

# "Les outils de détection , supervision et traitement des évènements de sécurité".

# Deux grands domaines de la supervision et la détection des incidents.

- Le "**Network Security Monitoring**"\*\* est une discipline mêlant les aspects instrumentation réseau (collecte de trafic) et détection d'intrusions, essentiellement via des NIDS (**Network Intrusion Detection Systems**).
- Le "**Host Intrusion Detection System**" est orienté "machine" et s'appuie sur les traces (logs) de ces dernières.

# Définitions

- "**Threat Hunting**" ou chasse aux cyber-menaces est une activité de défense active. Il s'agit de détecter des "**Advanced Persistent Threats**" ou menaces sophistiquées et persistantes dans la durée.
- "**Network FORENSICS**": cette activité désigne l'étude postmortem des traces réseaux après un incident de sécurité.
- **Détection d'intrusion**: il s'agit en temps réel de détecter et identifier des attaques déjà connues ou des anomalies.

# Deux modes de détection:

- Par **scénario**: il s'agit de reconnaître la signature d'une attaque en recherchant des chaînes de caractères dans les flux des données transitant sur le réseau. On peut choisir de "reconnaître et/ou de bloquer" (**Intrusion Detection System** versus **Intrusion Prevention System**).
- Par **comportement**: Il s'agit de détecter des changements dans le trafic (par exemple par analyse en composantes principales ou à l'aide des outils de type "machine learning").

# Les sources de détection

- Sniffing & PCAP & Netflow (Utilisation de TAP).
- Table CAM d'un switch.
- Table de routage.
- Logs DHCP, DNS et service réseaux...
- Logs systèmes (EVT, Rsyslog).
- Log IPS/IDS.
- Logs des firewall.
- Logs des proxy.
- Logs applicatifs.

# Les outils historiques de "*threat hunting*"

**Sniffers:** Wireshark, tcpdump

**Collecteurs & aggrégateurs NetFlow:** nfsen,  
nfpcap, argus 😊

# Définition des IDS et IPS

- Un **IDS** détecte des "motifs d'attaques" dans les trames réseaux et implémentés sous forme de signature. L'**IPS** est le pendant armé de l'IDS et va bloquer les IP qui ont levé une alerte.
- Ce type d'outil est moins efficace avec la généralisation du chiffrement mais est capable de donner des éléments sur les flux réseaux des attaquants. A ce titre il est toujours utile.
- Un **WAF (Web Application Firewall)** est un IPS dédié aux applications web.



## Outils IDP/IPS: **Snort!**

C'est un IDS historique. A ce titre le format de ses règles est un standard y compris pour son concurrent Suricata. Racheté par Cisco, il a été ré-écrit (version 3) pour des raisons de performance (multi-threading et inspection http entre autre).



## Outils IDP/IPS: Suricata

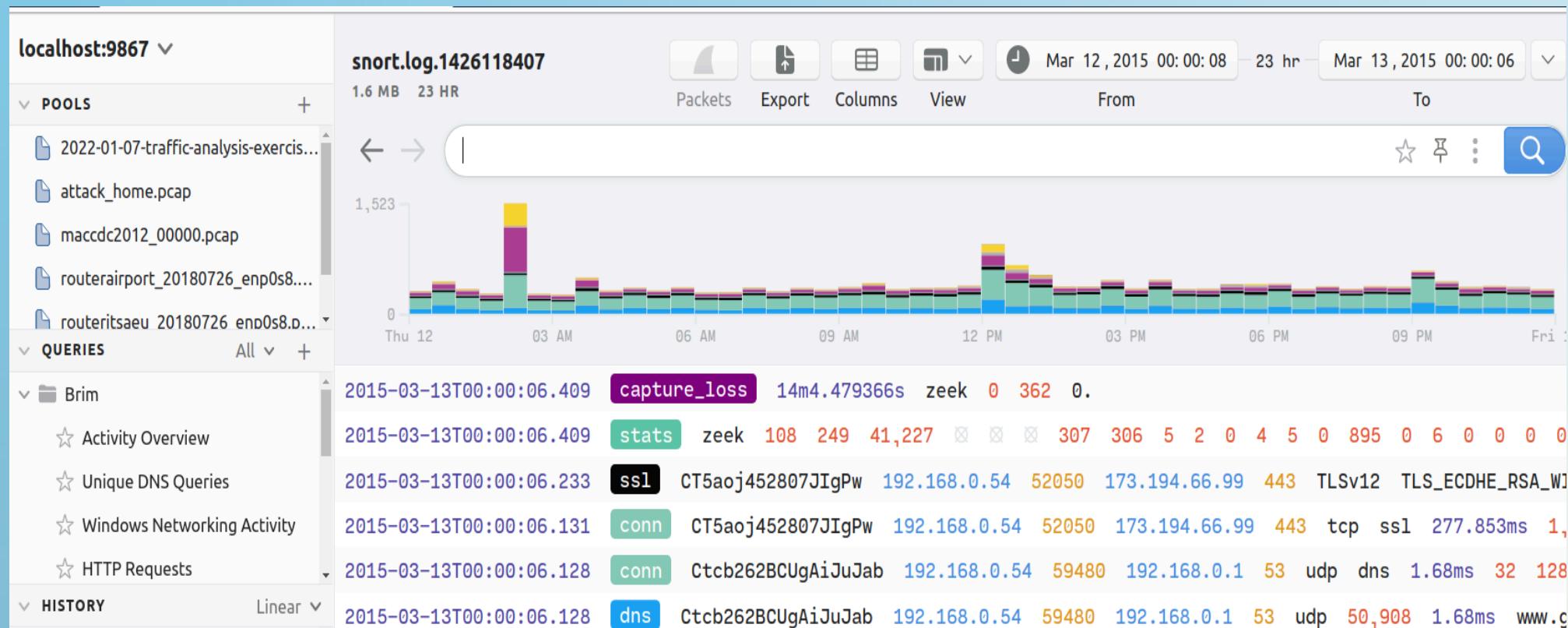
Plus récent (2009) que SNORT il a bénéficié d'une architecture modulaire et multi-threadé dès sa conception. C'est un outil intégré dans beaucoup d'autres du fait de son positionnement "opensource". Il ne sert pas qu'à la détection mais il est capable aussi



## **ZEEK est un analyseur comportemental du trafic**

Zeek (ex Bro) est un peu à part des deux précédents IDS/IPS. Il est capable d'analyse protocolaire. Programmable (conception évènementielle), il est hautement adaptable. C'est le concurrent de Suricata.

# Zui ❤️ est un client qui lit et "parse" rapidement de gros fichiers au format pcap



# Zui embarque Suricata et affiche ses alertes

The screenshot shows the ZUI (Zabbix User Interface) web application. On the left, there's a sidebar with a tree view:

- localhost:9867** (selected)
- POOLS**: 2022-01-07-traffic-analysis-exercis..., attack\_home.pcap, maccdc2012\_00000.pcap, routerairport\_20180726\_enp0s8...., routeritsaeu 20180726\_enp0s8.p...
- QUERIES**: All, +
  - HTTP Post Requests
  - Show IP Subnets
  - Suricata Alerts by Category
  - Suricata Alerts by Source and ...
  - Suricata Alerts by Subnet

The main area displays a log file named **snort.log.1426118407** with a size of **1.6 MB** and a duration of **23 HR**. The timestamp is **Mar 12, 2015 00:00:08**. The interface includes buttons for **Packets**, **Export**, **Columns**, and **View**. A search bar contains the query: **event\_type=="alert" | alerts := union(alert.category) by src\_ip, dest\_ip**.

| src_ip          | dest_ip      | alerts                                    |
|-----------------|--------------|---|
| 108.160.162.76  | 192.168.0.54 | [[Potential Corporate Privacy Violation]] |
| 95.154.26.34    | 192.168.0.2  | [[Generic Protocol Command Decode]]       |
| 108.160.170.50  | 192.168.0.54 | [[Potential Corporate Privacy Violation]] |
| 192.168.0.54    | 65.55.54.42  | [[Misc activity, Unknown Traffic]]        |
| 108.160.165.211 | 192.168.0.54 | [[Potential Corporate Privacy Violation]] |
| 108.160.165.84  | 192.168.0.54 | [[Potential Corporate Privacy Violation]] |
| 108.160.167.35  | 192.168.0.54 | [[Potential Corporate Privacy Violation]] |

**"Open Source", Zui s'appuie sur des données super-structurées formant un "data-lake", "parsable" avec un outil zq (modèle en "pipe" comme jq). Zed est un langage de requête qui permet de faire des recherches dans les données.**

The screenshot shows the Zed interface with the following details:

- File:** snort.log.1426118407
- Size:** 1.6 MB
- Duration:** 23 HR
- Time Range:** From Mar 12, 2015 00:00:08 to Mar 13, 2015 00:00:06
- Toolbar:** Packets, Export, Columns, View
- Search Bar:** method=="POST" | cut ts, uid, id, method, uri, status\_code | sort id.resp\_p
- Table Headers:** uid, id.orig\_h, id.orig\_p, id.resp\_h, id.resp\_p, method, uri
- Table Data:** (5 rows shown)

| uid                | id.orig_h      | id.orig_p | id.resp_h   | id.resp_p | method | uri                      |
|--------------------|----------------|-----------|-------------|-----------|--------|--------------------------|
| CI0Tqn2MKb8heZrczh | 217.195.49.146 | 54331     | 192.168.0.2 | 80        | POST   | /skyblue/index.php?pid=4 |
| Cn6gUH1nt3CNrLncj1 | 217.195.49.146 | 54314     | 192.168.0.2 | 80        | POST   | /skyblue/index.php?pid=4 |
| ChgT0tHLen9zgs8c1  | 217.195.49.146 | 54312     | 192.168.0.2 | 80        | POST   | /skyblue/index.php?pid=4 |
| CE4RNT2M4Wj3JbS04k | 217.195.49.146 | 54311     | 192.168.0.2 | 80        | POST   | /skyblue/index.php?pid=4 |
| C16Tvmeq98ydW4xo8  | 217.195.49.146 | 54304     | 192.168.0.2 | 80        | POST   | /skyblue/index.php?pid=4 |

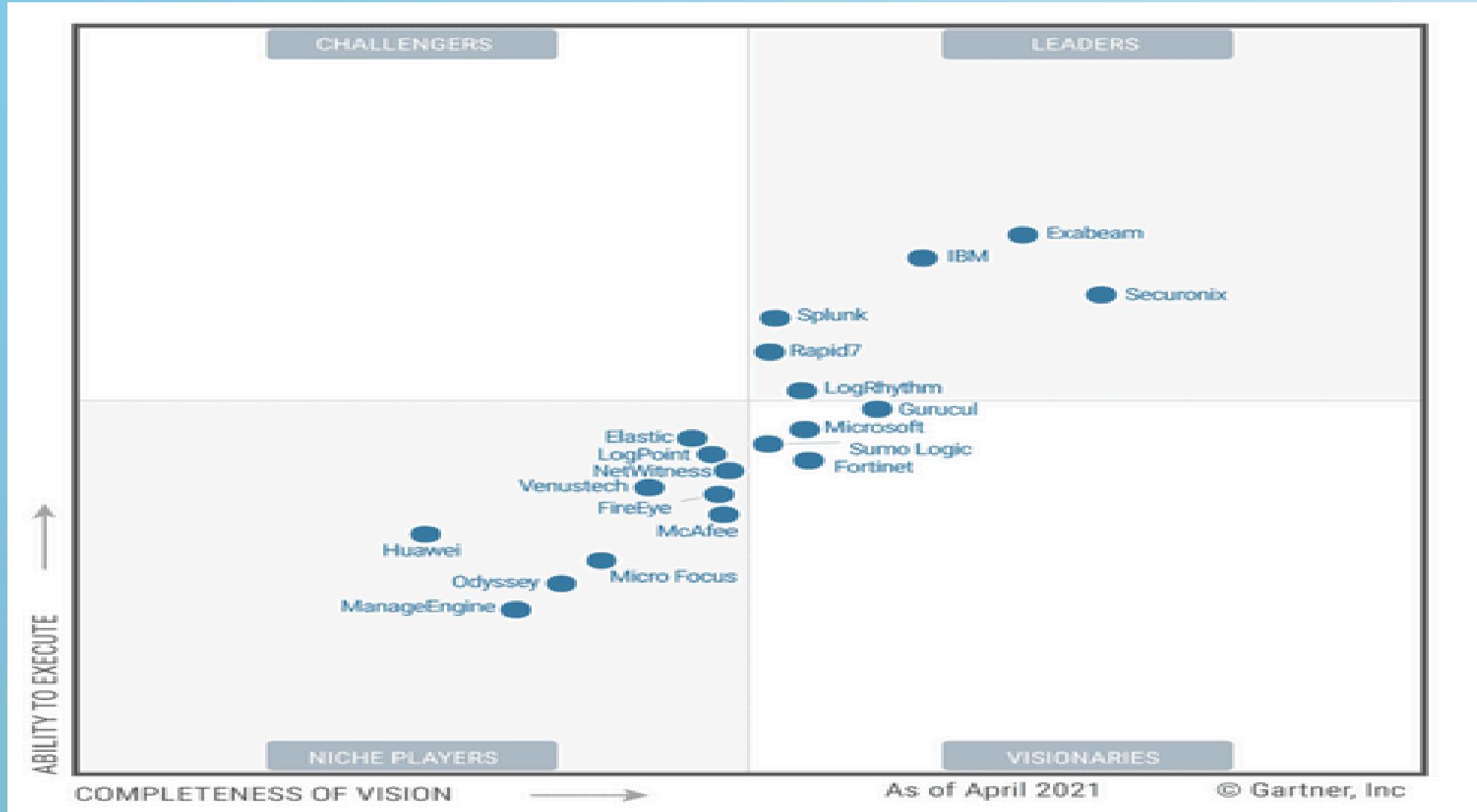
# Origine et définition du SIEM.

- **SIEM = SIM + SEM**
- Le **SIM** (Security Information Management) correspond à la partie stockage et analyse des données. Il comprend la collecte , l'indexation, le Stockage et la recherche des données.
- Le **SEM** (Security Event Management) correspond à la partie détection (temps réel ou différé), la corrélation des évènements et le traitement d'incidents de sécurité.

# Les SIEM (**S**ecurity **I**nformation and **E**vent **M**anagement )

- La détection et la neutralisation des menaces demandent de corrélérer les informations de multiples sources.
- C'est la promesse du **SIEM** qui est utile en particulier dans les S.O.C ("Security Operation Center").
- L'évolution des SIEM se fait comme dans d'autres secteurs de l'I.T. vers le **CLOUD** et l'**I.A.**.

# La concurrence est rude sur ce segment



# Les "10 commandements du SIEM"

(source podcast "no limit sécu" épisode 334 avec Etienne Ladent et Thomas Burnouf)

1. Des relais de collecte de logs **indépendants** tu mettras => pour avoir un tampon de données avant le SIEM et éviter le D.O.S. involontaire , la perte, filtrer et diminuer les coûts de stockage.
2. Les "**regex**" tu apprendras" => pour retrouver efficacement des motifs. Un savoir-faire des analystes.
3. Le "**zéro alerte**" tu viseras => "Trop d'alertes tue l'alerte". Paradigme bien connu en supervision et ailleurs.

## Les "10 commandements du SIEM"

4. Tes données d' **inventaire** tu collecteras => un analyste a besoin de ces données d'inventaire pour être efficace.
5. Sur tes **données utilisateurs** tu te formeras => Il faut comprendre les métiers).
6. Sur tes alertes tu **contextualiseras** => (Dans l'esprit d'EBIOS le métier oriente l'analyste et peut donner du sens à l'attaque).
7. Sur tes **recherches** tu refléchiras => les ressources sont limitées et un analyste ne doit pas les monopoliser.

## Les "10 commandements du SIEM"

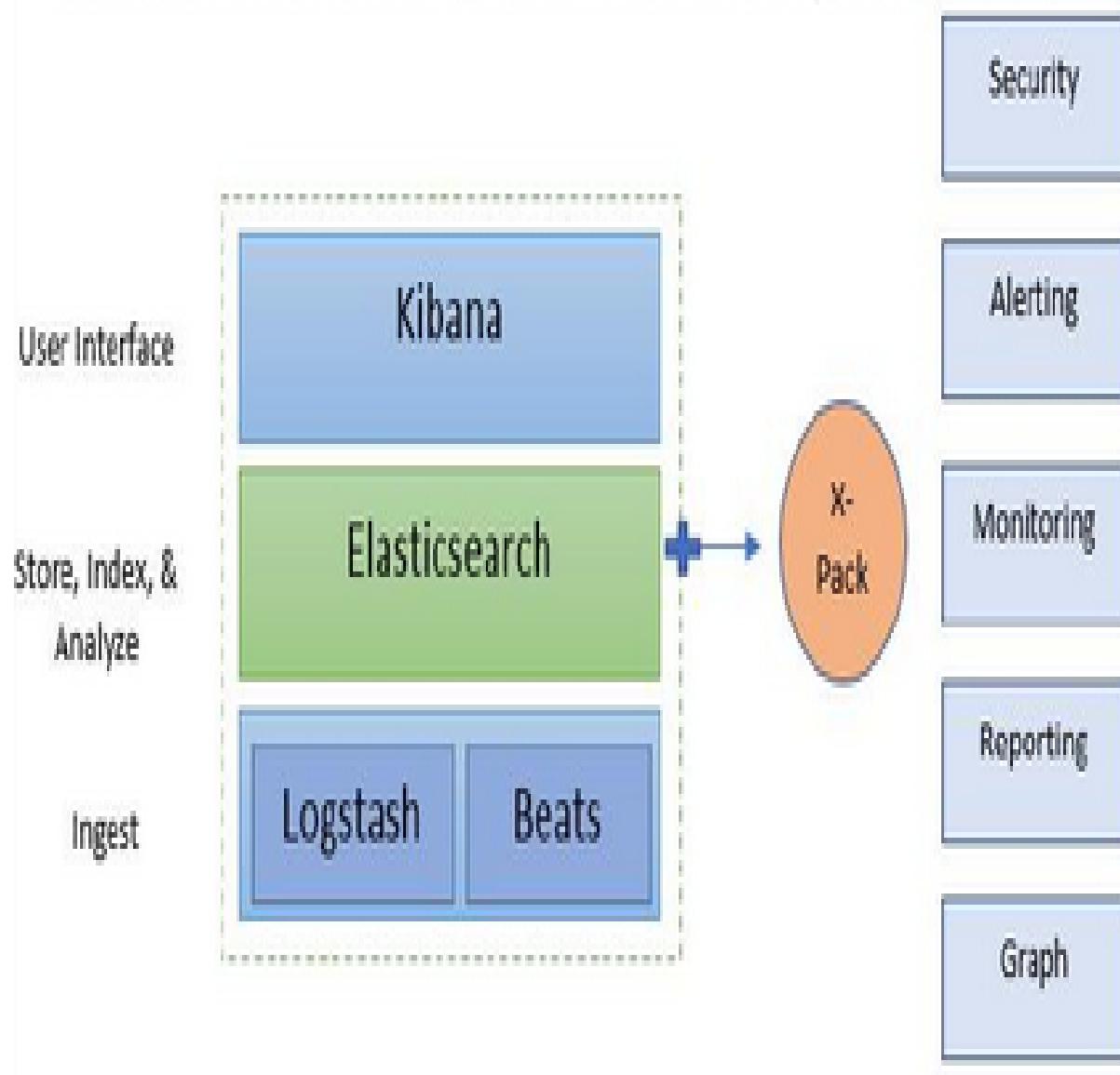
8. Tes alertes régulièrement tu contrôleras => **revue** des alertes suite à l'évolution du SI et des formats logs qui les rendent obsolètes == MCO
9. Tes données tu **documenteras** => évident.
10. Tes noms de données tu **nommeras**. (Un modèle de référence "Splunk Information Model" ) pour nommer de façon cohérente.

# Wazuh & OSSEC

- **Wazuh** est un HIPS (**H**ost **I**ntrusion **PS**ystem). Cette fonction est assurée par un agent qui détecte les anomalies/attaques/programmes malveillants/rootkit notamment à partir des fichiers de logs de la machine et l'écoute du réseau grâce à un "suricata" embarquée.
- **Wazuh** c'est aussi SIEM sur sa partie serveur qui agrège les logs des hôtes sous agent.
- **Wazuh** fait la correspondance entre les incidents de sécurité remontés par le agents et la matrice MITRE.

# Wazuh & OSSEC

- **Wazuh** est capable de faire de la réponse à incident en mode automatique (SOAR) en déclenchant une action sur évènement: par exemple création d'une règle de firewall suite à un scan, une classification virale avec Yara suite à la création d'un fichier.
- **Wazuh** s'assure de la conformité réglementaire des machines hôtes sur plusieurs référentiels.(openSCAP et osquery pour l'inventaire)



## Stack "Elastic"

- Elastic Search stocke les données sur un cluster de plusieurs noeuds
- La suite beats extrait des informations des fichiers de Log (Filebeat), du processus d'audit (Auditbeat), du traffic réseau (Packetbeat), ...

# "Elastic Search"

- C'est un moteur de recherche et d'analyse , accessible en mode "RESTfull" et distribué.
- IL stocke des informations au format JSON.
- Il est performant pour des recherches textes et permet l'indexation de documents non structurées.
- Ses capacités permettent à son utilisateur de trouver des informations via Curl ou au travers de son API.

# Logstash versus Beats

- C'est aussi un composant "Server Side" historique comme "Elastic Search". Écrit en JRuby (lent au démarrage et gourmand), il est important dans fonction ETL (transformation des données avec des regex standardisées).
- Les composants de Beats sont orientés "Client Side" et peuvent alimenter directement "Elastic Search". La "LibBeat" sert de base commune.

## X-Pack

- Il ajoute les fonctions de sécurité, de monitoring , de supervision , de reporting et de "machine learning". Cette dernière fonction est payante et a pour but de détecter des comportements anormaux.

# Elastic Agent & Fleet

Plutôt que de déployer chaque éléments de la suite beats, "Elastic Agent" s'installe sur les "endpoints" (les serveurs, les PC clients ...) et gère l'ensemble des "beats". Il est déployé via "Fleet" qui est un composant de "**Kibana**" (le front-end de la suite Elastic). **Fleet** permet de gérer les agents et de les déployer de façon centralisée.

## "Endpoint Protection Platform" et les "Endpoint Detection and Response"

Ces composants permettent de combiner la protection du poste de travail avec la détection et la remédiation en cas d'alerte virale. Ils sont chargés de la protection des équipements terminaux (pc, smartphone ..):

- Les **EPP** sont des antivirus avec des fonctionnalités de prévention des menaces.
- Les **EDR** sont orientés sur le détection comportementale via l'**IA**, la surveillance de la mémoire et la présence d' "**Indicateurs de Compromissions**" (**IoC**).

## "Dans la famille acronyme il y a" :

- SOC: "**S**ecurity **O**peration **C**enter.
- NDR: "**N**etwork **D**etection and **R**esponse.
- EDR: "**e**xtended **D**etection et **R**eponse".
- CIRT: "**C**omputer **S**ecurity **I**ncident **R**esponse **T**eam.
- SOAR: "**S**ecurity **O**rchestration **A**utomation and **R**esponse".
- MDR: "**M**anagement **D**etection and **R**esponse".

## Si vous voulez être incompris par les "moldus"

- Le **NDR** est le processus de haut niveau qui apporte une visibilité à l'échelle du réseau aux équipes **CSIRT** du **SOC** afin de détecter les comportements malveillants au niveau des infrastructures. Il ne s'occupe donc pas des points terminaux dévolus aux **EPP/EDR**.
- Le **XDR** réunit **NDR** et **EPP**. L'information est correlée au niveau de l'entreprise, il s'agit de centraliser.
- Le **MDR** regroupe les solutions managées qui permettent d'automatiser et de fluidifier au travers du **SOAR** (réponse aux incidents).

## Fonctions des EDR

- Enregistrer les comportements des systèmes sur une granularité "endpoints" ou des "hôtes".
- Déetecter des comportements anormaux.
- Fournir de l'information contextuelle.
- Bloquer les activités malveillantes.
- Permettre la remédiation.

# Elastic Security

**Kibana** est le front end de la suite Elastic et permet de déployer des tableaux de bord. Certains sont orientés sécurité et constituent un SIEM: "**Elastic Security**"

La corrélation est possible par adjonction d'une **TimeLine**. Il est possible d'ouvrir et de suivre des incidents de sécurité au travers de "Case".

## R5.cyber.11 Supervision de la sécurité: les outils de détection

**MA TIMELINE** Autosaved 21 seconds ago

Processes Users Hosts Source IPs Destination IPs

0 0 0 1 1.168k

Add to favorites Attach to case

Query 333338 Correlation Analyzer Notes 1 Pinned

Feb 25, 2022 @ 22:15:52.341 → Feb 26, 2022 @ 22:15:52.341 Refresh Data view

(`_id: "8QPjN38BKJEtSBKk2SMz" x`)

OR (`source.ip: "192.168.1.25" x`)

OR () + Add field

Filter Search KQL

+ Add filter

message event.category event.action host.name source.ip

@timestamp ↓ 1

# Comment tester son SIEM ?

En se basant sur le framework Mitre des techniques d'attaques observées.

| Reconnaissance<br>10 techniques          | Resource Development<br>7 techniques | Initial Access<br>9 techniques      | Execution<br>12 techniques              | Persistence<br>19 techniques               | Privilege Escalation<br>13 techniques      | Defense Evasion<br>40 techniques        |
|--|--------------------------------------|-------------------------------------|---|--|--|---|
| Active Scanning (0/2)                    | Acquire Infrastructure (0/6)         | Drive-by Compromise                 | Command and Scripting Interpreter (2/8) | Account Manipulation (2/4)                 | Abuse Elevation Control Mechanism (2/4)    | Abuse Elevation Control Mechanism (2/4) |
| Gather Victim Host Information (0/4)     | Compromise Accounts (0/2)            | Exploit Public-Facing Application   | Container Administration Command        | BITS Jobs                                  | Access Token Manipulation (0/5)            | Access Token Manipulation (0/5)         |
| Gather Victim Identity Information (0/3) | Compromise Infrastructure (0/6)      | External Remote Services            | Deploy Container                        | Boot or Logon Autostart Execution (1/15)   | Boot or Logon Autostart Execution (1/15)   | BITS Jobs                               |
| Gather Victim Network Information (0/6)  | Develop Capabilities (0/4)           | Hardware Additions                  | Exploitation for Client Execution       | Boot or Logon Initialization Scripts (1/5) | Boot or Logon Initialization Scripts (1/5) | Build Image on Host                     |
| Gather Victim Org Information (0/4)      | Establish Accounts (0/2)             | Phishing (0/3)                      | Inter-Process Communication (0/2)       | Browser Extensions                         | Create or Modify System Process (1/4)      | Deobfuscate/Decode Files or Information |
| Phishing for Information (0/3)           | Obtain Capabilities (0/6)            | Replication Through Removable Media | Native API                              | Compromise Client Software Binary          | Direct Volume Access                       | Deploy Container                        |
| Search Closed Sources (0/2)              | Stage Capabilities (0/5)             | Scheduled Task/Job (4/6)            | Scheduled Task/Job (4/6)                | Domain Policy Modification (1/2)           | Domain Policy Modification (1/2)           | Domain Policy Modification (1/2)        |
| Search Open Technologies                 | Supply Chain                         | Session偷听                           | Session偷听                               | Session偷听                                  | Session偷听                                  | Session偷听                               |

```
> pwsh
PowerShell 7.2.1
Copyright (c) Microsoft Corporation.

https://aka.ms/powershell
Type 'help' to get help.

PS /root> Invoke-AtomicTest T1046
PathToAtomsicsFolder = /root/AtomicRedTeam/atomics

Executing test: T1046-1 Port Scan
Done executing test: T1046-1 Port Scan
```

- On utilise des outils comme ceux d'**Atomic RedTeam** qui vont tester les défenses d'un hôte afin d'en vérifier la conformité.
- On peut aussi rejouer des pcaps qui contiennent des traces des tentatives d'intrusion.

## "Un Anneau pour les gouverner tous..."

- Chaque SIEM a son propre langage de requête avec des champs proches mais pas identiques (*ip\_addr*, *IPAddres...*).
- Développer des requêtes prend du temps et échapper au "*vendor lock-in*" est difficile.

# Les règles Sigma

- L'idée des "**SIGMA rules**" est de proposer un langage commun pour les requêtes de détection d'attaques. C'est un format générique décrivant les règles. C'est l'équivalent des règles Suricata pour les IDS/IPS et de "Yara" pour la détection de malwares.
- On a donc un format décrivant les règles de détection et un convertisseur qui permet de générer des règles adaptées à chaque SIEM (plus de vingt dont des solutions proches des endpoints comme **Chainsaw** et **Hayabusa**...).

# "Un anneau pour les gouverner tous..."



# Les en-têtes d'une règle sigma

```
title: Mimikatz Use
id: 06d71506-7beb-4f22-8888-e2e5e2ca7fd8
status: test
description: This method detects mimikatz keywords
in different Eventlogs (some of them only appear
in older Mimikatz version that are however still used by different threat groups)
references:
- https://tools.thehacker.recipes/mimikatz/modules
author: Florian Roth (Nextron Systems), David ANDRE (additional keywords)
date: 2017/01/10
modified: 2022/01/05
tags:
- attack.s0002
- attack.lateral_movement
- attack.credential_access
- car.2013-07-001
- car.2019-04-004
- attack.t1003.002
- attack.t1003.004
- attack.t1003.001
- attack.t1003.006
```

# Le corps d'une règle sigma

```
logsource:  
    product: windows  
detection:  
    keywords:  
        - 'dpapi::masterkey'  
        - 'eo.oe.kiwi'  
        - 'event::clear'  
        - 'event::drop'  
        - 'gentilkiwi.com'  
        - 'kerberos::golden'  
        ...  
        - 'sekurlsa::'  
filter:  
    EventID: 15 # Sysmon's FileStream Events  
        (could cause false positives when Sigma rules get copied on/to a system)  
condition: keywords and not filter  
falsepositives:  
    - Naughty administrators  
    - AV Signature updates  
    - Files with Mimikatz in their filename  
level: high
```

# Les siem cibles de sigma

| sigma list targets |  |                              |
|--------------------|--|------------------------------|
| Identifier         | Target Query Language                  | Processing Pipeline Required |
| elasticsearch      | Elasticsearch Lucene                   | Yes                          |
| opensearch         | OpenSearch Lucene                      | Yes                          |
| splunk             | Splunk SPL & tstats data model queries | Yes                          |

# Conversion d'une règle sigma en requête SIEM

```
sigma convert -t opensearch -f default -p sysmon win_alert_mimikatz_keywords.yml
Parsing Sigma rules [#####
("dpapi\\:\\masterkey"
OR "eo.oe.kiwi"
OR "event\\:\\clear"
OR "event\\:\\drop"
OR "gentilkiwi.com"
OR "kerberos\\:\\golden"
OR "kerberos\\:\\ptc"
...
OR "privilege\\:\\debug"
OR "privilege\\:\\driver" OR "sekurlsa\\:\\:")
AND (NOT EventID:15)
```

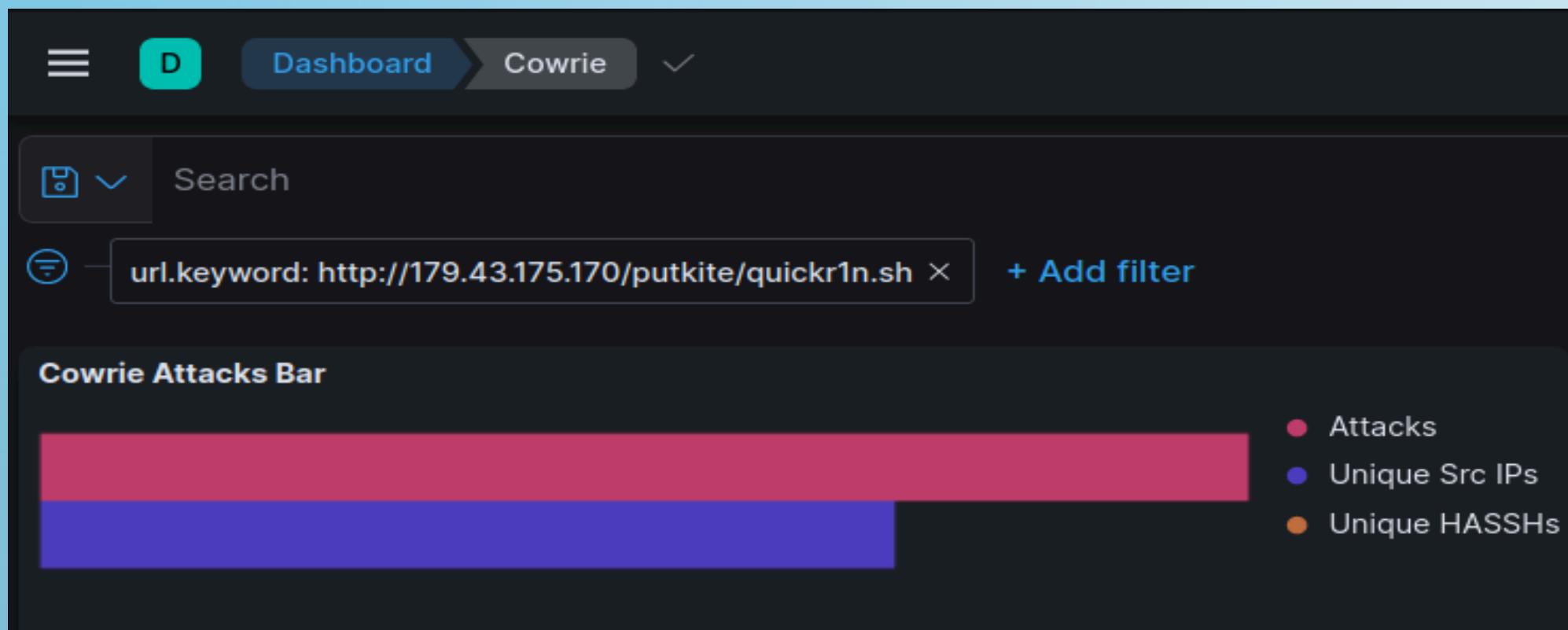
# Génération en language SIEM via sigmac (déprécié)

```
sigmac -t splunk -c splunk-dns
./net_dns_external_service_interaction_domains.yml
#Résultat en SPL
(query="*.interact.sh"
OR query="*.oast.pro"
OR query="*.oast.live"
OR query="*.oast.site"
OR query="*.oast.online"
OR query="*.oast.fun"
OR query="*.oast.me"
OR query="*.burpcollaborator.net"
OR query="*.oastify.com"
OR query="*.canarytokens.com"
OR query="*.requestbin.net"
OR query="*.dnslog.cn")
```

# HoneyPot qui mal y pense...

- Une façon de détecter les intrusions réseaux et d'analyser les évolutions des attaques c'est de mettre en place un **Honeypot**.
- Les honeypots à faible interaction offrent de faux services et n'offrent que peu de privilèges à l'attaquant. **L'interaction** avec l'attaquant est faible. Simple à mettre en oeuvre ils sont limités et recueillent les attaques automatisées et sans intelligence.
- Au contraire les honeypots à forte interaction permettent à l'attaquant d'aller loin dans son parcours, avec à la clef une analyse en profondeur du mode d'attaque mais ils demandent plus d'attention et de ressources.

**TPOT** combine de nombreux honeypots implémentés sous forme de containers Docker. Il donne des renseignements comme ici une URI de téléchargement d'un logiciel malveillant:



# R5.cyber.11 Supervision de la sécurité: les outils de détection



## Bibliographie:

- [PDIS ANSSI](#)
- Hands-On Network Forensic - auteur Nipun Jaswal- Packt
- Learning Elastic Stack 7.0 - Second Edition Shukla, Pranav Kumar M N, Sharath Packt
- Introduction to Network Forensics FINAL VERSION 1.1 ENISA
- Définitions [soc-siem-xdr-mdr](#) par Orange Cyberdéfense
- SOC - Stratégie de détection par Maurugeon Cédric Menelet Alain Misc 120

...