TP-IDS & IPS Luca BONA

IDS/IPS Suricata : Comment ça marche ?

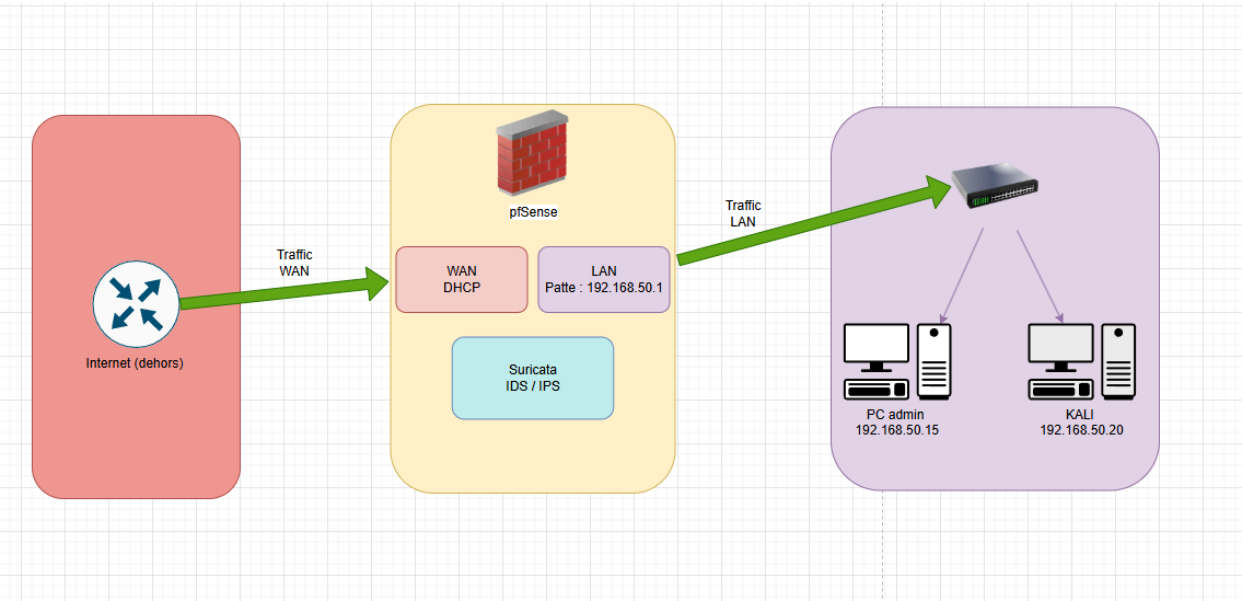
L'idée est simple : au lieu de simplement filtrer les entrées/sorties comme un pare-feu classique, Suricata agit comme un **vigile expert** qui fouille les sacs à l'entrée du bâtiment (le réseau). Il ne regarde pas seulement "qui" passe, mais "ce qu'il transporte" pour détecter des armes ou des comportements suspects.

Le Vigile en une image :

Imaginez une boîte de nuit où les clients (les paquets réseau) veulent entrer.

1. **Mode IDS (Détection) :** Le vigile voit quelqu'un avec un tournevis. Il ne l'arrête pas, mais il note dans son carnet : *"Individu suspect avec un outil au comptoir"* (Génération d'une alerte).
2. **Mode IPS (Prévention) :** Le vigile voit le tournevis, attrape l'individu et le met dehors immédiatement. L'accès est bloqué.

Schéma réseau :



1. Mise en place de l'infrastructure

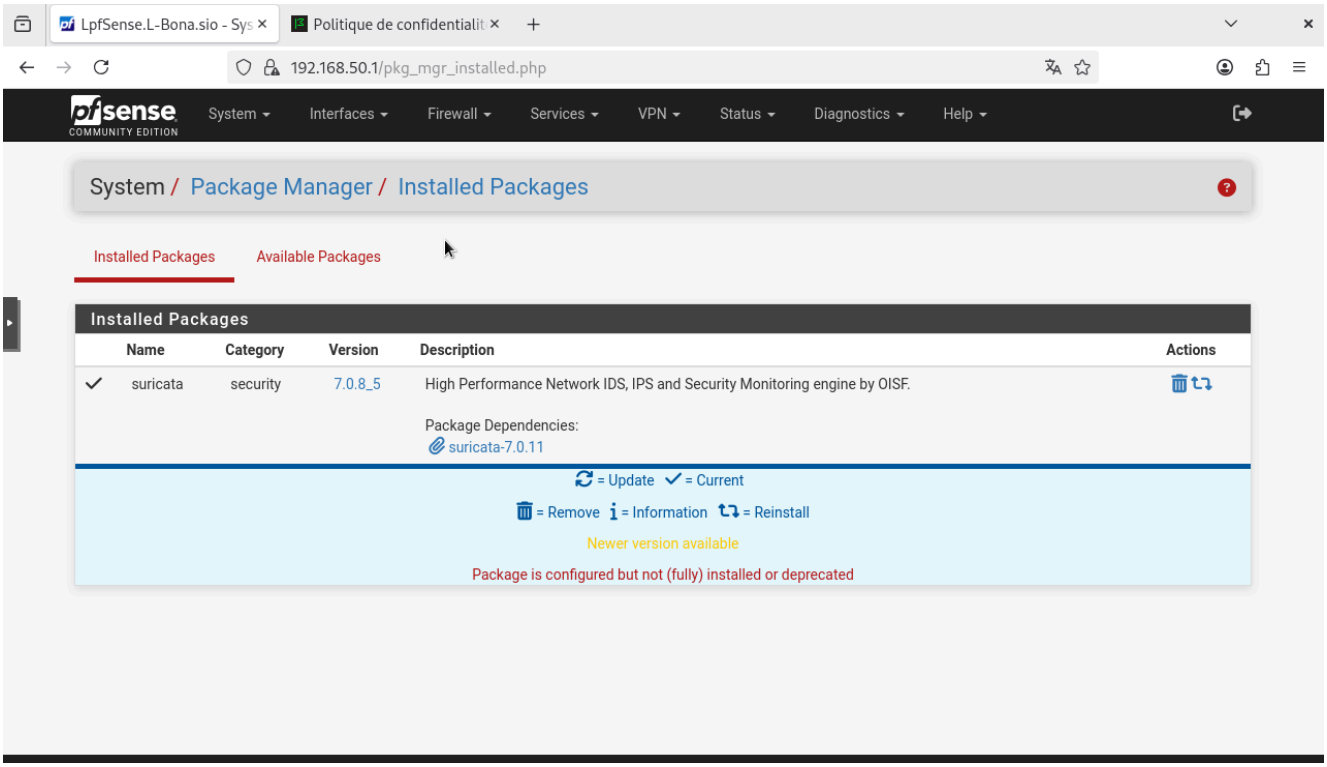
Le laboratoire utilise une architecture en coupure où pfSense sépare le monde extérieur (Kali Linux) de votre réseau interne (Serveur/PC).

Équipement	Rôle	Adresse IP
pfSense (LAN)	Passerelle / Sécurité	`192.168.50.1`
PC Admin	Configuration Web	`192.168.50.15`
Kali Linux	Attaquant (Pentest)	`192.168.50.20`

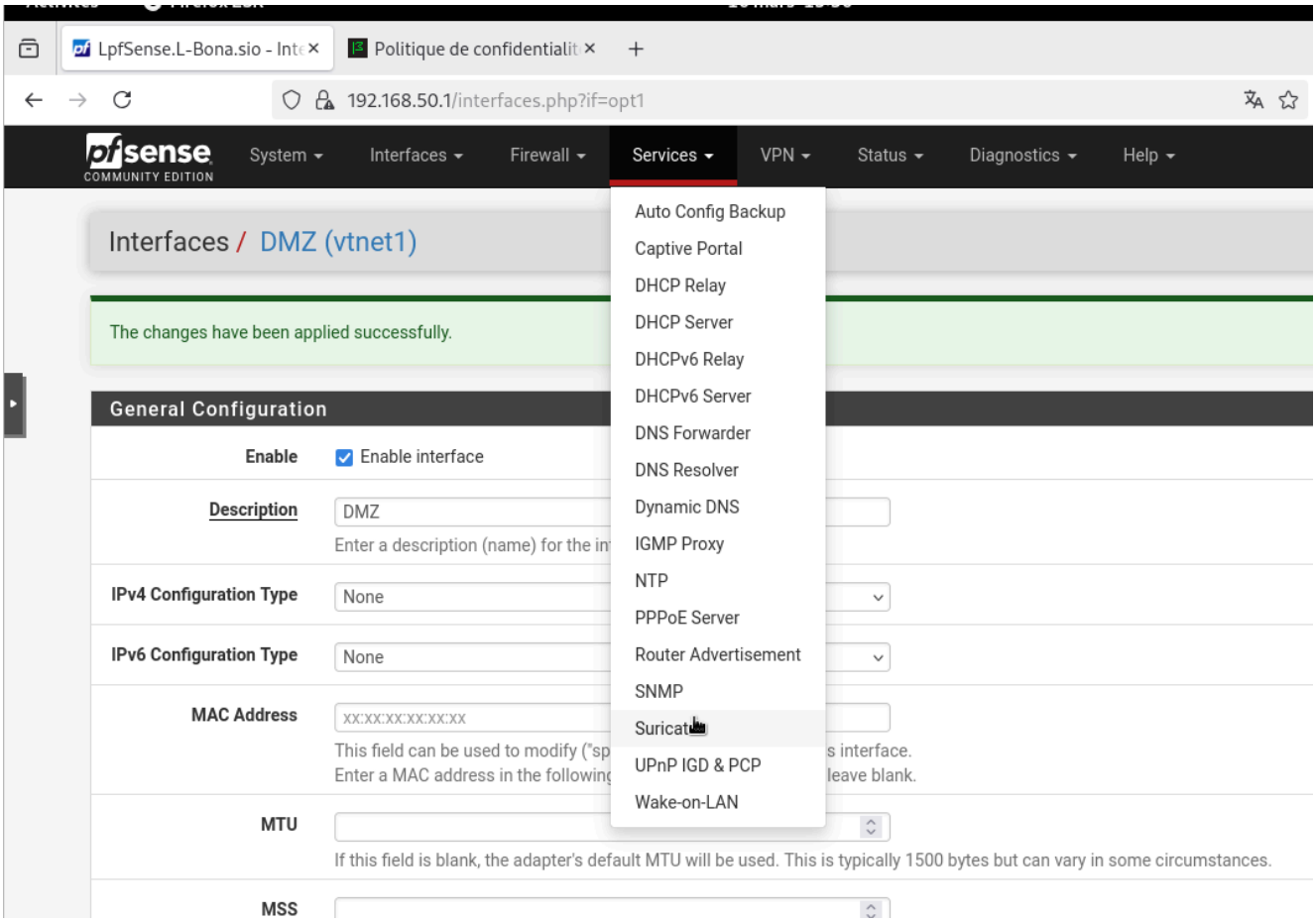
---

2. Installation du package Suricata

1. Allez dans **System > Package Manager > Available Packages**.



2. Cherchez **Suricata** et cliquez sur **Install**.
3. Vérifiez l'installation dans le menu **Services > Suricata**.

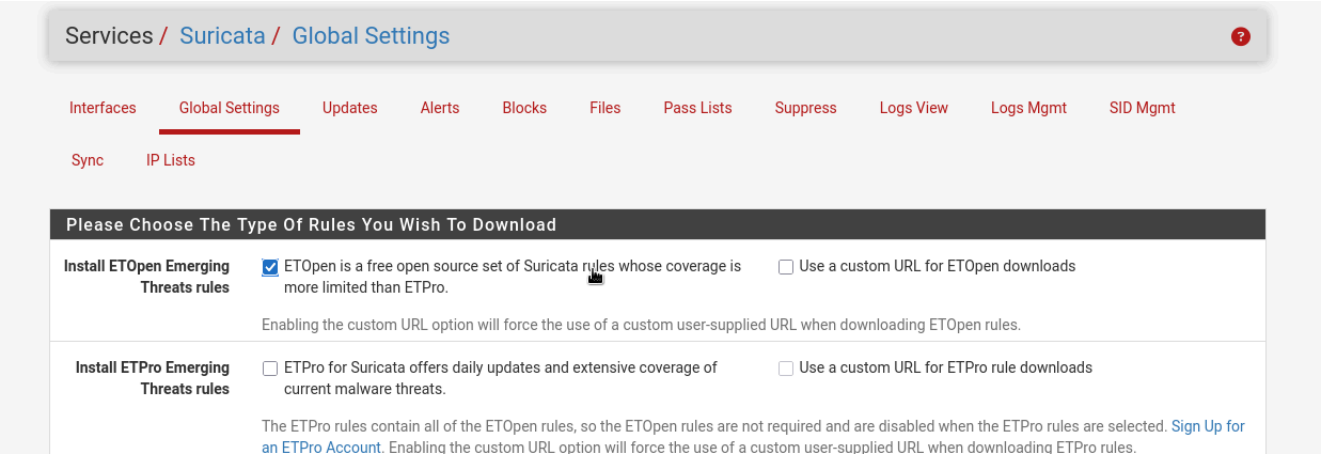


3. Configuration Globale (Le cerveau)

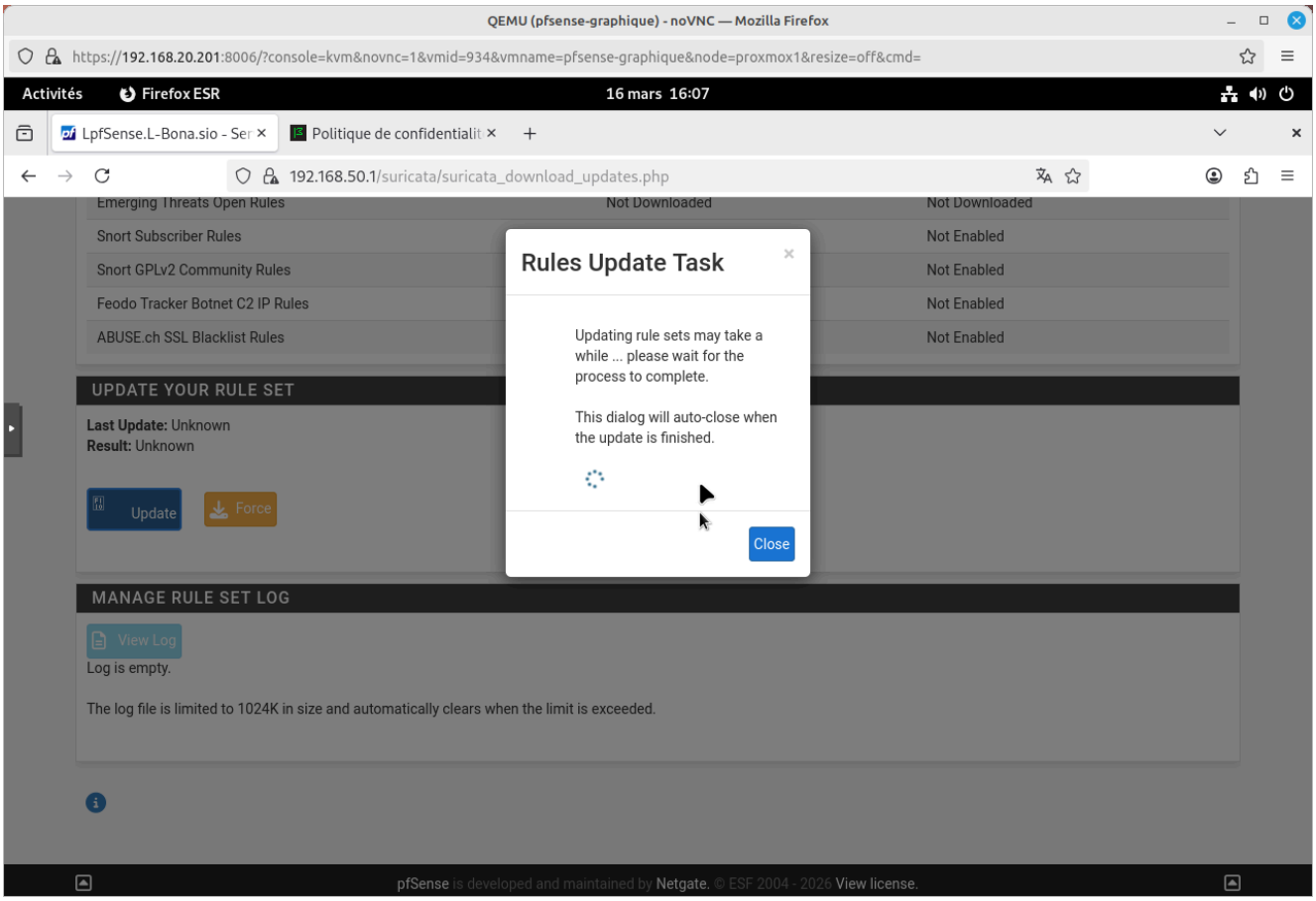
Avant de surveiller, Suricata a besoin de ses "fiches de police" (les règles).

3.1 Paramètres globaux et mise à jour des règles

- **Mise à jour :** Dans l'onglet **Global Settings**, cochez **Enable Emerging Threats Open rules**. Ce sont des règles gratuites et performantes.



- **Téléchargement :** Allez dans l'onglet **Updates** et cliquez sur **Update Rules**. Attendez que la barre soit verte.



> **Q2.** Voici la distinction majeure entre les deux flux :

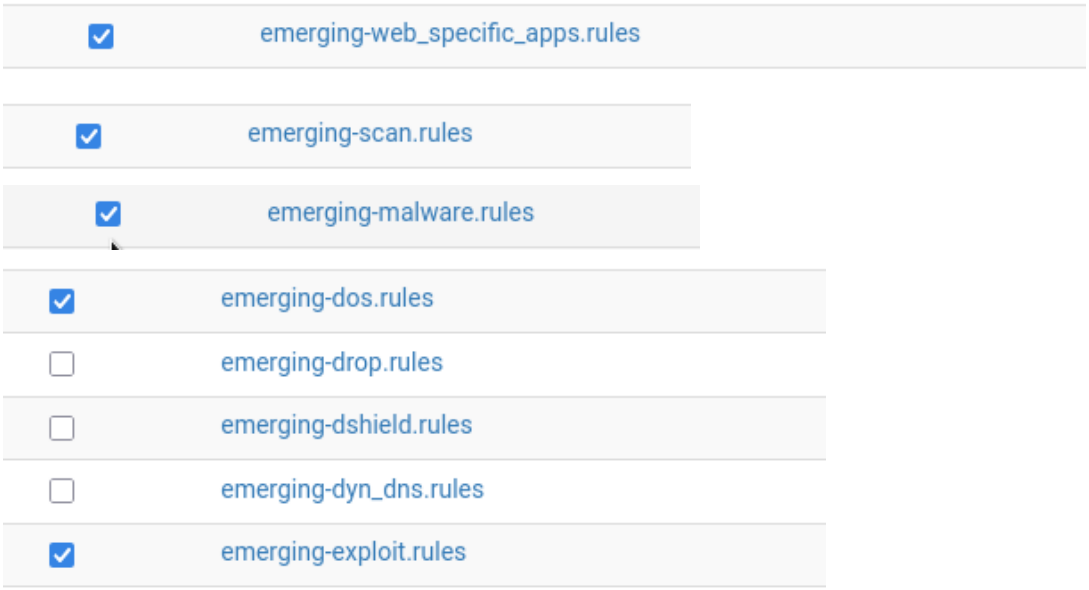
- **ET Open** : Version gratuite alimentée par la communauté, idéale pour les tests et les PME.
- **ET Pro** : Version payante offrant une protection contre les menaces **"Zero-day"** avec des mises à jour beaucoup plus fréquentes et une plus grande précision.._

3.2 Ajouter Suricata sur l'interface LAN

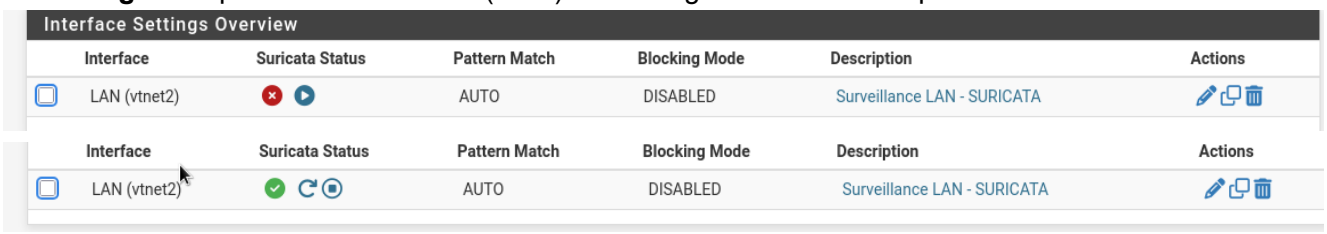
. Configuration de l'Interface (Le vigile au travail)

Il faut maintenant dire à Suricata d'écouter sur le **LAN**.

1. **Services > Suricata > Interfaces > Add**.
2. **Interface :** LAN.
3. **Description :** Surveillance TP.
4. **Logging :** Cochez **Send Alerts to System Log**.
5. **Categories :** Cochez :



6. **Démarrage :** Cliquez sur le bouton ▶ (Start) dans l'onglet Interfaces. La pastille doit devenir verte.



---

4. Création de règles personnalisées

Parfois, les règles par défaut ne suffisent pas. On utilise alors des **Custom Rules**.

- **Syntaxe d'une règle :** `action protocole source -> destination (message; options; sid;)`

Rendez-vous dans **Interfaces > Edit (LAN) > LAN Rules** et ajoutez ces lignes :

Bash

```
# Règle 1 : Détecter un scan ICMP (ping) depuis l'extérieur alert icmp any any ->
$HOME_NET any (msg:"ICMP Ping détecté"; itype:8; sid:1000001; rev:1;)

# Règle 2 : Détecter une tentative de connexion SSH alert tcp any any -> $HOME_NET 22
(msg:"Tentative de connexion SSH"; flags:S; sid:1000002; rev:1;)

# Règle 3 : Détecter un scan de ports SYN (5 tentatives/30s) alert tcp any any ->
$HOME_NET any (msg:"Scan de ports SYN detecte"; flags:S; threshold:type threshold, track
by_src, count 5, seconds 30; sid:1000003; rev:1;)

# Règle 4 : Détecter le mot 'password' en clair dans HTTP alert http any any -> $HOME_NET
any (msg:"Mot de passe en clair HTTP"; content:"password"; nocase; http_uri; sid:1000004;
rev:1;)

# Règle 5 : Détecter un scan nmap via le User-Agent HTTP alert http any any -> $HOME_NET
any (msg:"Scan nmap detecte - User-Agent"; content:"Nmap"; http_user_agent; sid:1000005;
rev:1;)
```

Explication des règles :

Règle 1 : Détection du Ping (ICMP)

`alert icmp any any -> $HOME_NET any (msg:"ICMP Ping détecté"; itype:8; sid:1000001; rev:1;)`

- `alert icmp` : Génère une alerte si le protocole est ICMP.
- `any any -> $HOME_NET any` : Concerne n'importe quelle adresse/port source vers votre réseau local (`$HOME_NET`).
- `itype:8` : C'est l'option clé. Le type 8 correspond spécifiquement à une requête "**Echo Request**" (le signal envoyé lors d'un ping).
- `sid:1000001` : Identifiant unique pour cette règle locale.

Règle 2 : Tentative de connexion SSH

`alert tcp any any -> $HOME_NET 22 (msg:"Tentative de connexion SSH"; flags:S; sid:1000002; rev:1;)`

- `tcp ... -> $HOME_NET 22` : Surveille le trafic TCP à destination du port 22 (port standard du SSH).
- `flags:S` : Filtre uniquement les paquets avec le flag **SYN** activé. Cela signifie qu'on ne détecte que le début d'une connexion (la tentative d'ouverture), pas tout le flux de données.

Règle 3 : Scan de ports SYN (Seuil de détection)

`alert tcp any any -> $HOME_NET any (msg:"Scan de ports SYN detecte"; flags:S; threshold:type threshold, track by_src, count 5, seconds 30; sid:1000003; rev:1;)`

- `flags:S` : Comme pour la règle 2, on cherche des tentatives de connexion (SYN).

- `threshold` **(Le seuil)** : C'est l'intelligence de la règle.
  - `track by_src` : On surveille l'activité par adresse IP source (l'attaquant).
  - `count 5, seconds 30` : L'alerte ne se déclenche que si une même IP tente **5 connexions** en l'espace de **30 secondes**. Cela permet de distinguer un utilisateur normal d'un outil de scan automatique comme Nmap.

Règle 4 : Mot de passe en clair (HTTP)

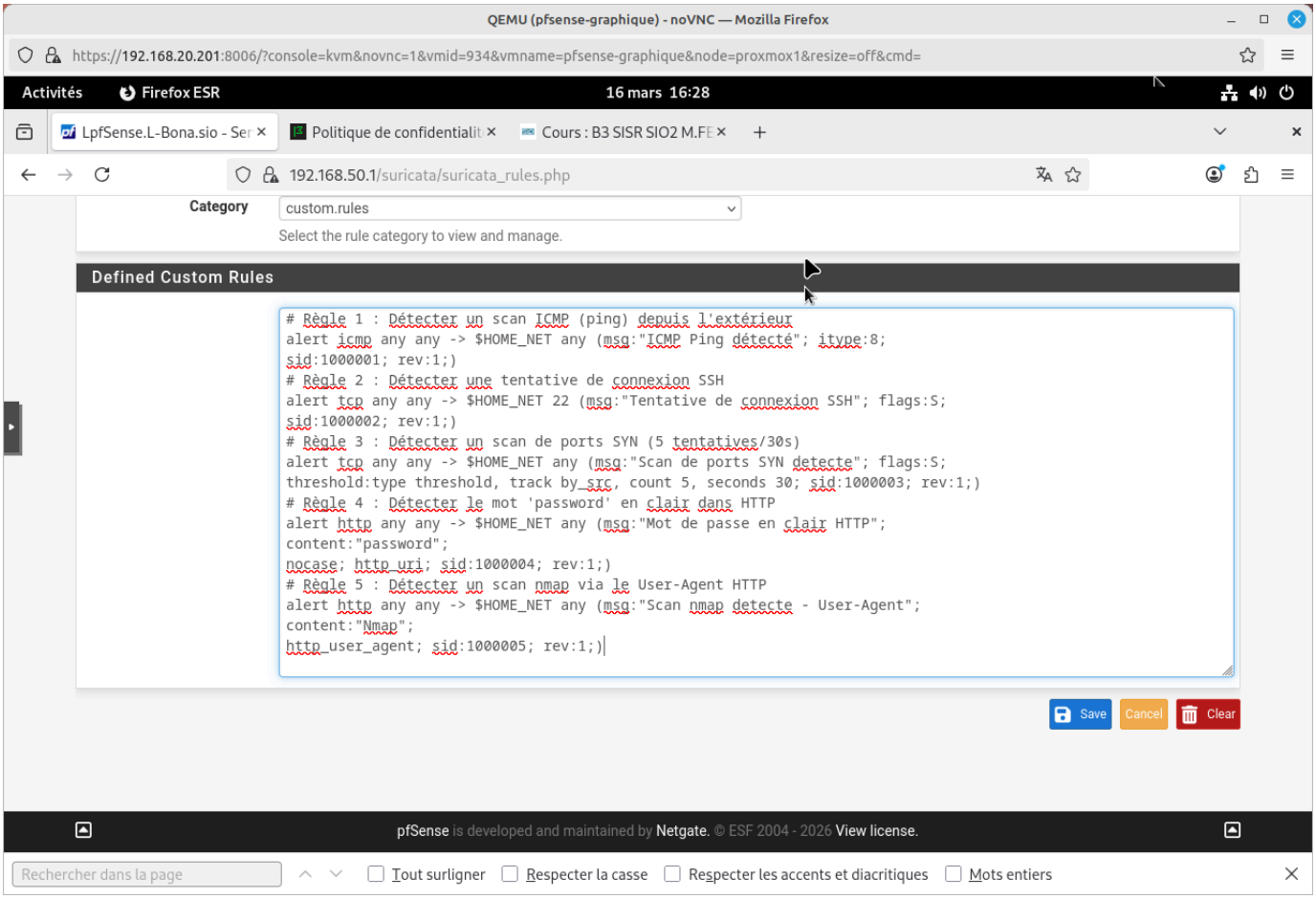
```
alert http any any -> $HOME_NET any (msg:"Mot de passe en clair HTTP"; content:"password"; nocase; http_uri; sid:1000004; rev:1;)
```

- `alert http` : Utilise le moteur d'inspection applicative de Suricata pour analyser le protocole Web.
- `content:"password"` : Cherche la chaîne de caractères précise "password" dans le paquet.
- `nocase` : Rend la recherche insensible à la casse (détecte "Password", "PASSWORD", etc.).
- `http_uri` : Précise que la recherche doit se faire uniquement dans l'URL de la requête.

Règle 5 : Détection de Nmap (User-Agent)

```
alert http any any -> $HOME_NET any (msg:"Scan nmap detecte - User-Agent"; content:"Nmap"; http_user_agent; sid:1000005; rev:1;)
```

- `content:"Nmap"` : Cherche la signature de l'outil de scan.
- `http_user_agent` : Au lieu de chercher partout, Suricata regarde spécifiquement dans le champ "**User-Agent**" du navigateur. Par défaut, Nmap s'identifie lui-même dans ce champ lorsqu'il scanne des services Web.
-

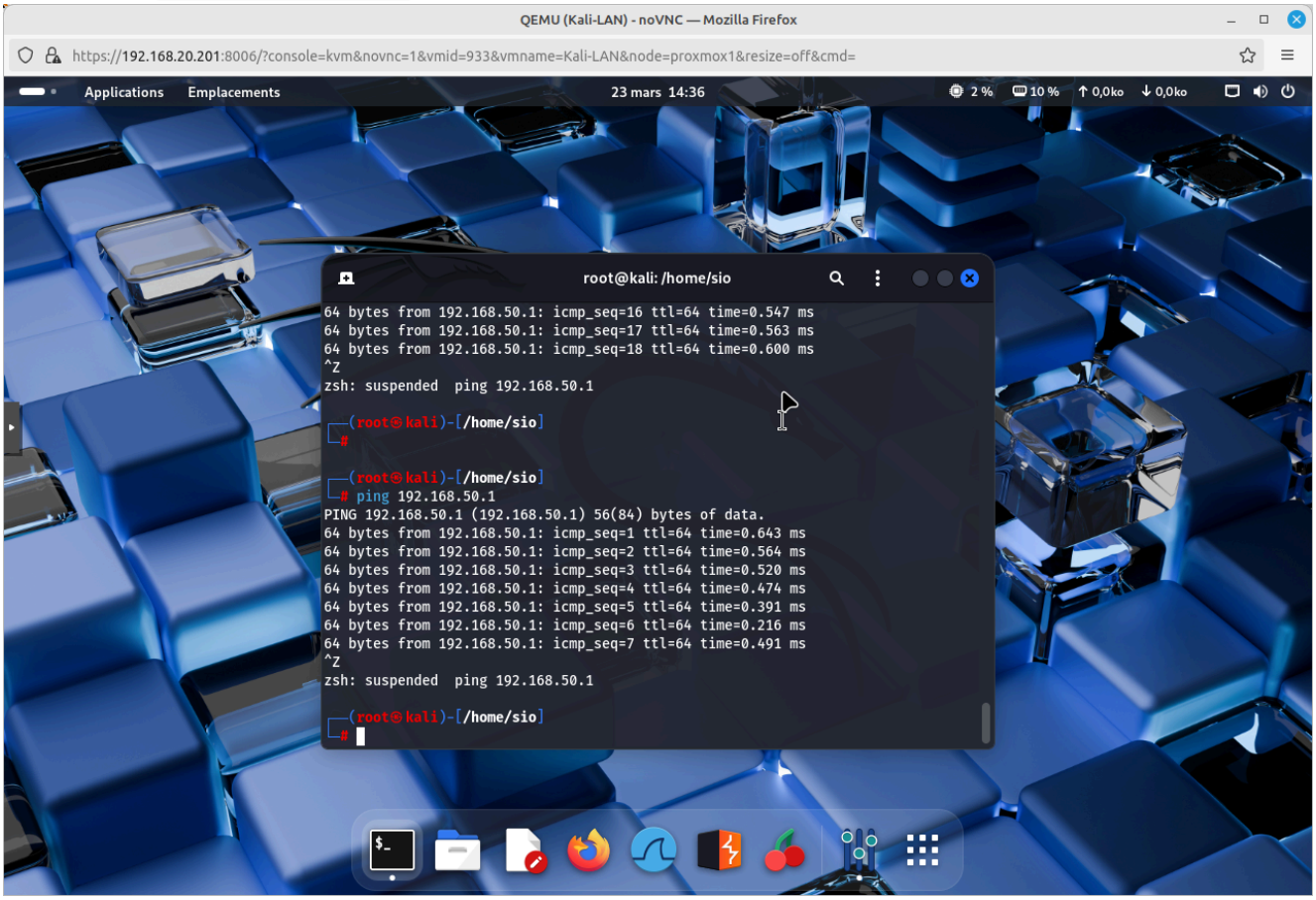


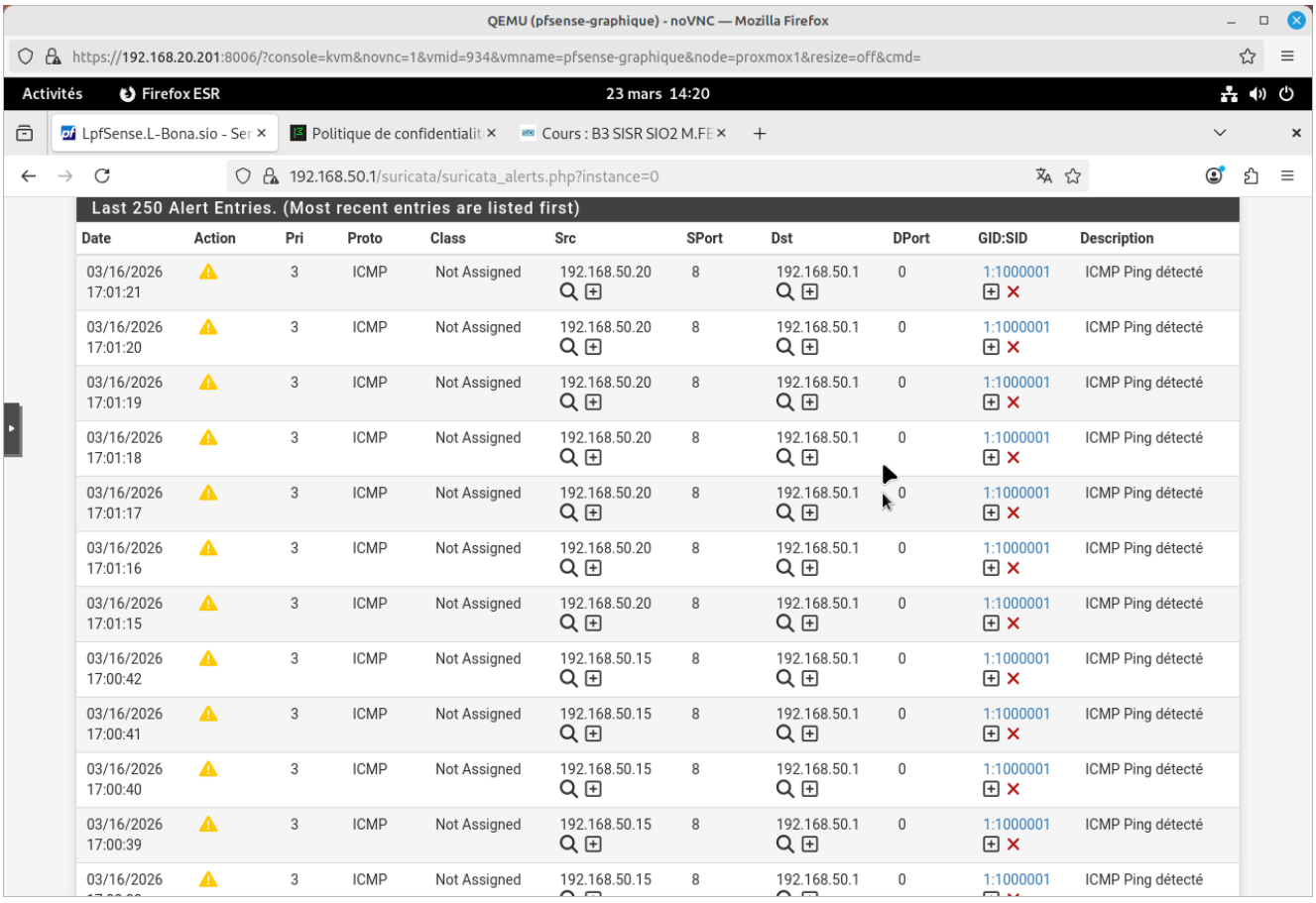
---

6. Simulation d'attaque et Analyse des Logs

C'est le moment de tester si notre vigile est réveillé. Depuis la machine **Kali Linux** :

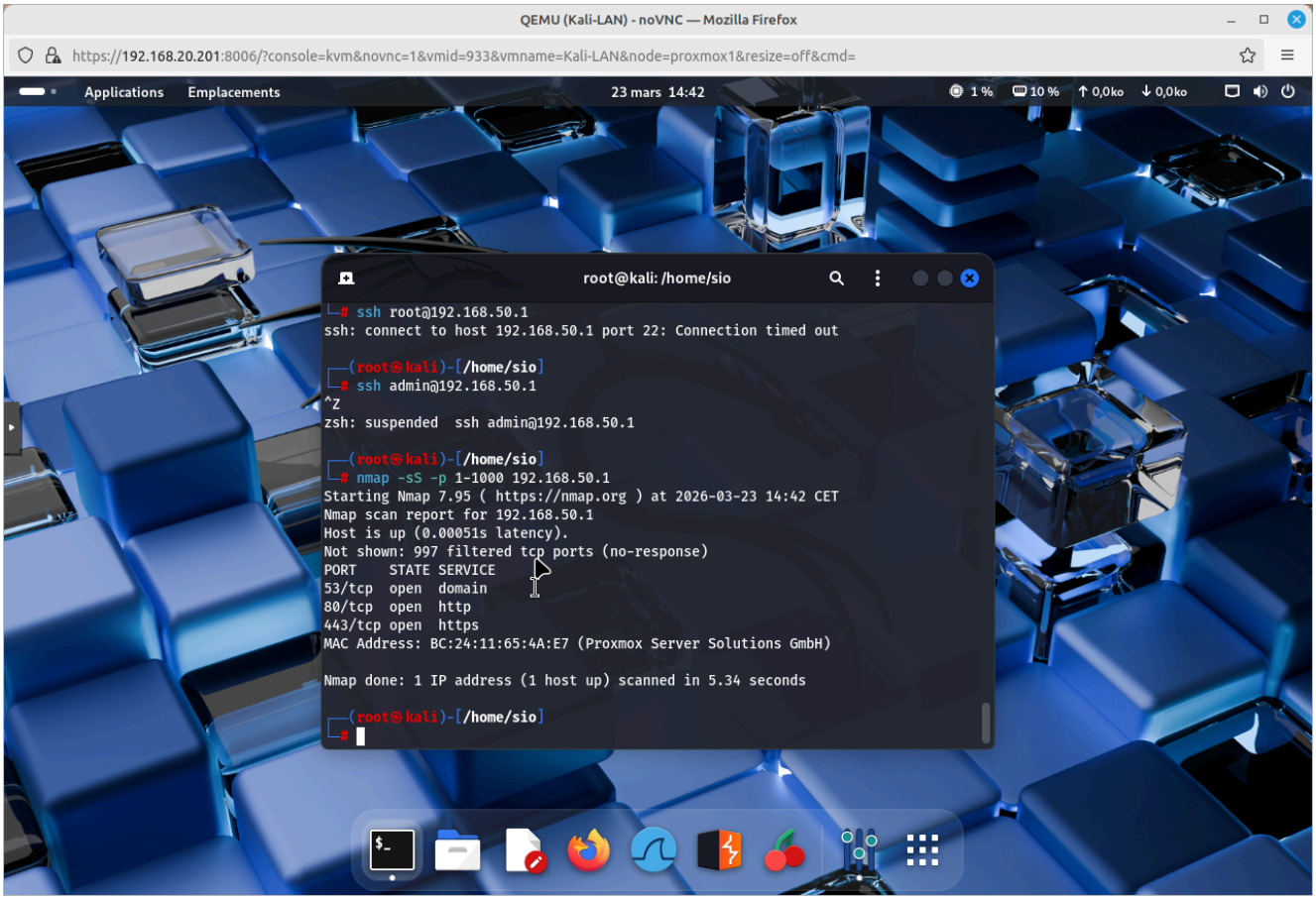
1. **Test Ping :** `ping 192.168.50.1` / **Analyse :** Allez dans pfSense sous **Services > Suricata > Alerts**.

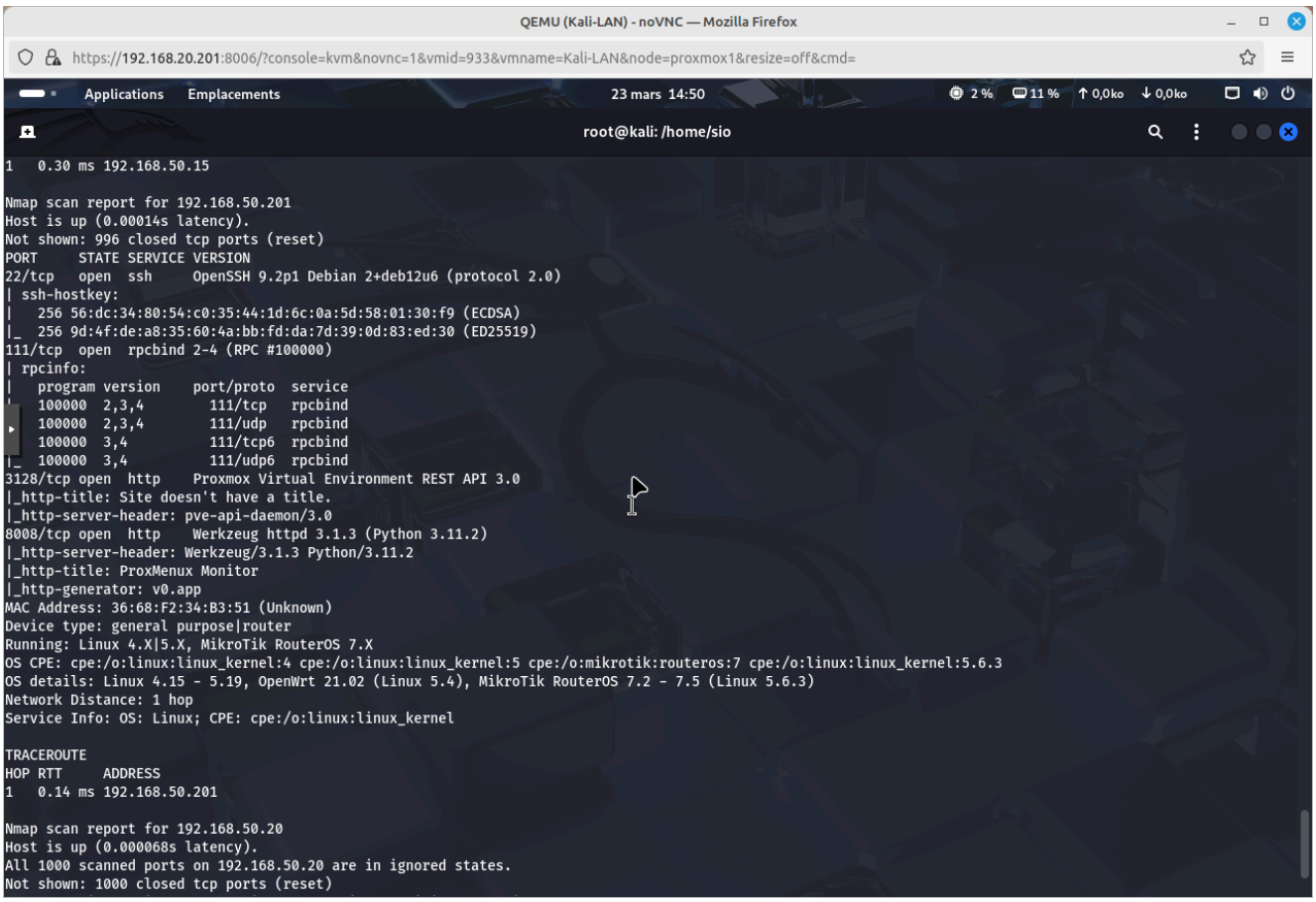


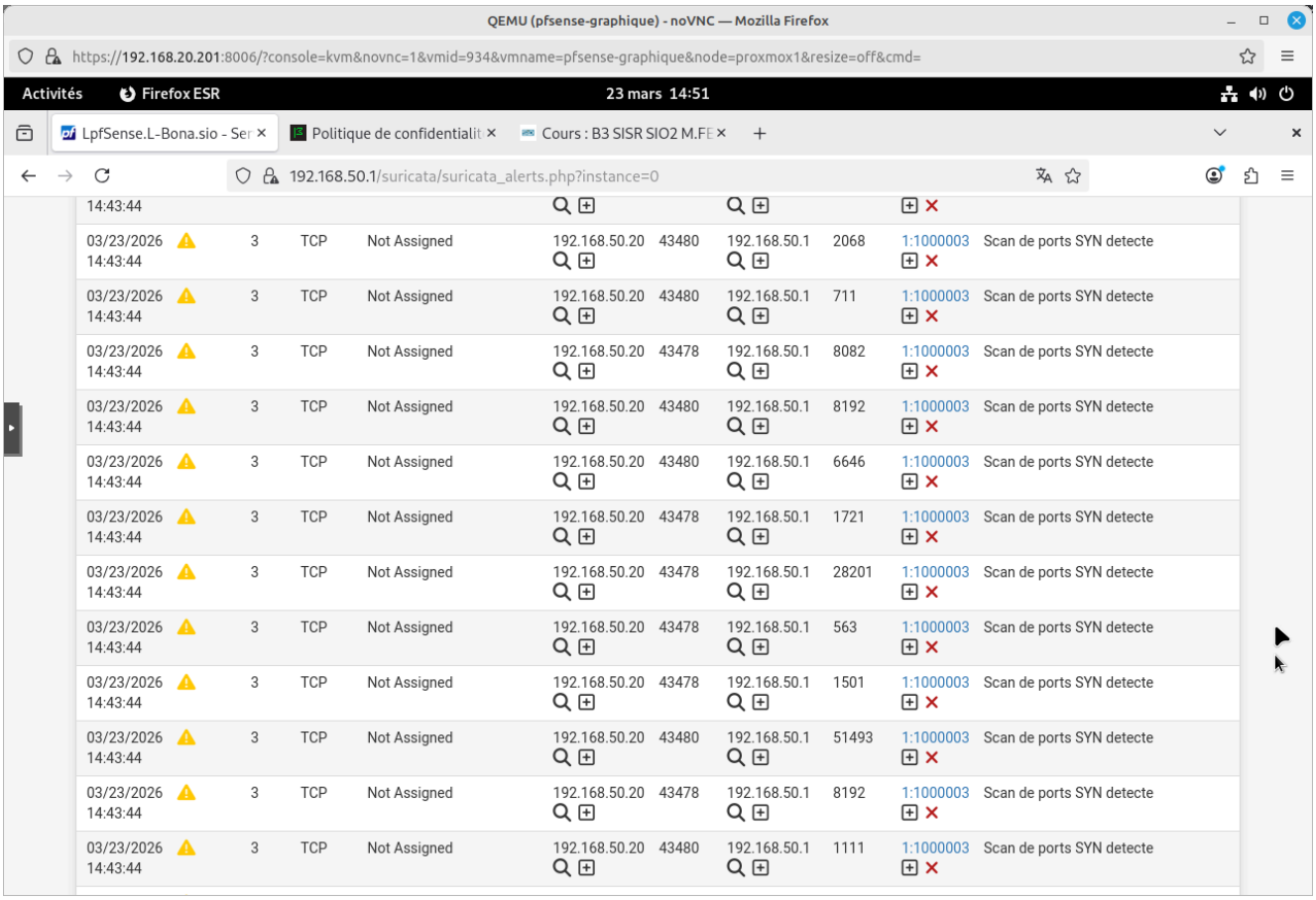


Date	Action	Pri	Proto	Class	Src	SPort	Dst	DPort	GID:SID	Description
03/16/2026 17:01:21	⚠	3	ICMP	Not Assigned	192.168.50.20	8	192.168.50.1	0	1:1000001	ICMP Ping détecté
03/16/2026 17:01:20	⚠	3	ICMP	Not Assigned	192.168.50.20	8	192.168.50.1	0	1:1000001	ICMP Ping détecté
03/16/2026 17:01:19	⚠	3	ICMP	Not Assigned	192.168.50.20	8	192.168.50.1	0	1:1000001	ICMP Ping détecté
03/16/2026 17:01:18	⚠	3	ICMP	Not Assigned	192.168.50.20	8	192.168.50.1	0	1:1000001	ICMP Ping détecté
03/16/2026 17:01:17	⚠	3	ICMP	Not Assigned	192.168.50.20	8	192.168.50.1	0	1:1000001	ICMP Ping détecté
03/16/2026 17:01:16	⚠	3	ICMP	Not Assigned	192.168.50.20	8	192.168.50.1	0	1:1000001	ICMP Ping détecté
03/16/2026 17:01:15	⚠	3	ICMP	Not Assigned	192.168.50.20	8	192.168.50.1	0	1:1000001	ICMP Ping détecté
03/16/2026 17:00:42	⚠	3	ICMP	Not Assigned	192.168.50.15	8	192.168.50.1	0	1:1000001	ICMP Ping détecté
03/16/2026 17:00:41	⚠	3	ICMP	Not Assigned	192.168.50.15	8	192.168.50.1	0	1:1000001	ICMP Ping détecté
03/16/2026 17:00:40	⚠	3	ICMP	Not Assigned	192.168.50.15	8	192.168.50.1	0	1:1000001	ICMP Ping détecté
03/16/2026 17:00:39	⚠	3	ICMP	Not Assigned	192.168.50.15	8	192.168.50.1	0	1:1000001	ICMP Ping détecté
03/16/2026	⚠	3	ICMP	Not Assigned	192.168.50.15	8	192.168.50.1	0	1:1000001	ICMP Ping détecté

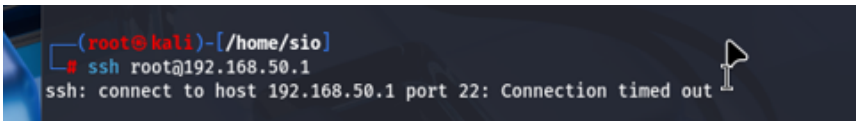
2. **Test Scan de ports :** `nmap -sS 1-1000 192.168.50.1` / **Analyse :** Allez dans pfSense sous **Services > Suricata > Alerts**.

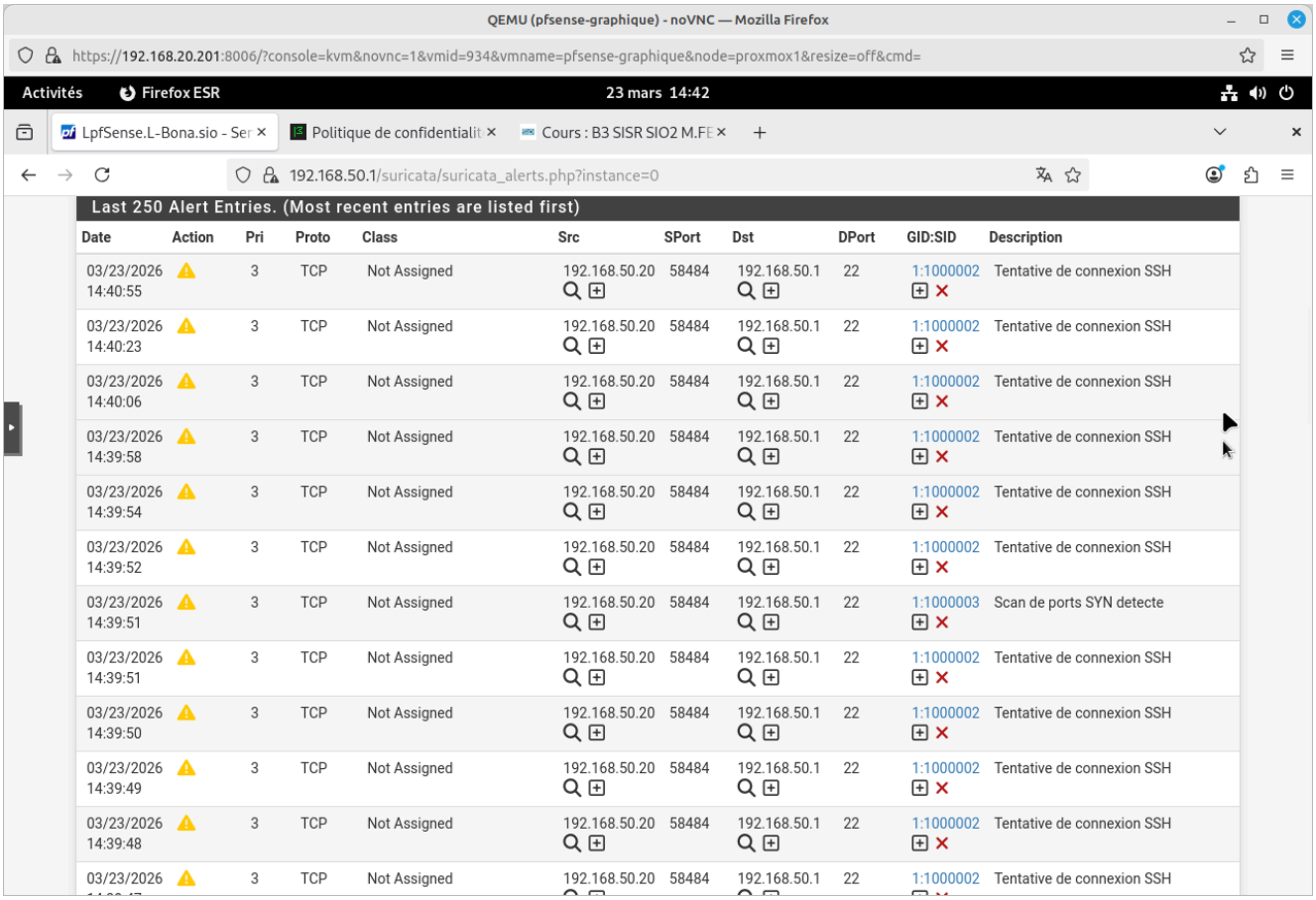

QEMU (Kali-LAN) - noVNC — Mozilla Firefox
https://192.168.20.201:8006/?console=kvm&novnc=1&vmid=933&vmname=Kali-LAN&node=proxmox1&resize=off&cmd=
Applications
Emplacements
23 mars 14:42
1 %
10 %
0,0ko
0,0ko
root@kali: /home/sio
ssh root@192.168.50.1
ssh: connect to host 192.168.50.1 port 22: Connection timed out
(root㉿kali)-[/home/sio]
ssh admin@192.168.50.1
^Z
zsh: suspended ssh admin@192.168.50.1
(root㉿kali)-[/home/sio]
nmap -sS -p 1-1000 192.168.50.1
Starting Nmap 7.95 (https://nmap.org) at 2026-03-23 14:42 CET
Nmap scan report for 192.168.50.1
Host is up (0.00051s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT STATE SERVICE
53/tcp open domain
80/tcp open http
443/tcp open https
MAC Address: BC:24:11:65:4A:E7 (Proxmox Server Solutions GmbH)
Nmap done: 1 IP address (1 host up) scanned in 5.34 seconds
(root㉿kali)-[/home/sio]


QEMU (Kali-LAN) - noVNC — Mozilla Firefox
https://192.168.20.201:8006/?console=kvm&novnc=1&vmid=933&vmname=Kali-LAN&node=proxmox1&resize=off&cmd=
Applications
Emplacements
23 mars 14:50
2 %
11 %
0,0ko
0,0ko
root@kali: /home/sio
1 0.30 ms 192.168.50.15
Nmap scan report for 192.168.50.201
Host is up (0.00014s latency).
Not shown: 996 closed tcp ports (reset)
PORT STATE SERVICE VERSION
22/tcp open ssh OpenSSH 9.2p1 Debian 2+deb12u6 (protocol 2.0)
| ssh-hostkey:
| 256 56:dc:34:80:54:c0:35:44:1d:6c:0a:5d:58:01:30:f9 (ECDSA)
|_ 256 9d:4f:de:a8:35:60:4a:bb:fd:da:7d:39:0d:83:ed:30 (ED25519)
111/tcp open rpcbind 2-4 (RPC #100000)
| rpcinfo:
| program version port/proto service
| 100000 2,3,4 111/tcp rpcbind
| 100000 2,3,4 111/udp rpcbind
| 100000 3,4 111/tcp6 rpcbind
|_ 100000 3,4 111/udp6 rpcbind
3128/tcp open http Proxmox Virtual Environment REST API 3.0
|_http-title: Site doesn't have a title.
|_http-server-header: pve-api-daemon/3.0
8008/tcp open http Werkzeug httpd 3.1.3 (Python 3.11.2)
|_http-server-header: Werkzeug/3.1.3 Python/3.11.2
|_http-title: ProxMenux Monitor
|_http-generator: v0.app
MAC Address: 36:68:F2:34:B3:51 (Unknown)
Device type: general purpose|router
Running: Linux 4.X|5.X, MikroTik RouterOS 7.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5 cpe:/o:mikrotik:routeros:7 cpe:/o:linux:linux_kernel:5.6.3
OS details: Linux 4.15 - 5.19, OpenWrt 21.02 (Linux 5.4), MikroTik RouterOS 7.2 - 7.5 (Linux 5.6.3)
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
TRACEROUTE
HOP RTT ADDRESS
1 0.14 ms 192.168.50.201
Nmap scan report for 192.168.50.20
Host is up (0.000068s latency).
All 1000 scanned ports on 192.168.50.20 are in ignored states.
Not shown: 1000 closed tcp ports (reset)



3. **Test Scan de connexion SSH:** `ssh root@192.168.50.1` / **Analyse :** Allez dans pfSense sous **Services > Suricata > Alerts**.





Comprendre les logs de Suricata :

État dans les logs	Signification
Alert	Une activité suspecte a été vue, mais le trafic est passé.
Priority 1	Alerte critique (tentative d'intrusion).
SID	L'identifiant unique de la règle qui a "matché".

---

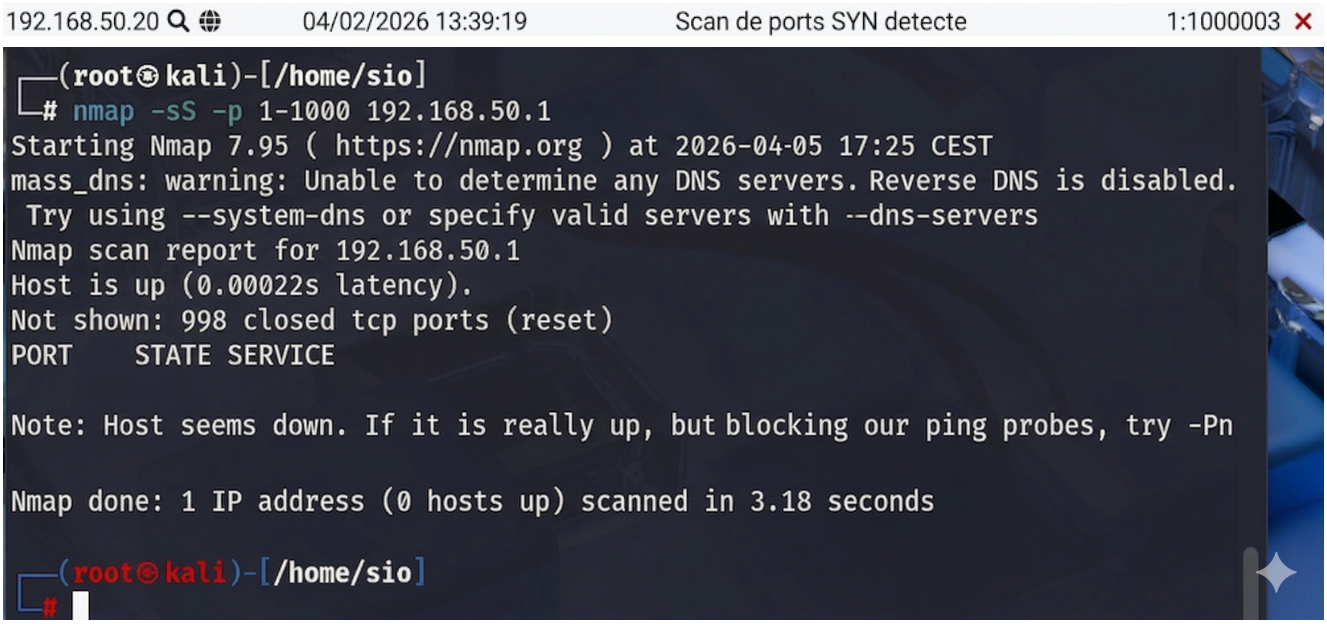
7. Passage en mode IPS (Le blocage)

Pour l'instant, Suricata ne fait que crier "Au voleur !". Pour qu'il arrête le voleur :

1. Dans les paramètres de l'interface LAN, cochez **Block Offenders**.
2. **Mode :** Legacy (par défaut sur pfSense).
3. Dans LAN Rules, modifier la regle du scan en drop au lieu de alert

```
drop tcp any any -> $HOME_NET any (msg:"Scan de ports SYN BLOQUE"; flags:S;threshold:type threshold, track by_
```

4. **Vérification :** Allez dans l'onglet **Blocked Hosts**. L'IP de Kali (192.168.50.20) doit y être listée. Elle ne peut plus communiquer avec le réseau.



---

8. Gestion des Faux Positifs

Si Suricata bloque votre PC d'administration parce qu'il fait trop de requêtes, c'est un **Faux Positif**.

- **Identifier le problème** : Si votre PC Admin génère des alertes à chaque fois qu'il teste la connectivité du serveur, ce n'est pas une attaque, c'est un flux légitime.
- **Action corrective** : Créer une règle `pass` (exception).
  - Va dans **LAN Rules**.
  - Ajoute cette règle **avant** les autres : `pass icmp 192.168.1.10 any -> $HOME_NET any (msg:"Pass ICMP Admin"; itype:8; sid:9000001; rev:1;)`.

```
alert icmp any any -> $HOME_NET any (msg:"ICMP Ping détecté"; itype:8;sid:1000001; rev:1;)

alert tcp any any -> $HOME_NET 22 (msg:"Tentative de connexion SSH"; flags:S;sid:1000002; rev:1;)

drop tcp any any -> $HOME_NET any (msg:"Scan de ports SYN BLOQUE"; flags:S;threshold:type threshold, track by_

alert http any any -> $HOME_NET any (msg:"Mot de passe en clair HTTP";content:"password";nocase; http_uri;flow

alert http any any -> $HOME_NET any (msg:"Scan nmap detecte - User-Agent";content:"Nmap";http_user_agent; sid:

pass icmp 192.168.50.15 any -> $HOME_NET any(msg:"Pass ICMP Admin; itype:8 ; sid:9000001; rev:1;)
```

> **Q10. Rapport d'incident - NetSecure SA**
> Le 2 Avril 2026, a 13h39 et 19 seconde, une série de tentatives d'intrusion a été détectée sur le segment LAN par Suricata. L'attaquant (`192.168.50.20`) a débuté une reconnaissance ICMP et un scan de ports agressif (SID 1000003), ciblant le serveur (`192.168.50.1`). Des tentatives de connexion SSH non autorisées et des requêtes HTTP contenant des identifiants en clair ont également été bloquées. Grâce au passage en mode IPS, l'IP source a été bannie après 3 tentatives de scan infructueuses, protégeant

> ainsi l'intégrité de nos données. **Préconisations :** Désactivation du SSH par mot de passe au profit de clés SSH et mise en place d'un tunnel HTTPS obligatoire pour tous les services web internes.