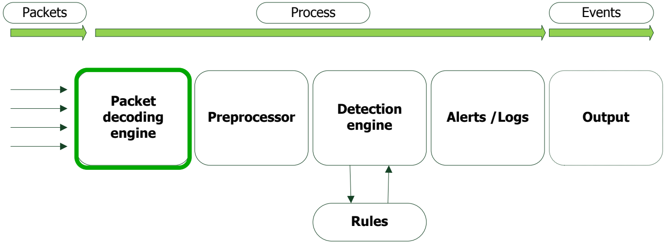
Snort®?

- Open-source NID(P)S project started in 1998 by Martin Roesch
- Now supported and developed by Sourcefire

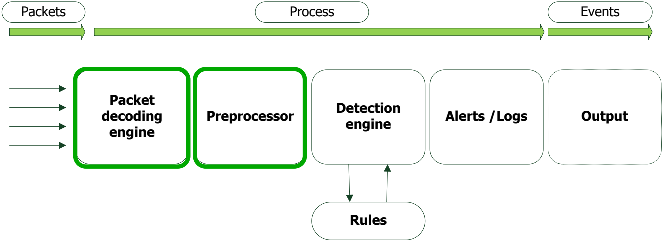
ID(P)S?

- Monitor network traffic
- Perform protocol analysis and content searching/matching
- Generate alerts based on signatures

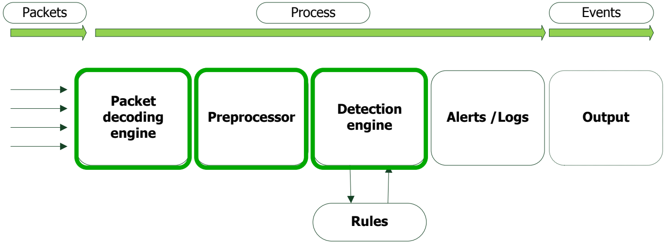
Components


Packets
Process
Events
Packet decoding engine
Preprocessor
Detection engine
Alerts / Logs
Output
Rules

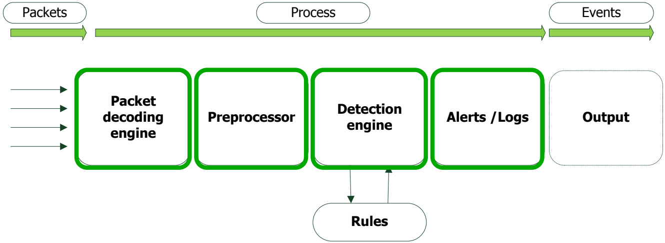
Components


Packets
Process
Events
Packet decoding engine
Preprocessor
Detection engine
Alerts / Logs
Output
Rules

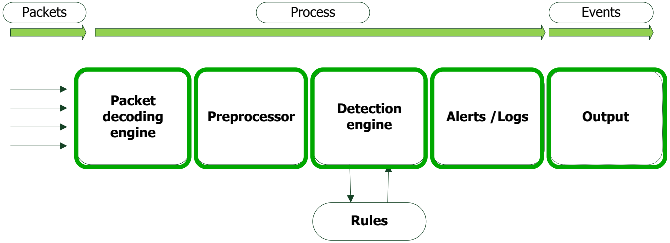
Components


Packets
Process
Events
Packet decoding engine
Preprocessor
Detection engine
Alerts / Logs
Output
Rules

Components


Packets
Process
Events
Packet decoding engine
Preprocessor
Detection engine
Alerts / Logs
Output
Rules

Components


Packets
Process
Events
Packet decoding engine
Preprocessor
Detection engine
Alerts / Logs
Output
Rules

Outline

Our work

- Detection
- Analysis
- Incident

Outline

Our work

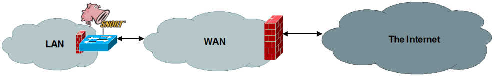
- Detection
- Analysis
- Incident

Our challenges

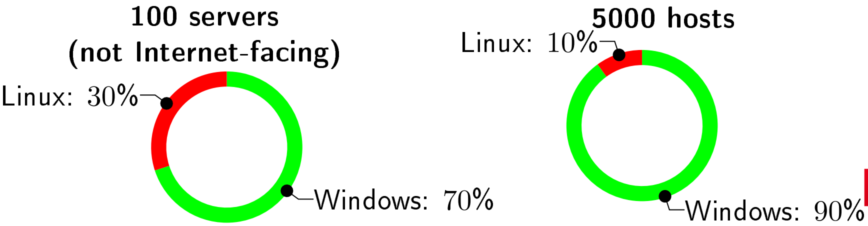
- Deal with a huge number of events
- Transfer partial analysis to administrators and help desk
- Automation (detection and incident handling)

Architecture

- Sensor
  - 1 sensor connected on switch spanning port
  - 1 Gbit/s of monitored traffic
  - LAN <-> LAN and LAN <-> Internet traffic



- Assets repartition



Main threats

- Compromized hosts (trojans, etc.) ⇒ Immediate action
- Policy violations (applications, etc.) ⇒ Send periodic report
- Inside threats (scans, etc.) ⇒ Things to look at

Management

- Emerging Threats, Sourcefire VRT and homemade rules
  - Daily updates with PulledPork
  - Disable rulesets inappropriate for our environment
  - Identify useless rules (obsolete, ineffective, etc.)
  - Review rules on analysis

Why?

Packet payload contains some useful protocol informations: User-Agent, Host, URL, etc.

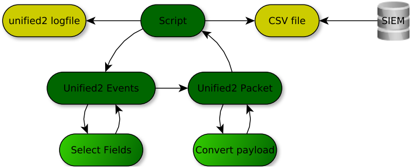
- Track false-positives
- Detect suspicious activities
- Categorize an alert
- Full text search (SIEM)

Bad-unknown alert?

```
262;0;|Tue May 27 08:55:20 2014|;2012810;|ET CURRENT_EVENTS HTTP Request to a *.tk
domain|;1;7;3;bad-unknown;2;X.X.X.X;Y.Y.Y.Y;55543;80;6;0;194;|.......x.......d..E...F1...b...e......P.-
..Z...P..../..GET./podcast/feed.xml.HTTP/1.1..Accept-
Encoding:.gzip,.*.. User-Agent:.RSSOwl/2.2.1 .(Windows;.U;.fr)..Host:.Z.Z.Z.Z....|
```

How?

PERL script based on SnortUnified module (like Barnyard, with CSV output and ASCII payload):



Example

```
15655;0;Tue May 13 17:42:30 2014;2016223;ET TROJAN Andromeda Checkin;1;6;21;trojan-
activity;1;X.X.X.X;Y.Y.Y.Y;4598;80;6;0;225; ...........Z......E...G.@...d.....[.h..2.P.......[P.......
POST./one/image.php.HTTP/1.1..Host:.Z.Z.Z.Z..User-Agent:.Mozilla/4.0..
Content-Type:.application/x-www-form-urlencoded..Content-Length:.100..Connection:.close....
```

External IP

Reputation

- Sources: Emerging Threats, AlienVault, SpyEye, etc.
- Data: Range, IP and Domain
- Process: Snort® IP reputation preprocessor or SIEM

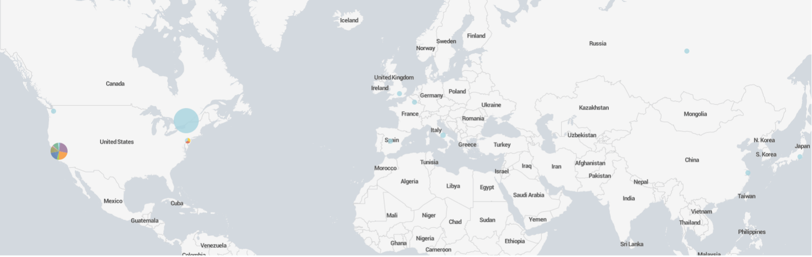
Limits

- Reputation preprocessor only works with IP (no threat score, no bad reputation type)
- Reputation preprocessor alerts don't provide list sources

External IP

Geolocation

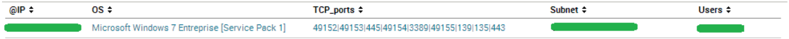
- Sources: MAXMIND
- Data: IP
- Process: SIEM or PERL script (not scalable)



Internal IP

Assets technical properties

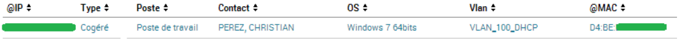
- What, who, etc.?
- Sources: SIEM, SCCM, OCS-NG



@IP ⬍	OS ⬍	TCP_ports ⬍	Subnet ⬍	Users ⬍
[redacted]	Microsoft Windows 7 Entreprise [Service Pack 1]	49152|49153|445|49154|3389|49155|139|135|443	[redacted]	[redacted]

Assets organisational properties

- Contact, sensibility, etc.?
- Sources: CMDB



@IP ⬍	Type ⬍	Poste ⬍	Contact ⬍	OS ⬍	Vlan ⬍	@MAC ⬍
[redacted]	Cogéré	Poste de travail	PEREZ, CHRISTIAN	Windows 7 64bits	VLAN_100_DHCP	D4:BE:[redacted]

Why?

Way to

- Detect typical suspicious traffic patterns
- Detect behavior changes
- Detect anomalies
- Map traffic in contextual view
- Show security metrics

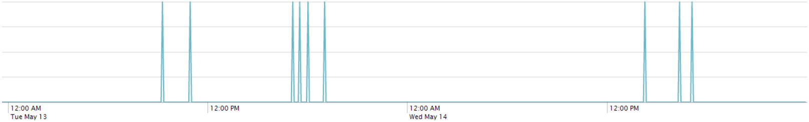
⇒ And delegate partial analysis to administrators and help desk!

Examples - Malware (1/2)

Principal steps

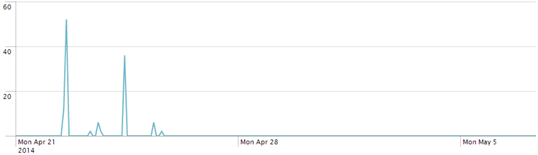
- Check connectivity
- Connect C&C
- Data exchange (orders, data exfiltration, update, etc.)

Trojan periodic requests:

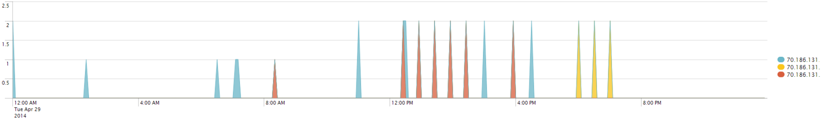


Examples - Malware (2/2)

Number of events:



Multiple requests to hosts in the same IP range:



Limits

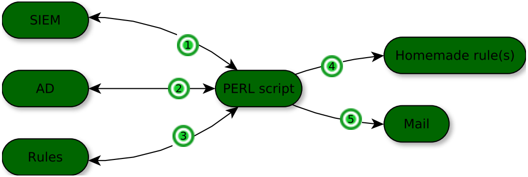
Automation

- Difficulty to define simple metrics (number of events, timeline, etc.)
- Metrics need to be often updated (malware evolution, etc.)

Objectives

- Give help desk useful informations (IP, location, etc.)
- Keep incident informations for further analysis (management metrics, etc.)
- Validate resolution

How?



Steps

1. Request alert(s) and connected user
2. Request connected user informations (fullname, mail)
3. Parse rule informations
4. Create rule with specific informations: message, classtype, reference, sid and validate conformity
5. Send mail to help desk with link to SIEM dashboard

Example

```
alert tcp XX.XX.XX.XX any -> YY.YY.YY.YY ZZ
(msg:" INCIDENT-DDMMYYYY-ID ";flow:to_server,established; urilen:>80; content:"GET";
http_method; content:"User-Agent|3a| Mozilla/5.0 (compatible|3b| MSIE 9.0|3b| Windows NT 6.1|3b|
Trident/5.0)|0d 0a|"; fast_pattern:57,20; depth:77; http_header; content:!"Referer|3a| "; http_header;
content:!"Accept|3a| "; http_header; reference:url,www-xxx.cea.fr/incidents/xxxx ;
classtype:incidents ; sid:30000000 ; rev:1;)
```

Is Snort® useful in our context?

Of course

Permit to detect compromized hosts without false-positives

Accuracy in policy violation detection

However

Many rules became obsolete with network encryption generalization

Many ways to bypass IDS

Automation and transfer are still a great challenge not completely resolved at this time

Questions?