

ADRAR FORMATION

# Mise en place d'un NIPS avec SIEM



MARAVAL Liam  
28/02/2025

Ce document sera décomposé en plusieurs chapitres :

- 1.Contexte : Définition des besoins ainsi que le choix des solutions en réponse à la demande client.
2. Configuration technique : Procédure technique de mise en place des solutions
3. Conclusion

## Table des matières

1.	Contexte .....	2
1.1	Demandes du client.....	2
1.2	Evolution de l'infrastructure .....	3
1.3	Choix de la solution .....	4
1.4	Sécurisation des accès .....	5
2.	Configuration technique.....	5
2.1	Installation et configuration de Snort .....	5
2.2	Configuration des règles Snort DDoS ICMP.....	8
2.3	Configuration règle Snort Brut Force FTP.....	11
2.4	Configuration Snort en mode NIDS.....	12
2.5	Intégration avec Graylog.....	15
2.6	Configuration des règles « Snort Community » .....	32
3.	Conclusion.....	36
4.	Annexes .....	38

# 1. Contexte

## 1.1 Demandes du client

Nous sommes contactés par l'ADRAR afin de mettre en place une solution pour sécuriser le réseau de potentielles attaques type DDOS, Brut force FTP, ARP Spoofing ect..

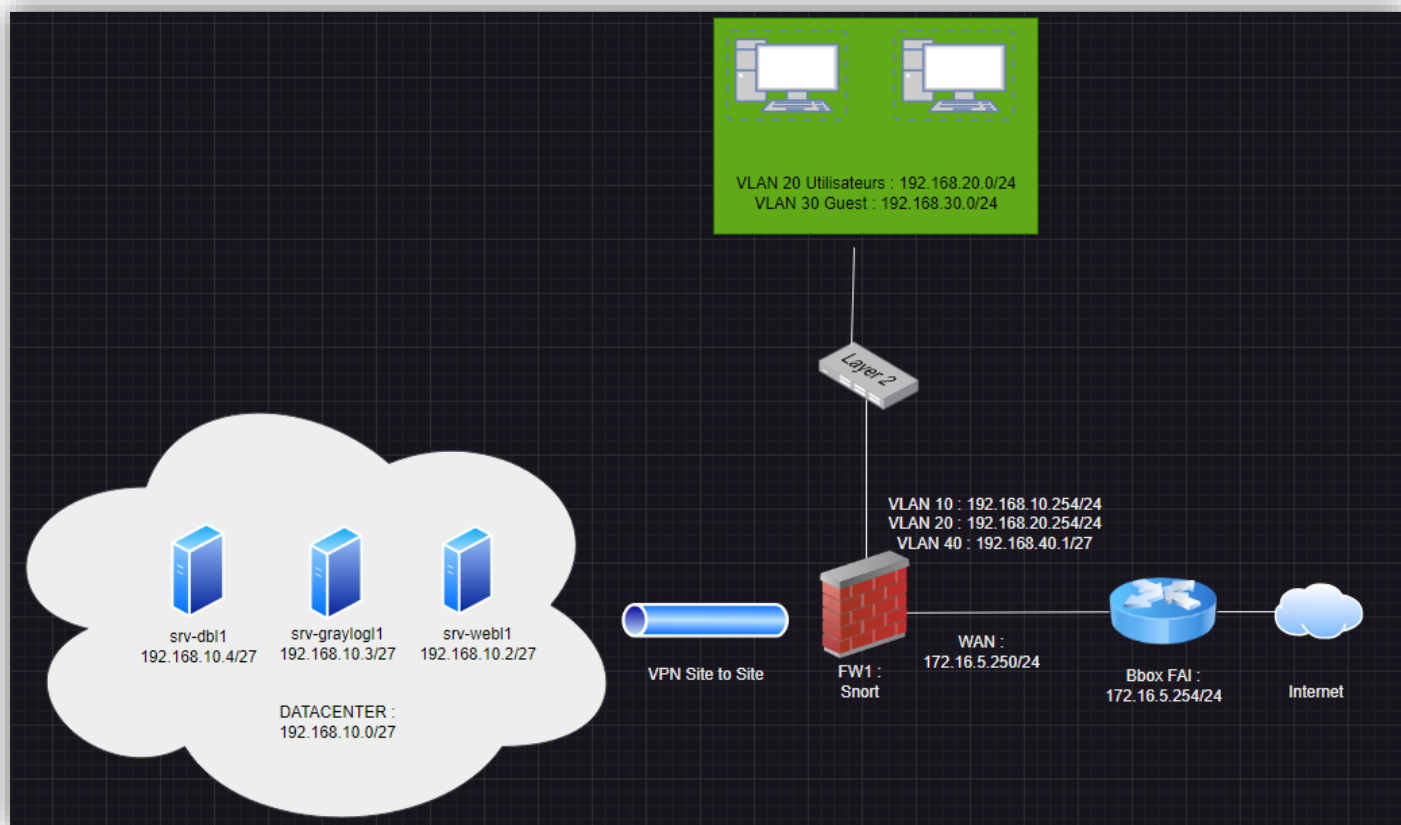
De plus, ils nous est demandés de mettre en place un outil de centralisation des logs afin de faciliter la gestion des informations et événements lié à la sécurité.

Voici le cahier des charges défini avec l'ADRAR :

- Installation d'un service NIDS
- Configuration du service NIDS afin de protéger le réseau d'attaques Brut Force, DDoS ICMP
- Installation et configuration d'un outil de centralisation des logs (SIEM)
- Mise en place des règles communautaires de Snort pour prévenir des attaques les plus courantes

## 1.2 Evolution de l'infrastructure

Pour mieux comprendre la mise en place de la nouvelle infrastructure, veuillez-vous référer au nouveau schéma effectué :



Afin de répondre à la demande de l'ADRAR, nous allons donc implémenter « **Snort** » sur le Pfsense existant.

Ces services auront la charge de détecter le trafic suspect et de générer des alertes.

Nous configurons élégamment Snort en mode NIPS afin qu'il puisse bloquer le trafic en plus de remonter les alertes, cependant cette pratique peut être inadaptée car il pourrait bloquer du trafic légitime.

C'est pour cela que dès la fin du POC NIPS l'option sera désactivée, mais pourra être réactivée à l'avenir si l'ADRAR le souhaite en suivant la même procédure.

De plus, nous allons mettre en place un serveur Graylog, celui-ci servira afin de récupérer les logs du service **Snort** ainsi que du pare-feu Pfsense plus généralement afin de les centraliser dans sa console.

Voici ce qui va être mis en place :

- Intégration avec le pare-feu et avec Snort
- Configuration d'une politique de rétention des logs (Plus de détail dans la partie **1.4 Sécurisation des accès**)
- Configuration d'un stream afin de filtrer et afficher uniquement les logs du service Snort
- Configuration d'alertes afin de remonter les événements liés à la détection de trafic non légitime par Snort.

- Configuration d'envoi de mail lors du déclenchement de l'alerte.

Dans le cadre de notre POC, nous avons choisi de surveiller l'interface LAN à des fins des tests, cependant Snort est configurable sur n'importe quelle interface du pare-feu.

Une fois notre POC validé dans l'ensemble, nous mettrons en place les « **Community rules** » de Snort qui regroupe toutes les règles créées par la communauté afin de remonter les alertes des attaques les plus courantes (toutes les règles ne seront pas activées nous sélectionnerons uniquement celles voulues).

### 1.3 Choix de la solution

Nous avons opté pour **Snort** intégré à Pfsense afin d'assurer le service NIPS, voici les raisons qui ont motivé notre choix :

- Intégration complète dans Pfsense
- Interface graphique intuitive
- Positionnement stratégique car permet de surveiller toutes les pattes de nos réseaux
- Leger et optimisé
- Beaucoup de règles communautaires disponibles gratuitement

En ce qui concerne Graylog, voici pourquoi nous l'avons choisi :

- Centralisation des logs en temps réel
- Interface web intuitive pour l'analyse et la visualisation
- Recherche avancée et filtrage puissant des logs
- Alertes et tableaux de bord personnalisables
- Compatibilité avec de nombreux formats de logs (Syslog, JSON, GELF...)
- Extensible grâce aux plugins et intégrations tierces
- Gestion efficace de grande quantité de données

Concernant les couts, nos 2 solutions sont open source.

Une version « **business** » est disponible pour **Snort** afin d'accéder à un support en priorité, un accès à des règles anticipé ect...

Une version « **entreprise** » de **Graylog** est disponible qui permet d'accéder à un support premium ainsi que des fonctionnalités supplémentaires comme des alertes via teams, discord..., une intégration avec O365, Azure, AWS, Palo Alto Networking, des modules avancés de gestion ect...

Une version « **sécurité** » est aussi disponible qui ajoute des fonctionnalités intéressantes. Voici les principales :

- Une détection des menaces avancées avec les réponses associées
- Détection d'anomalie via des événements inhabituels
- Enrichies les journaux d'événement avec une des informations complémentaires recueillis via une base de données des menaces connues
- Intégration du framework MITRE ATT&CK afin d'aider à identifier les techniques utilisées par des attaquants.

La version business de Snort ne présente pas de réels avantages à ce jour pour l'ADRAR. Cependant, la version de Security de Graylog offre un vrai avantage à notre sens.

Voici les couts pour chaque version :

- Snort business : 399\$/an et par capteur Snort
- Graylog Security : 1550\$/mois

Concernant le temps de déploiement de la solution il est estimé à quatre jours.

## 1.4 Sécurisation des accès

Afin de sécuriser notre solution, nous allons mettre en place les configurations suivantes :

- Modification des ports par défaut de Snort et Graylog
- Compte d'administration de Graylog dédié
- Règle Snort afin de détecter de potentielle attaques DDOS ICMP Flood
- Règle Snort afin de détecter de potentielle attaque Brut Force du serveur FTP.
- Règles communautaires pour détecter les scans réseau via NMAP.

Conformément aux recommandations de la CNIL, une politique de rétention des logs sur le SIEM sera configurée en suivant ces règles :

- 30 jours de conservation des logs avant archive
- Archive conservée pendant 1 an avant suppression

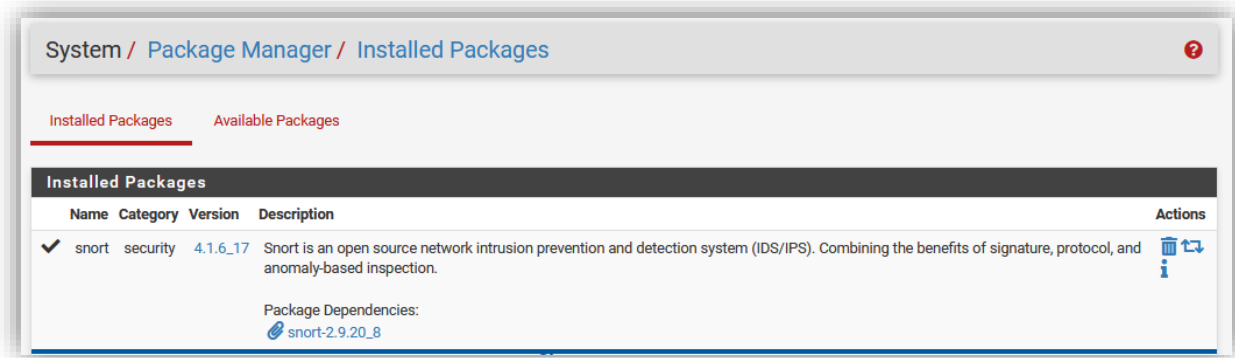
## 2. Configuration technique

Afin de faciliter la compréhension de cette procédure voici quelques termes à connaitre :

- **NIDS** : Système qui permet de détecter en temps réel de potentielles attaques réseau
- **NIPS** : Système qui permet de détecter et bloquer en temps réel de potentielles attaques réseau
- **SIEM** : Système qui permet de collecter, analyser et centraliser les données de sécurité pour détecter et répondre aux incidents
- **DDoS** : Une attaque qui consiste à inonder un serveur de requête (ICMP, HTTP, SYN ect...) depuis plusieurs sources afin de le rendre indisponible.
- **Brut Force** : Attaque qui consiste à essayer de deviner un mot de passe ou une clé en essayant un grand nombre de combinaisons possible automatiquement
- **Extractor** : Dans Graylog un extractor permet d'extraire et structurer des informations spécifiques à partir de logs bruts pour une meilleure analyse.

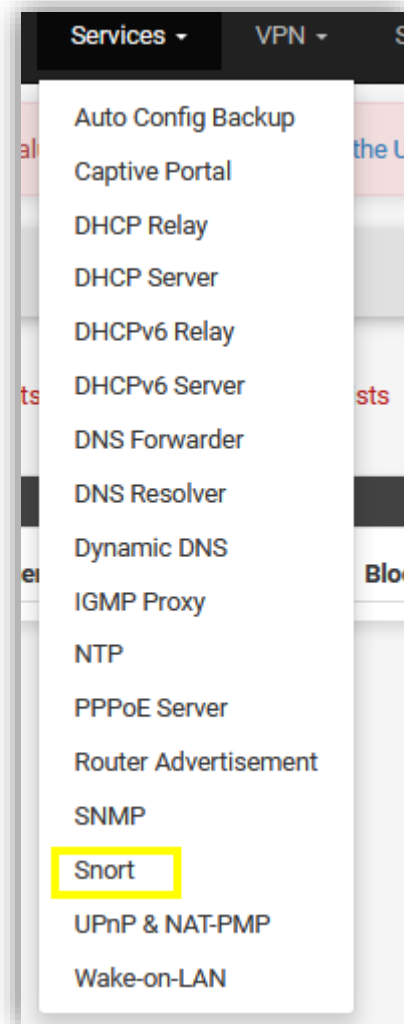
### 2.1 Installation et configuration de Snort

Pour l'installation du service Snort, il faut se rendre dans le package manager de Pfsense :

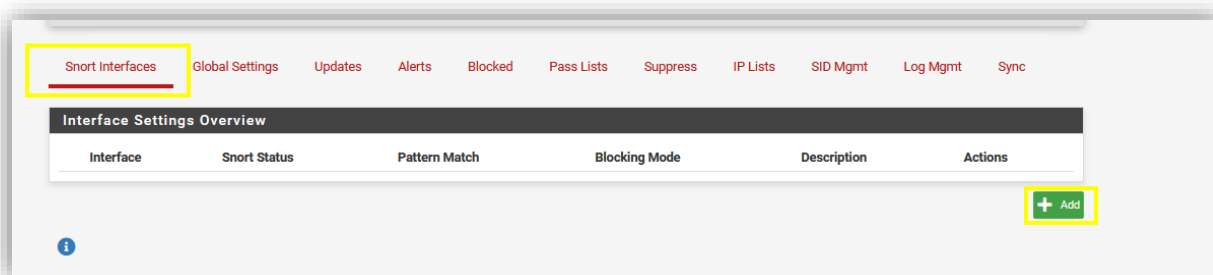


Une fois cela fait, nous allons commencer la configuration.

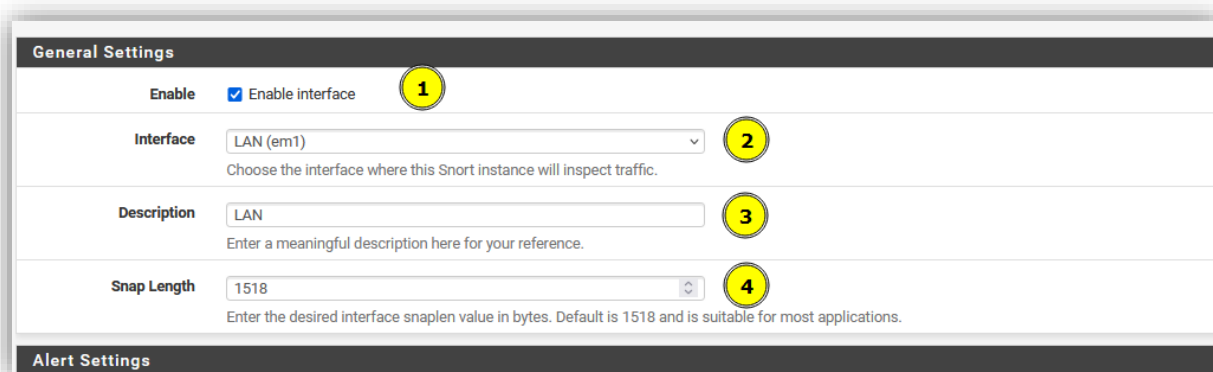
Nous allons dans le menu « **Services -> Snort** » :



Ensuite nous dans le menu « **Snort** » interface nous allons cliquer sur « **Add** » afin d'ajouter et configurer notre service sur l'interfaces de notre pare feu que nous souhaitons :

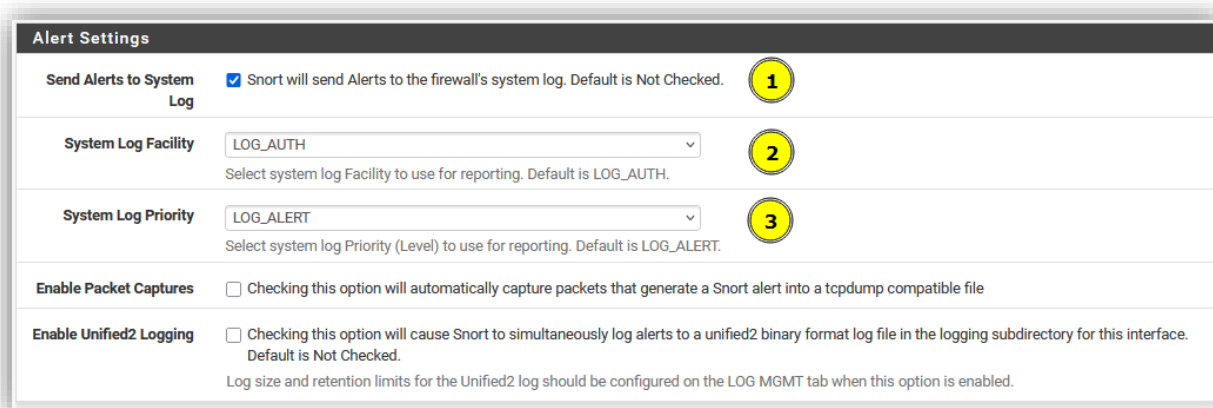


Dans « **General Settings** » nous configurons :



1. Active l'interface
2. Interface d'écoute
3. Description
4. Taille maximal des paquets analysés

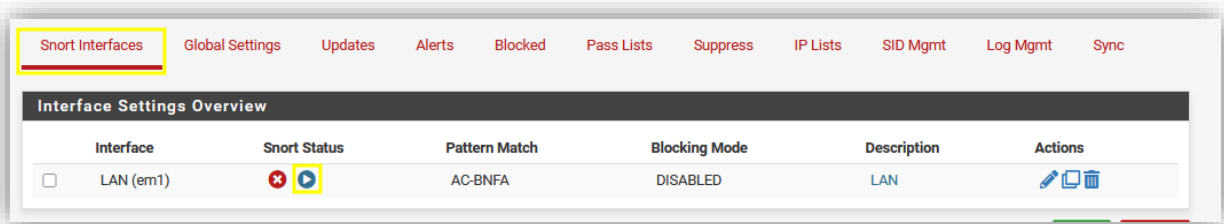
Dans « **Alert Setting** » :



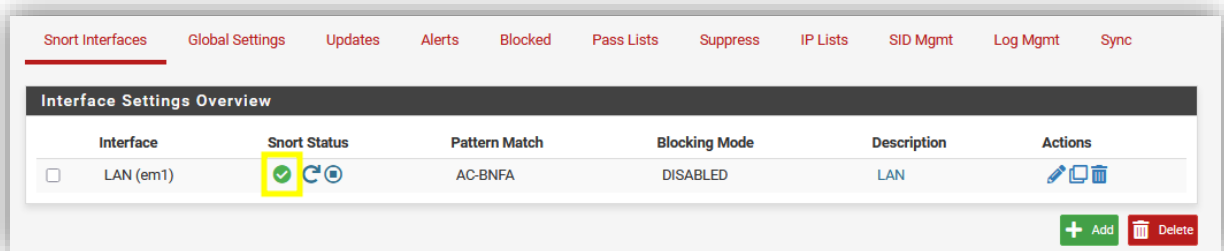
1. Envoyer les alertes dans les logs systèmes du pare feu
2. Envoie des logs de type sécurité et authentification
3. Envoie les logs de type alertes



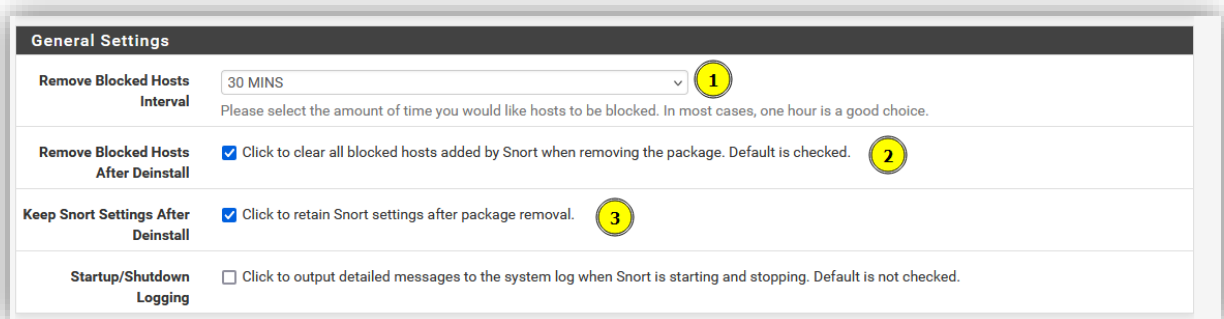
Nous sauvegardons ensuite la configuration et lançons le service :



Snort est désormais actif :



Nous allons maintenant dans le menu « **Snort -> Global Settings** » nous allons renseigner les paramètres suivants :

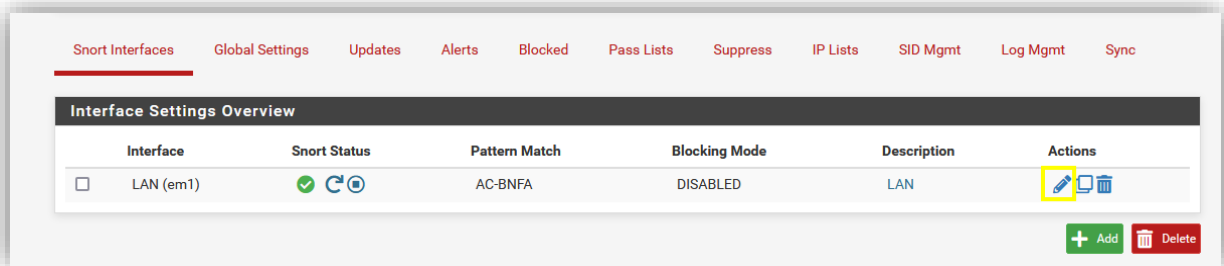


1. Vide la liste des hôtes bloqués toutes les 30mins (sera utile pour la partie NIPS que nous configurerons plus tard dans la procédure)
2. Vide la liste des hôtes bloqués après désinstallation du service
3. Conserve les paramètres de Snort après désinstallation

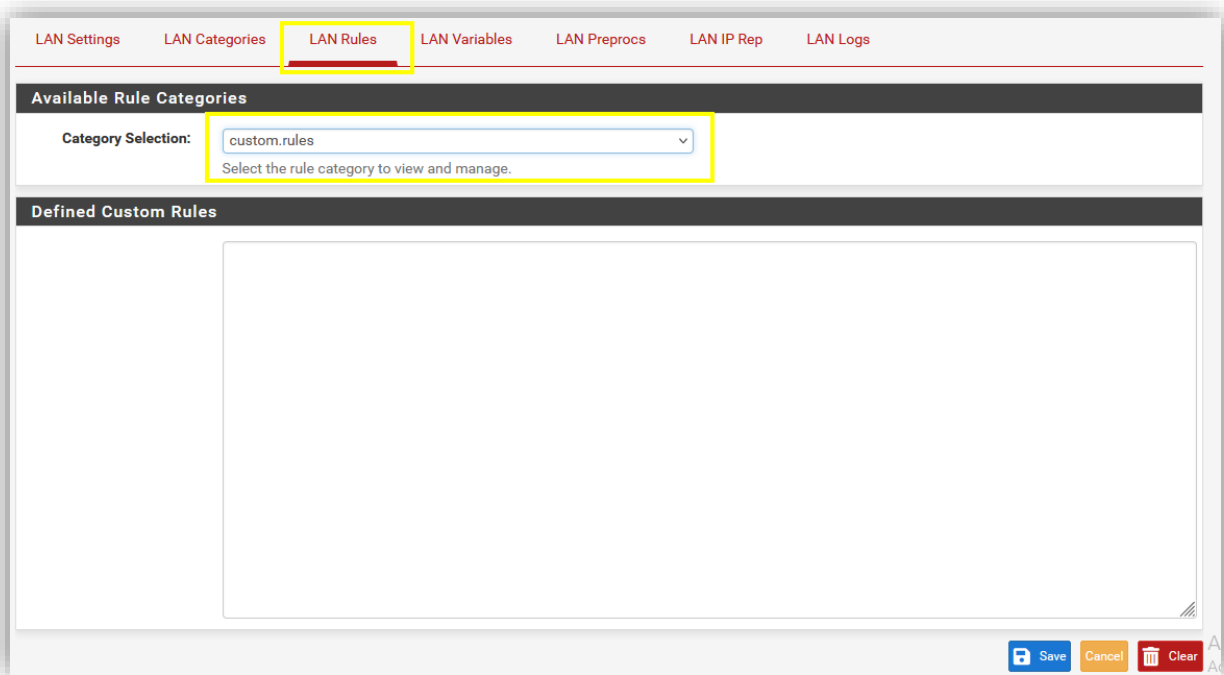
Maintenant que Snort est configuré nous allons pouvoir commencer la création de nos règles personnalisées.

## 2.2 Configuration des règles Snort DDoS ICMP

Pour configurer nos règles nous allons éditer notre interface créée précédemment :



Puis dans « **LAN Rules** » nous allons dans « **custom rules** » afin d'écrire nos règles personnalisées



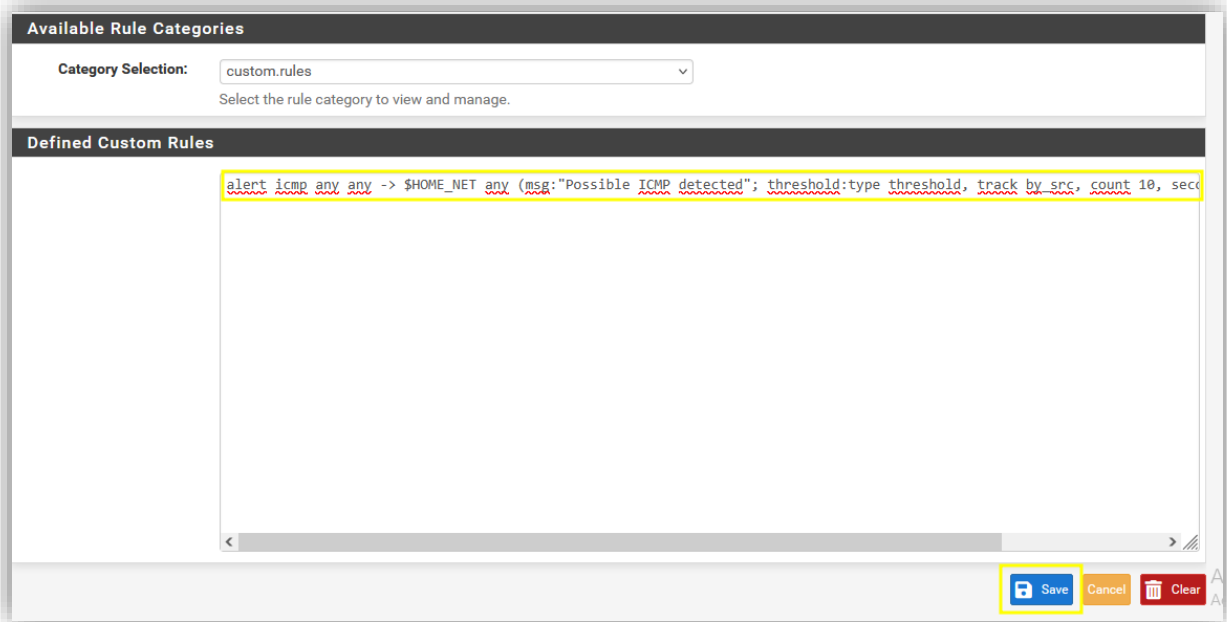
Voici la règle que nous avons mise en place :

```
alert icmp any any -> $HOME_NET any (msg:"Possible DDoS ICMP detected";
threshold:type threshold, track by src, count 100, seconds 1; sid:1000002;)
```

Nous allons l'analyser en détail :

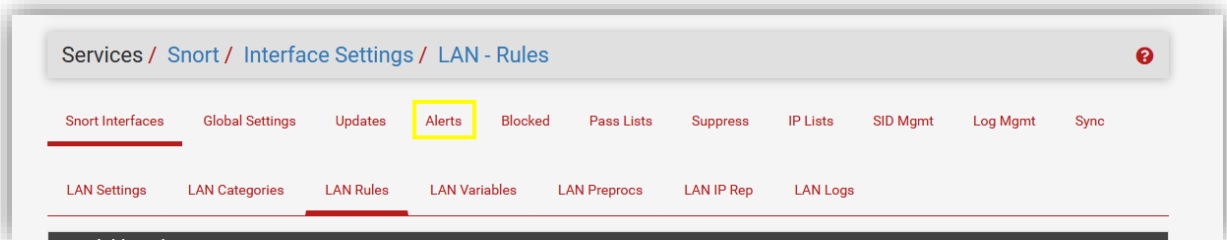
- « **Alert** » : crée une alerte dès le match de la règle
- « **Icmp** » : définit le protocole ICMP
- « **Any any** » : IP et port sources
- « **-> \$HOME\_NET any** » : flux en direction de la variable \$HOME\_NET qui regroupe nos LAN vers n'importe quels ports
- « **Msg** » : Message à afficher
- « **threshold:type threshold, track by\_src, count 100, seconds 1** » : Définit le nombre de requêtes à 100 en l'intervalle de 1 second par source pour considérer le trafic comme non légitime et générer l'alerte
- « **sid:1000002** » : Identifiant de la règle

Une fois celle-ci rédigée, nous sauvegardons la configuration :











Maintenant nous allons tester notre règle via un Kali Linux en mettant en place une simulation de DDoS ICMP.

Une fois cela fait, nous allons nous rendre dans le menu « **Alerts** » pour voir les logs remonter :



Nous voyons donc plusieurs informations :

Most Recent 250 Entries from Active Log										
Date	Action	Pri	Proto	Class	Source IP	SPort	Destination IP	DPort	GID:SID	Description
2025-02-13 13:29:25		0	ICMP		192.168.10.50		192.168.10.2		1:1000002	Possible ICMP detected
2025-02-13 13:29:25		0	ICMP		192.168.10.2		192.168.10.50		1:1000002	Possible ICMP detected
2025-02-13 13:29:25		0	ICMP		192.168.10.50		192.168.10.2		1:1000002	Possible ICMP detected
2025-02-13 13:29:25		0	ICMP		192.168.10.2		192.168.10.50		1:1000002	Possible ICMP detected
2025-02-13 13:29:25		0	ICMP		192.168.10.50		192.168.10.2		1:1000002	Possible ICMP detected
2025-02-13 13:29:25		0	ICMP		192.168.10.2		192.168.10.50		1:1000002	Possible ICMP detected
2025-02-13 13:29:25		0	ICMP		192.168.10.50		192.168.10.2		1:1000002	Possible ICMP detected
2025-02-13 13:29:25		0	ICMP		192.168.10.2		192.168.10.50		1:1000002	Possible ICMP detected

- La date et l'heure
- Le protocole
- L'IP source
- L'IP de destination
- Le SID de la règle
- Le message défini précédemment

Notre règle est donc fonctionnelle, nous allons pouvoir passer à la règle pour le Brut Force du FTP

### 2.3 Configuration règle Snort Brut Force FTP

Dans la même logique que précédemment nous allons créer notre règle pour prévenir un éventuel Brut Force sur le serveur FTP.

Voici la règle que nous allons mettre en place :

```
alert tcp $HOME_NET 21 -> any any (msg:"Potential FTP brute force attack";
flow:established,from_server; content:"530 "; threshold:type threshold, track
by_src, count 5, seconds 60; sid:1000003;)
```

Le début de la règle suit la même logique que pour le DDoS ICMP cependant quelques éléments changent :

- « **flow:established,from\_server; content:"530 "** » : Analyse les réponse du serveur FTP contenant le code 530 qui renvoi un message « login/password incorrect »
- « **threshold:type threshold, track by\_src, count 5, seconds 60;** » : Définie le seuil à 5 tentatives de connexion échouées en 60 secondes par hôte pour générer une alerte

Afin de s'assurer du bon fonctionnement de notre règle nous allons simuler une attaque brut force par dictionnaire via notre Kali Linux et ensuite nous analysons les logs :

Most Recent 250 Entries from Active Log										
Date	Action	Pri	Proto	Class	Source IP	SPort	Destination IP	DPort	GID:SID	Description
2025-02-14 10:28:33		0	TCP		192.168.10.2	21	192.168.10.50	34422	1:1000003	Potential FTP brute force attack
2025-02-14 10:28:33		0	TCP		192.168.10.2	21	192.168.10.50	34520	1:1000003	Potential FTP brute force attack

Ci-dessus nous retrouvons :

- Les IP source et destination
- Les ports sources et destination
- L'heure, le numéro de la règle et le message défini dans la règle.

## 2.4 Configuration Snort en mode NIPS

Comme vu ci-dessus, il est possible de configurer Snort afin de détecter mais également bloquer le trafic suspect.

Nous allons donc nous rendre dans le menu « **Snort interfaces** » et nous éditons notre interface.

Dans le menu « **Global Settings** » nous allons appliquer les paramétrages suivants :

Block Settings

Block Offenders

☒ Checking this option will automatically block hosts that generate a Snort alert. Default is Not Checked.

IPS Mode

Inline Mode

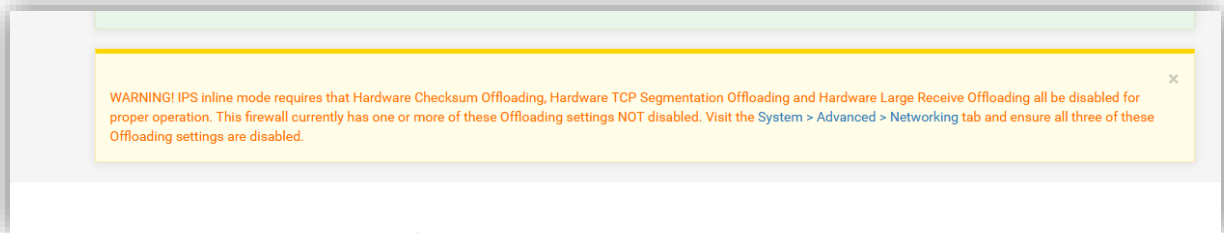
Select blocking mode operation. Legacy Mode inspects copies of packets while Inline Mode inserts the Snort inspection engine into the network stack between the NIC and the OS. Default is Legacy Mode.

Legacy Mode uses the PCAP engine to generate copies of packets for inspection as they traverse the interface. Some "leakage" of packets will occur before Snort can determine if the traffic matches a rule and should be blocked. Inline mode instead intercepts and inspects packets before they are handed off to the host network stack for further processing. Packets matching DROP rules are simply discarded (dropped) and not passed to the host network stack. No leakage of packets occurs with Inline Mode. WARNING: Inline Mode only works with NIC drivers which properly support Netmap! Supported drivers: bnxt, cc, cxgbe, cxl, em, em, ena, ice, igb, igc, ix, ixgbe, ixl, lem, re, vmx, vtnet. If problems are experienced with Inline Mode, switch to Legacy Mode instead.

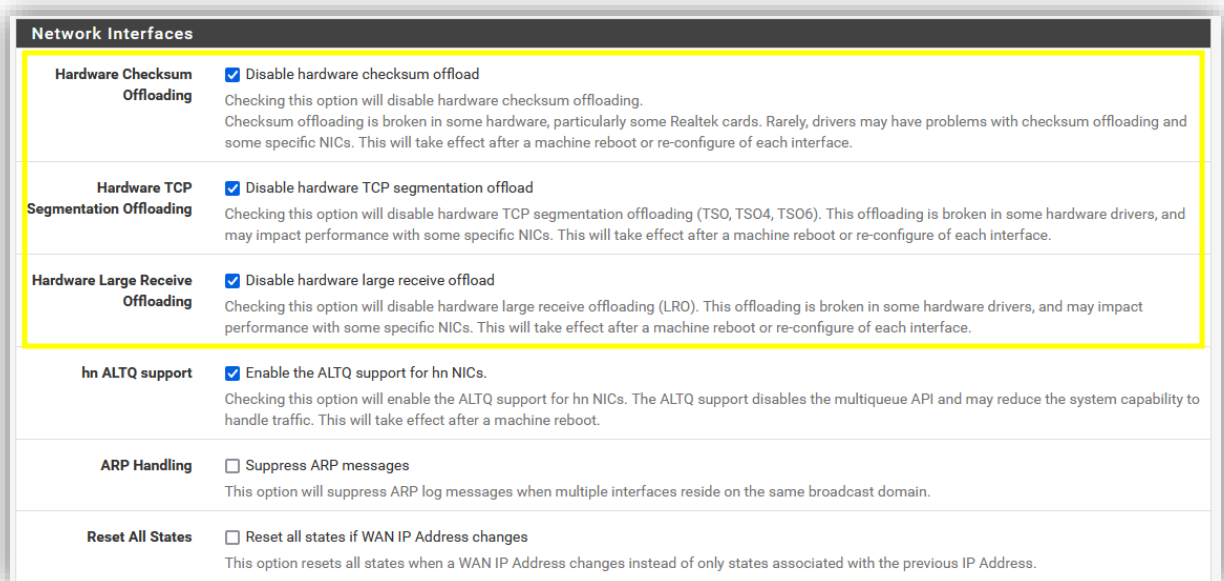
1. Cocher pour activer le mode NIPS
2. Mode d'inspection des paquets réseau en direct sans créer de copies de paquets.  
Contrairement au mode legacy qui crée des copies de paquets et qui peut permettre au trafic malveillant d'atteindre le réseau

**Attention : Quelques configurations sont nécessaires pour le fonctionnement en mode IPS dans notre LAB.**

Dans un 1<sup>er</sup> temps nous avons ce message qui apparaît :



Nous allons donc désactiver les options suivantes dans le menu « **System -> Advanced -> Networking** » :



Attention : Comme nous allons tester notre NIPS depuis notre LAN avec une machine Kali Linux, il est nécessaire de créer une passlist personnalisée car celle par défaut whitelist les hôtes de nos LAN, donc notre machine générant du trafic malveillant ne sera jamais bloqué car elle en fait partie. Pour cela il faut se rendre dans le menu « **Snort -> PassList** » et cliquer sur « **Add** » :

**Pass Lists**

**General Information**

Name:   
The list name may only consist of the characters 'a-z, A-Z, 0-9 and \_'.

Description:   
You may enter a description here for your reference.

**Auto-Generated IP Addresses**

**Local Networks** ☐ Add firewall Locally-Attached Networks to the list (excluding WAN). Default is Checked.

**WAN Gateways** ☒ Add WAN Gateways to the list. Default is Checked.

**WAN DNS Servers** ☒ Add WAN DNS servers to the list. Default is Checked.

**Virtual IP Addresses** ☒ Add Virtual IP Addresses to the list. Default is Checked.

**VPN Addresses** ☒ Add VPN Addresses to the list. Default is Checked.

**Custom IP Addresses and Configured Firewall Aliases**

Hint: Enter as many IP addresses or alias names as desired. Enter ONLY an IP address, IP subnet or alias name! Do NOT enter a FQDN (fully qualified domain name) directly! To use a FQDN, first create the necessary firewall alias, and then provide the alias name here. FQDN aliases are periodically re-resolved and updated by the firewall. You can also provide an IP subnet with a proper netmask of the form network/mask such as 1.2.3.0/24.

IP or Alias:  [Delete](#)

[Save](#) [+ Add IP](#)

Ici nous décochons donc « **Local Networks** » pour enlever le LAN de la whitelist.

Enfin dans le menu « **Snort Interfaces -> LAN Settings** » nous sélectionnons notre nouvelle Pass List :

**Choose the Networks Snort Should Inspect and Whitelist**

**Home Net**  [View List](#)  
Choose the Home Net you want this interface to use.  
Default Home Net adds only local networks, WAN IPs, Gateways, VPNs and VIPs.  
Create an Alias to hold a list of friendly IPs that the firewall cannot see or to customize the default Home Net.

**External Net**  [View List](#)  
Choose the External Net you want this interface to use.  
External Net is networks that are not Home Net. Most users should leave this setting at default.  
Create a Pass List and add an Alias to it, and then assign the Pass List here for custom External Net settings.

**Pass List**  [View List](#)  
Choose the Pass List you want this interface to use.  
The default Pass List adds local networks, WAN IPs, Gateways, VPNs and VIPs. Create an Alias to customize.  
This option will only be used when block offenders is on and IPS Mode is set to Legacy Mode.

**Choose a Suppression or Filtering List (Optional)**

**Alert Suppression and Filtering**  [View List](#)  
Choose the suppression or filtering file you want this interface to use.

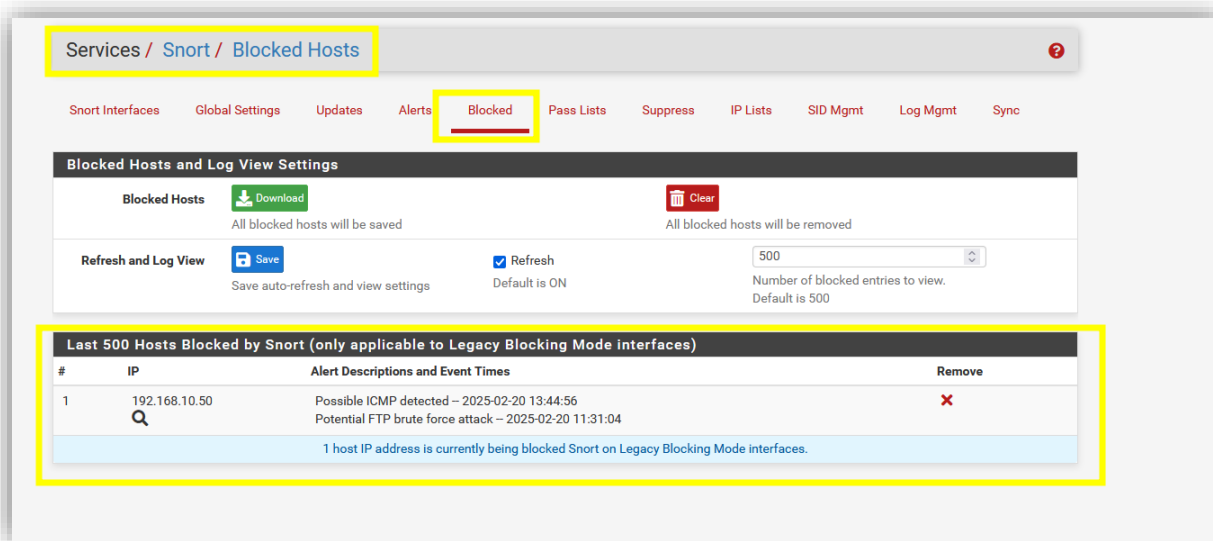
**Custom Configuration Options**

**Advanced Configuration Pass-Through**

Enter any additional configuration parameters to add to the Snort configuration here, separated by a newline

[Save](#)

Maintenant nous pouvons retenter notre attaque, puis dans le menu « **Snort -> Blocked** » nous retrouvons notre hôte bloqué :



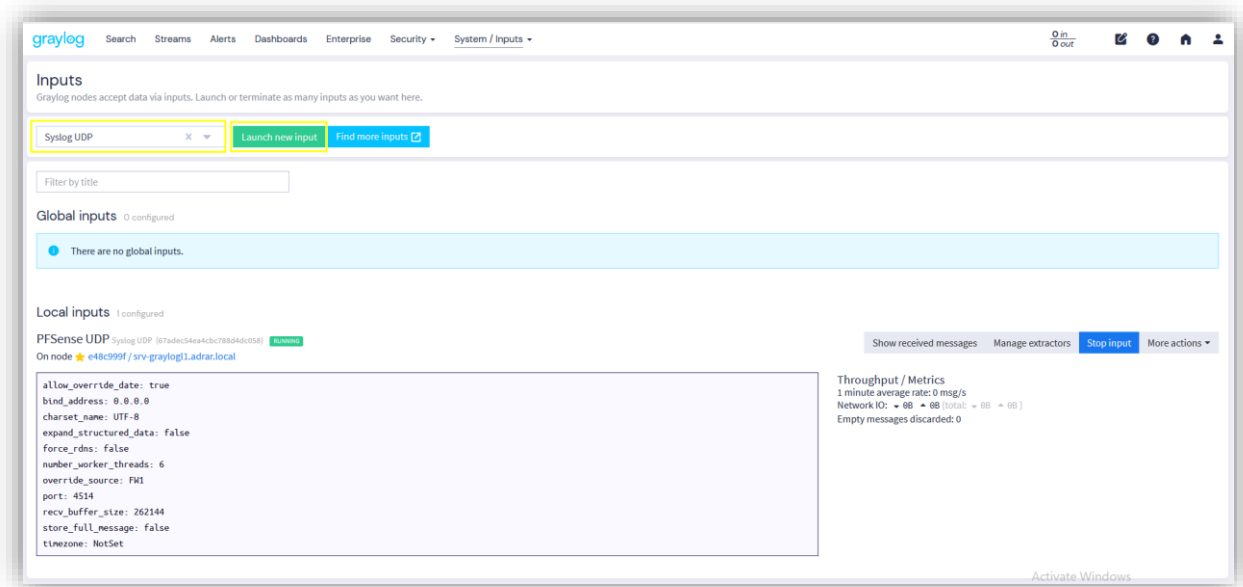
Pour rappel, ce mode peut potentiellement bloquer du trafic légitime c'est pour cela que nous avons décidé de désactiver ce mode.

## 2.5 Intégration avec Graylog

Nous allons désormais intégrer nos logs dans notre solution SIEM qui est « **Graylog** ». L'installation est considérée comme faite, la documentation officielle d'installation est disponible dans la partie Annexes de ce document.

Nous allons nous rendre sur notre serveur afin de configurer une entrée « **input** » pour que Graylog puisse recevoir les logs venant du Pfsense. Pour cela nous allons dans le menu « **System / Inputs -> Inputs** » et sélectionnons « **Syslog UDP** » et « **Launch New Inputs** » :





Ensuite nous allons renseigner les paramètres suivants :

Launch new *Syslog UDP* input

☐ Global  
Should this input start on all nodes

**Node**  

e48c999f / srv-graylog1.adrar.local 1

On which node should this input start

**Title**  

PFSense UDP 2

**Bind address**  

0.0.0.0 3

Address to listen on. For example 0.0.0.0 or 127.0.0.1.

**Port**  

4514 4

Port to listen on.

**Receive Buffer Size (optional)**  

262144 5

The size in bytes of the `recvBufferSize` for network connections to this input.

**No. of worker threads (optional)**  

2 6

Number of worker threads processing network connections for this input.

1. Nom du serveur Graylog
2. Nom du connecteur
3. Adresse d'écoute
4. Port d'écoute
5. Taille du buffer de réceptions des logs
6. Nombre de worker pour traiter les logs entrants

Ensuite nous cochons « **Allow overriding date** » afin de pouvoir réécrire la date si nécessaire et nous sauvegardons notre input :

☐ Force rDNS?  
 Force rDNS resolution of hostname? Use if hostname cannot be parsed. (Be careful if you are sending DNS logs into this input because it can cause a feedback loop.)

☒ Allow overriding date?  
 Allow to override with current date if date could not be parsed?

☐ Store full message?  
 Store the full original syslog message as full\_message?

☐ Expand structured data?  
 Expand structured data elements by prefixing attributes with their SD-ID?

**Time Zone (optional)**  

Not configured

 Default time zone used when no timezone detected

Cancel

Launch Input

Maintenant nous allons adapter un fichier .json que nous avons trouvé sur Github (lien en annexe) qui va servir à extraire et transformer les logs envoyées par Pfsense. Celui-ci étant pour Suricata nous allons l'adapter pour Snort

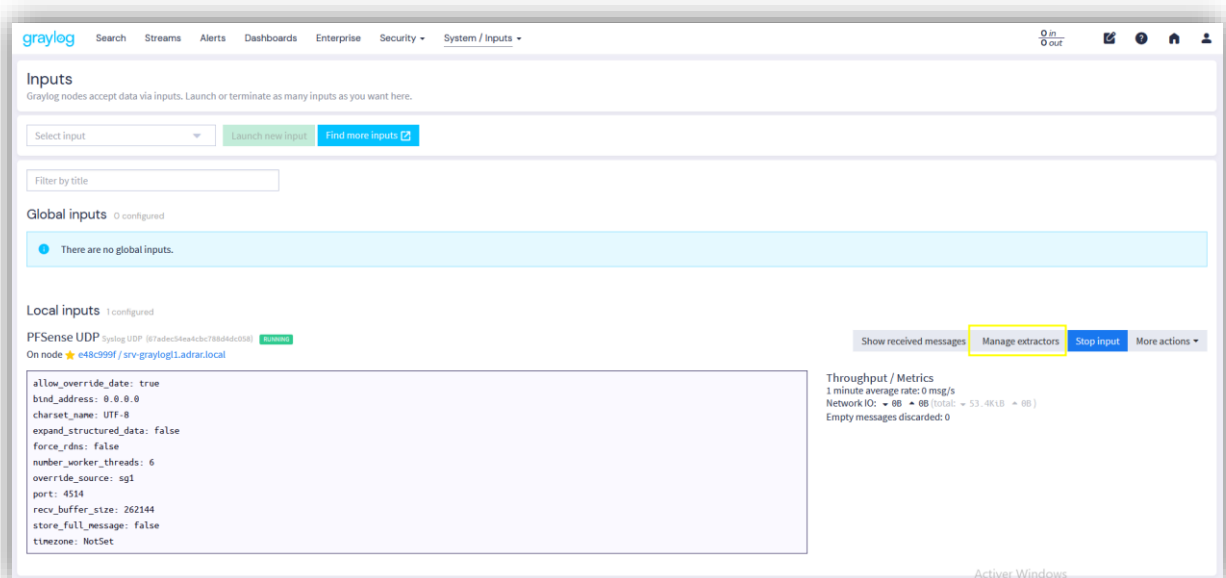
```
{
  "extractors": [
    {
      "title": "Extract Snort Application Name",
      "extractor_type": "regex",
      "converters": [
        {
          "type": "lowercase",
          "config": {}
        }
      ],
      "order": 0,
      "cursor_strategy": "copy",
      "source_field": "message",
      "target_field": "application_name",
      "extractor_config": {
```

```

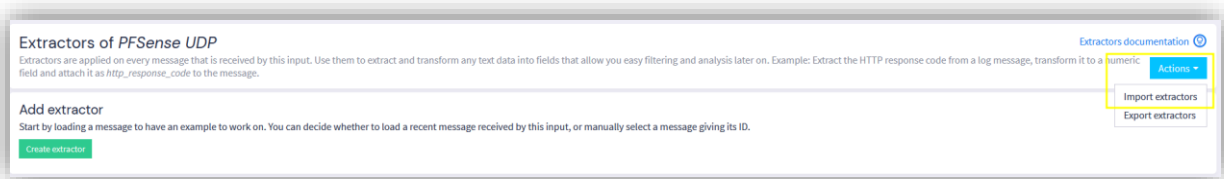
    "regex_value": "\\[[0-9]:[0-9]*:[0-9]\\] (SNORT) .*"
  },
  "condition_type": "regex",
  "condition_value": "\\[[0-9]:[0-9]*:[0-9]\\] (SNORT) .*"
},
{
  "title": "General Application Name Extractor",
  "extractor_type": "regex",
  "converters": [],
  "order": 0,
  "cursor_strategy": "copy",
  "source_field": "message",
  "target_field": "application_name",
  "extractor_config": {
    "regex_value": "([a-z-]+).*"
  },
  "condition_type": "regex",
  "condition_value": "([a-z-]+).*"
}
],
"version": "3.2.5"
}

```

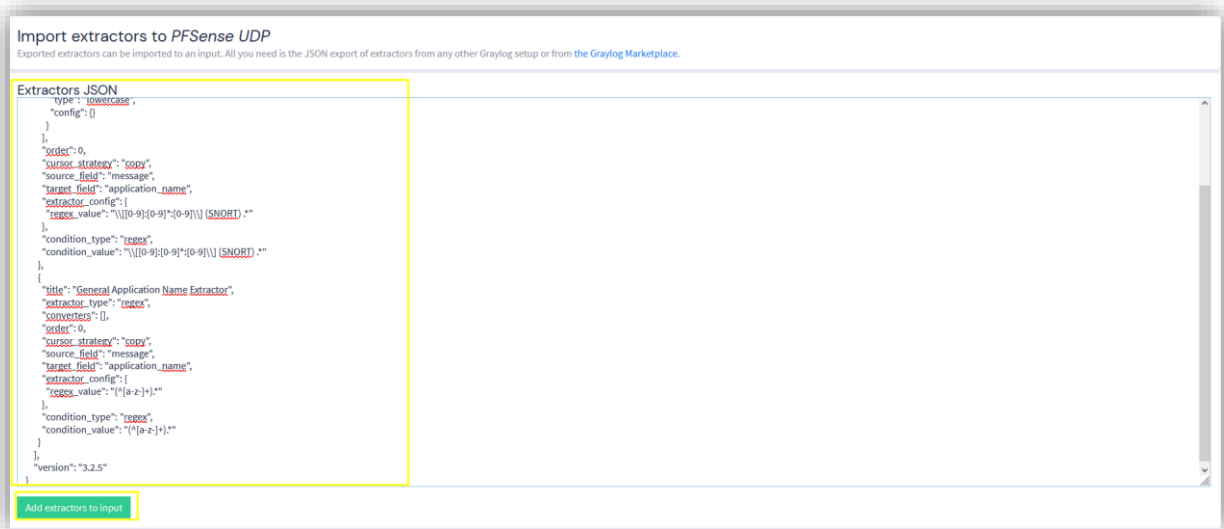
Dans le menu « **System / Inputs -> Inputs** » de Graylog nous allons donc cliquer sur « **Manage Extractor** » :



Puis « **Actions -> Import extractor** »



Ensuite nous collons notre code et ajoutons l'extractor à l'input :



Maintenant une configuration est nécessaire sur Pfsense.

Nous allons dans le menu « **Status -> System Log -> Setting** » dans le bas de la page nous

trouvons les configurations pour envoyer les logs à un hôte distant :

**Remote Logging Options**

**Enable Remote Logging** ☒ Send log messages to remote syslog server **1**

**Source Address** LAN **2**  
 This option will allow the logging daemon to bind to a single IP address, rather than all IP addresses. If a single IP is picked, remote syslog servers must all be of that IP type. To mix IPv4 and IPv6 remote syslog servers, bind to all interfaces.  
 NOTE: If an IP address cannot be located on the chosen interface, the daemon will bind to all addresses.

**IP Protocol** IPv4 **3**  
 This option is only used when a non-default address is chosen as the source above. This option only expresses a preference; If an IP address of the selected type is not found on the chosen interface, the other type will be tried.

**Remote log servers** 192.168.10.3:4514 **4** IP[:port] IP[:port]

**Remote Syslog Contents** ☒ Everything **5**  
☐ System Events  
☐ Firewall Events  
☐ DNS Events (Resolver/unbound, Forwarder/dnsmasq, filterdns)  
☐ DHCP Events (DHCP Daemon, DHCP Relay, DHCP Client)  
☐ PPP Events (PPPoE WAN Client, L2TP WAN Client, PPTP WAN Client)  
☐ General Authentication Events  
☐ Captive Portal Events  
☐ VPN Events (IPsec, OpenVPN, L2TP, PPPoE Server)  
☐ Gateway Monitor Events  
☐ Routing Daemon Events (RADVD, UPnP, RIP, OSPF, BGP)  
☐ Network Time Protocol Events (NTP Daemon, NTP Client)  
☐ Wireless Events (hostapd)  
 Syslog sends UDP datagrams to port 514 on the specified remote syslog server, unless another port is specified. Be sure to set syslogd on the remote server to accept syslog messages from pfSense.

**Save**

1. Active les logs distants
2. Interface utilise pour envoyer les logs
3. Protocole IPv4
4. IP et port utilisé du serveur Graylog
5. Informations envoyées au serveur Graylog (que nous pourrions affiner via ce menu plus tard ou restreindre via la création de Dashboard personnalisé dans Graylog)

Une fois cela fait dans le menu « **Search** » nous pouvons effectuer notre recherche de log en indiquant les filtres voulus :

**graylog** Search Streams Alerts Dashboards Enterprise Security System

Search • Unsaved Search

From: 5 minutes ago Until: Now

Select streams the search should include. Searches in all streams if empty.

\_response\_code:[400 TO 494]

Save Load Share

**Search Time Range**

Relative Absolute Keyword Save as preset Load Preset

From:  All Time Hours ago

Until: ☒ Now  Seconds ago

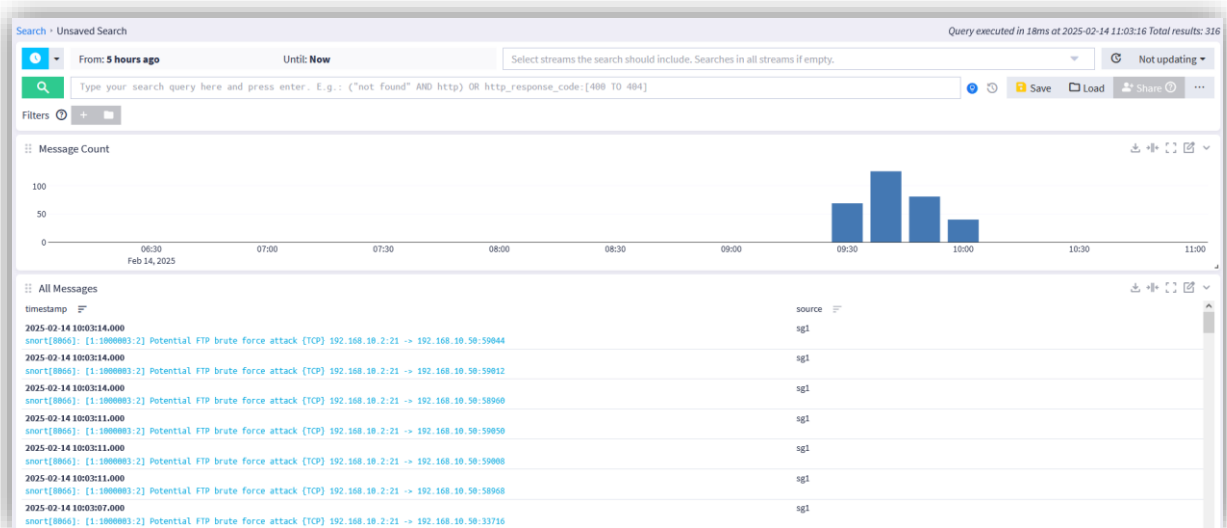
All timezones using UTC Cancel Update time range

Feb 14, 2025

10:58:00 10:58:30 10:59:00 10:59:30 11:00:00 11:00:30

All Messages

Une fois la recherche lancée nous retrouvons les logs générés par notre Pfsense :



Puis en cliquant sur une ligne, dans notre cas la tentative de brut force, nous avons plus de détail :



1. Heure de l'alerte
2. Input utilisé pour la réception des logs
3. Service qui envoi l'alerte
4. Message de l'application
5. Source de l'envoi des logs (notre pare feu Pfsense)

Beaucoup d'options sont possibles pour les recherches par exemples des filtrages par noms d'événements, nom de service, SID d'une règle, IP sources ou destination ect...

Nous allons maintenant configurer une politique de rétention des logs.

Pour cela, nous allons dans le menu « **System -> Indices** » puis « **create index set** » :

**Create Index Set**

Create a new index set that will let you configure the retention, sharding, and replication of messages coming from one or more streams.

[Index model documentation](#)

[Select Template](#)

**Configuration Information**

**Title**  
Pfsense-logs (1)  
Descriptive name of the index set.

**Description**  
Pfsense-logs (2)  
Add a description of this index set.

**Details**

**Index prefix**  
pfsense (3)  
A unique prefix used in Elasticsearch indices belonging to this index set. The prefix must start with a letter or number, and can only contain letters, numbers, '-', '\_', and '\*'.

**Analyzer**  
standard (4)  
Elasticsearch analyzer for this index set.

**Index Shards**  
1 (5)  
Number of search cluster Shards used per index in this Index Set. Increasing the Index Shards improves the search cluster write speed of data stored to this Index Set by distributing the active write index over multiple search nodes. Increasing the Index Shards can degrade search performance and increases the memory footprint of the index. This value should not be set higher than the number of search nodes.

**Index Replica**  
1 (6)  
Number of search cluster Replica Shards used per index in this Index Set. Adding Replica Shards improves search performance during parallel reads of the index, such as occurs on dashboards, and is a component of HA and backup strategy. Each Replica Shard set multiplies the storage requirement and memory footprint of the index. This value should not be set higher than the number of search nodes, and typically not higher than 1.

**Maximum Number of Segments**  
1 (7)  
Advanced Option: Maximum number of segments per Search Cluster Index after optimization (force merge). Setting higher values decreases the compression ratio of Index Optimization.

[Activate Windows](#)  
Go to Settings to activate Windows.

1. Titre de l'indice
2. Description
3. Préfix utilisé pour les logs
4. Nombre de fragment (Shards) défini pour diviser l'index (Dépend du nombre de nœud Graylog, ne pas mettre une valeur trop élevée au risque réduire les performances)
5. Nombre de copie de fragment (Shards) à effectuer
6. Copies des données effectuées pour en éviter la perte et accélérer la lecture.
7. Compression des index pour optimiser l'espace et les performances.

**Rotation & Retention**

[Data Streams](#) [Legacy \(Deprecated\)](#)

30 days

Deleted after 365 days

**Max. days in storage**  
365 (1)  
After how many days your data should be deleted.

**Min. days in storage**  
30 (2)  
How many days at minimum your data should be stored.

**Field Type Profile**

With index set field type profiles you can bundle up custom field types into profiles. You can assign any profile to this index set. To see and use profile setting for index set, you have to rotate indices.

**Index field type mapping profile**  
Select index field type profile

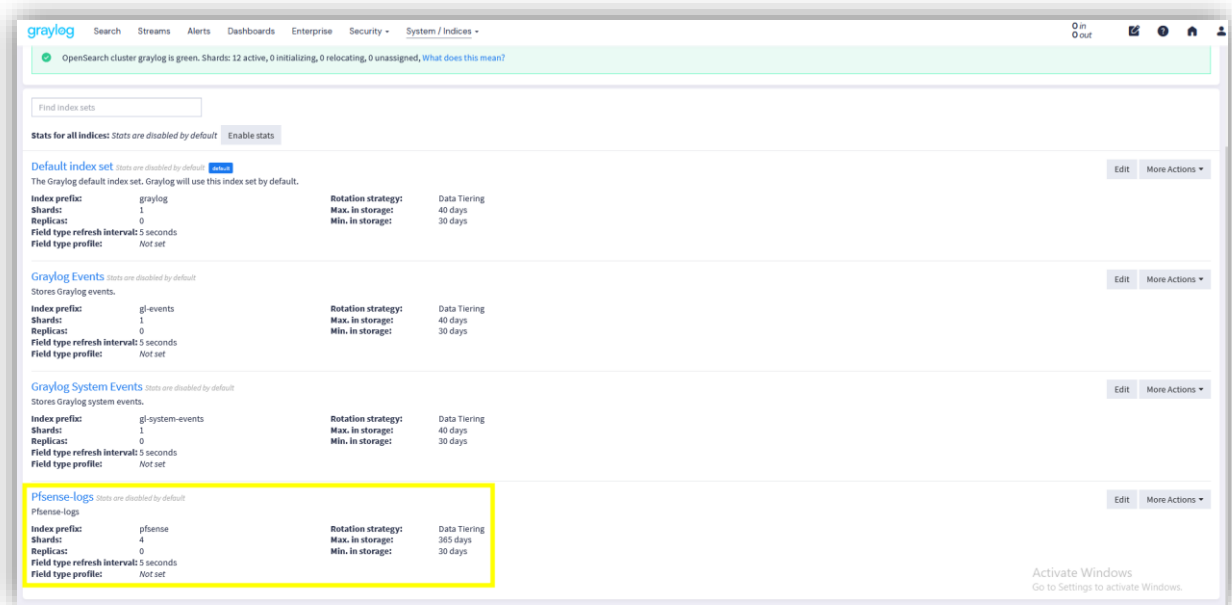
[Activate Windows](#)  
Go to Settings to activate Windows.

[Create index set](#) [Cancel](#)

1. Durée de conservation des archives
2. Durée de conservation des logs avant archives

Nous retrouvons maintenant notre indice créé avec les paramètres définis :





Nous pouvons maintenant affiner nos recherches en créant un stream qui permettra de filtrer les logs et afficher uniquement celle concernant Snort.

Pour cela nous allons dans le menu « **Streams** » puis « **create streams** » et renseignons les paramètres suivants :

Editing Stream

Title

Snort

1

A descriptive name of the new stream

Description (Opt.)

Log de Snort

2

What kind of messages are routed into this stream?

Index Set

Pfsense-logs

3

Messages that match this stream will be written to the configured index set.

☒ Remove matches from 'Default Stream'

4

Don't assign messages that match this stream to the 'Default Stream'.

Cancel

Update stream

1. Titre du stream
2. Description
3. Indice utilisé pour la rétention des logs (celui créé précédemment)
4. Supprime les logs du stream par défaut afin de ne pas avoir de doublon et surcharger le stockage

Nous voyons ensuite notre stream créé, nous allons donc pouvoir créer notre règle de filtrage des logs via « **manage rules** » :

graylog

Search Streams Alerts Dashboards Enterprise Security System

0 in 0 out

Streams

You can route incoming messages into streams by applying rules against them. Messages matching the rules of a stream are routed into it. A message can also be routed into multiple streams.

Streams documentation

Create stream

Search for streams

Filters

Bulk actions

Title

Index Set

Archiving

Rules

Pipelines

Outputs

Throughput

Status

Actions

<input type="checkbox"/> All events	Graylog Events					0 msg/s	Running	<a href="#">Data Routing</a>	<a href="#">Share</a>	<a href="#">More</a>
<input type="checkbox"/> All system events	Graylog System Events					0 msg/s	Running	<a href="#">Data Routing</a>	<a href="#">Share</a>	<a href="#">More</a>
<input type="checkbox"/> Default Stream <a href="#">Duplicate</a>	Default index set					0 msg/s	Running	<a href="#">Data Routing</a>	<a href="#">Share</a>	<a href="#">More</a>
<input type="checkbox"/> Snort Pfsense-logs	Pfsense-logs					0 msg/s	Running	<a href="#">Data Routing</a>	<a href="#">Share</a>	<a href="#">More</a>

Stop Stream

Quick add rule

Edit stream

Manage Rules

Manage Outputs

Manage Alerts

Set as startpage

Clone this stream

Delete this stream

Nous renseignons les paramètres suivants :

### New Stream Rule

**Field**

**Type**

**Value**

☐ Inverted

**Description (Opt.)**

**Result:** *source must contain snort*

Cancel

Create Rule

The server will try to convert to strings or numbers based on the matcher type as well as it can.

[Take a look at the matcher code on GitHub](#)

Regular expressions use Java syntax.

1. Filtrage des logs sur la source
2. Doit contenir la valeur dans le 3
3. Valeur à respecter pour la règle

Nous allons maintenant lancer notre stream en cliquant sur le bouton played :

Streams

You can route incoming messages into streams by applying rules against them. Messages matching the rules of a stream are routed into it. A message can also be routed into multiple streams.

Search for streams

Filters

All events

Stream containing all events created by Graylog

All system events

Stream containing all system events created by Graylog

Default Stream

Contains messages that are not explicitly routed to other streams

Snort

PfSense-logs

Index Set

Graylog Events

Graylog System Events

Default index set

PfSense-logs

Archiving

Rules

Pipelines

Outputs

Throughput

Status

Actions

0 msg/s

Running

0 msg/s

Running

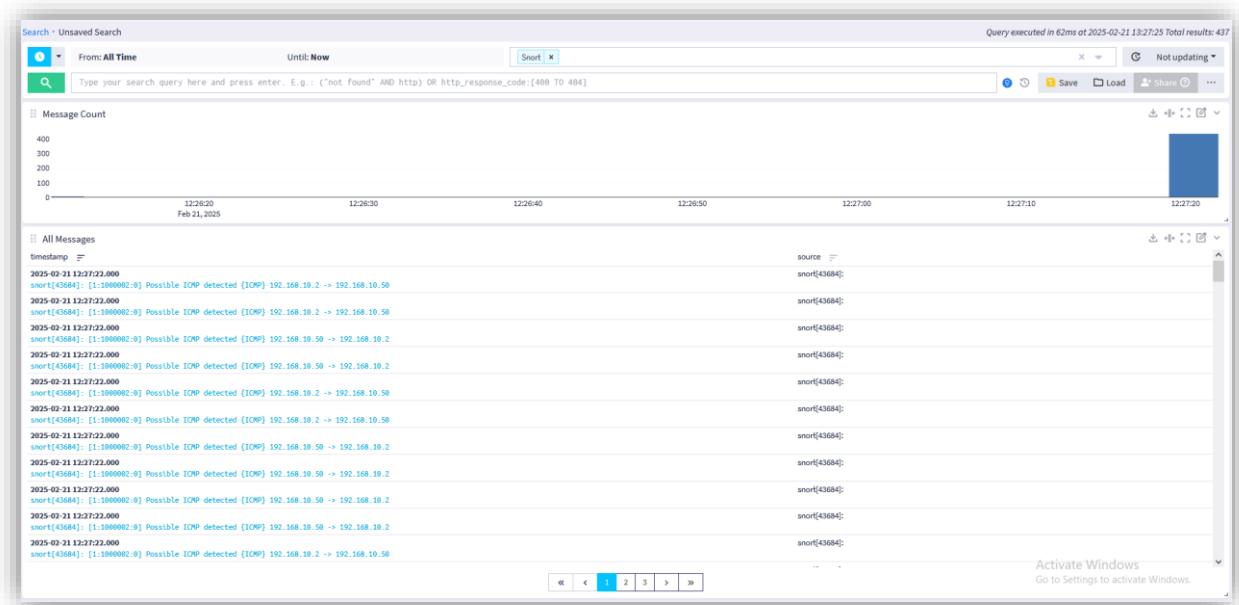
0 msg/s

Running

0 msg/s

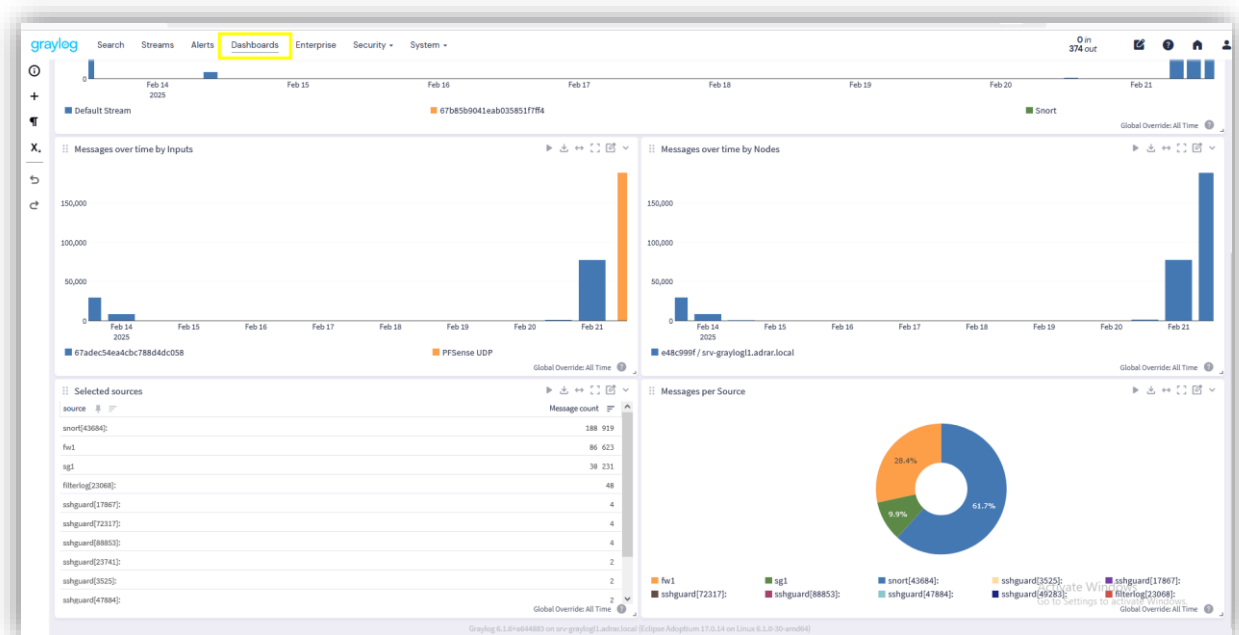
Paused

En cliquant sur notre stream nous retrouvons maintenant les logs concernant Snort uniquement :



Dans l'onglet Dashboard nous pouvons aussi créer des vues personnalisées et les associer à notre stream pour avoir des statiques sur les logs traité par Graylog (nombre d'alertes par services, date des alertes, statistiques sur les attaques remontées par Snort ect...).

Voici un exemple en utilisant le Dashboard par défaut que nous retrouvons dans le menu « **Dashboard** » :



Il est possible de personnaliser ces Dashboards à souhait afin d'affiner les statistiques selon les besoins cependant un certain temps de configuration sera nécessaire afin d'extraire correctement les informations des logs via des regexs et extractors personnalisés.

Maintenant nous allons configurer les remontées d'alertes par mail.

Nous commençons par configurer le serveur SMTP dans le fichier

« **/etc/graylog/server/server.conf** »

```

liam@srv-graylog1: ~
GNU nano 7.2 /etc/graylog/server/server.conf *
# if you encounter MongoDB connection problems.
mongodb_max_connections = 1000

# Maximum number of attempts to connect to MongoDB on boot for the version probe
#
# Default: 0, retry indefinitely until a connection can be established
#mongodb_version_probe_attempts = 5

# Email transport
transport_email_enabled = true
transport_email_hostname = smtp.freesmtpservers.com
transport_email_port = 25
transport_email_use_auth = false
transport_email_from_email = graylog@adrar.local
transport_email_socket_connection_timeout = 10s
transport_email_socket_timeout = 10s

# Encryption settings
#
# ATTENTION:

^G Aide      ^O Écrire    ^W Chercher  ^K Couper    ^T Exécuter  ^C Emplacement
^X Quitter   ^R Lire fich.^_ Remplacer  ^U Coller    ^J Justifier ^/ Aller ligne
  
```

Maintenant nous allons dans le menu « Alerts → Notifications » puis « create Notifications »

**Title**

Email Notification

Title to identify this Notification.

**Description (Optional)**

Longer description for this Notification.

**Notification Type**

Email Notification

Choose the type of Notification to create.

**Subject**

Graylog event notification: \${event\_definition\_title}

The subject that should be used for the email notification.

**Reply-To (Optional)**

The email address that recipients should use for replies.

☐ Use lookup table for Reply To email

**Sender (Optional)**

The email address that should be used as the notification sender. Leave it empty to use the default sender address.

☐ Use lookup table for Sender email

**User recipient(s) (Optional)**

graylog-sidecar (Sidecar System User (built-in))

Select Graylog users that will receive this Notification.

1. Titre de la notification
2. Type de notification
3. Objet du mail
4. Utilisateur pour l'envoi du mail

Puis nous cliquons sur « Create notification »

**Time zone for date/time values (Optional)**

UTC

Time zone used for timestamps in the email body.

**Body Template**

```

1 --- [Event Definition] -----
2 Title:      ${event_definition_title}
3 Description: ${event_definition_description}
4 Type:      ${event_definition_type}
5 --- [Event] -----
6 Alert Replay:  ${http_external_url}alerts/${event.id}/replay-search
7 Timestamp:    ${event.timestamp}
8 Message:      ${event.message}
9 Source:       ${event.source}
10 Key:         ${event.key}
11 Priority:     ${event.priority}
12 Alert:       ${event.alert}
13 Timestamp Processing: ${event.timestamp_start}
14 Tlrange Start:  ${event.tlrange_start}
15 Tlrange End:    ${event.tlrange_end}

```

The template that will be used to generate the email body.

**HTML Body Template**

```

16 <tr><td>Timestamp Processing</td><td>${event.timestamp}</td></tr>
17 <tr><td>Tlrange Start</td><td>${event.tlrange_start}</td></tr>
18 <tr><td>Tlrange End</td><td>${event.tlrange_end}</td></tr>
19 <tr><td>Source Streams</td><td>${event.source_streams}</td></tr>
20 <tr><td>Fields</td><td><ul style="list-style-type:square;">${foreach event.fields field}<li>${field.key}: ${field.value}</li></ul></td></tr>
21 </tbody></table>
22 <div>
23 <div>
24 <table width="100%" border="0" cellpadding="10" cellspacing="0" style="background-color:#f9f9f9;border:none;line-height:1.2"><tbody>
25 <tr><td colspan="2">${foreach backlog message}
26 <tr><td colspan="2">${message}</td></tr>
27 </tr></td></tr>
28 </tbody></table>
29 </div>
30 </div>

```

The template that will be used to generate the email HTML body.

**Test Notification (Optional)**

Execute Test Notification

Execute this Notification with a test Alert.

Create notification Cancel

Nous voyons également ci-dessus le « **Body template** » qui est 100% personnalisable afin d'envoyer les informations voulues concernant l'événement.

Maintenant nous allons dans le menu « **Alerts -> Events definition** » puis nous allons cliquer sur « **create event definition** »

**New Event Definition 'Event Snort'**

Event Definitions allow you to create Alerts from different Conditions and alert on them.

Alerts documentation

Event Details Condition Fields Notifications Summary

**Event Details**

**Title**

Event Snort

Title for this Event Definition, Events and Alerts created from it.

**Description (Optional)**

Longer description for this Event Definition.

**Remediation Steps (Optional)**

Edit Preview

1

**Priority**

Normal

Choose the priority for Events created from this Definition.

Previous Next

1. Titre de l'événement
2. Possibilité de rajouter des étapes de remédiation
3. Priorité de l'événement

graylog Search Streams Alerts Dashboards Enterprise Security System 3

Configure how Graylog should create Events of this kind. You can later use those Events as input on other Conditions, making it possible to build powerful Conditions based on others.

**Condition Type**

Filter & Aggregation 1

Choose the type of Condition for this Event.

**Filter**

Add information to filter the log messages that are relevant for this Event Definition.

**Search Query**

level:1 2

Search query that Messages should match. You can use the same syntax as in the Search page, including declaring Query Parameters from Lookup Tables by using the `$newParameter$` syntax.

**Streams (Optional)**

Snort x 3

Select streams the search should include. Searches in all streams if empty.

**Search within the last**

24 4 hours

☐ Use Cron Scheduling

Schedule this event with a Quartz cron expression

**Execute search every**

5 5 minutes

☒ Enable

Should this event definition be executed automatically?

1. Paramètre de déclenchement de l'alerte
2. Filtre sur les alertes de niveau 1 minimum (pour nos tests, le niveau d'alerte est à ajuster au besoin)
3. Filtre sur le stream créé précédemment pour les alertes Snort
4. Recherche dans les logs générés depuis les dernières 24 heures
5. Exécute la recherche toutes les 5 mins

Puis nous définissons la notification créée précédemment :

Alerts & Events Event Definitions Notifications

New Event Definition "Event Short"

Event Definitions allow you to create Alerts from different Conditions and alert on them.

Alerts documentation

Event Details Filter & Aggregation Fields Notifications Summary

**Add Notification**

Choose Notification

Email Notification x

Select a Notification to use on Alerts of this kind or create a new Notification that you can later use in other Alerts.

Add notification Cancel

Previous Next

Ensuite nous effectuons un test de notification de l'alerte :

Subject : [Graylog event notification: Snort\\_alerts](#)

From : graylog@adrr.local To : sidecar@graylog.local

Dated : 2025-02-23 18:00:45 ( [Delete](#) )

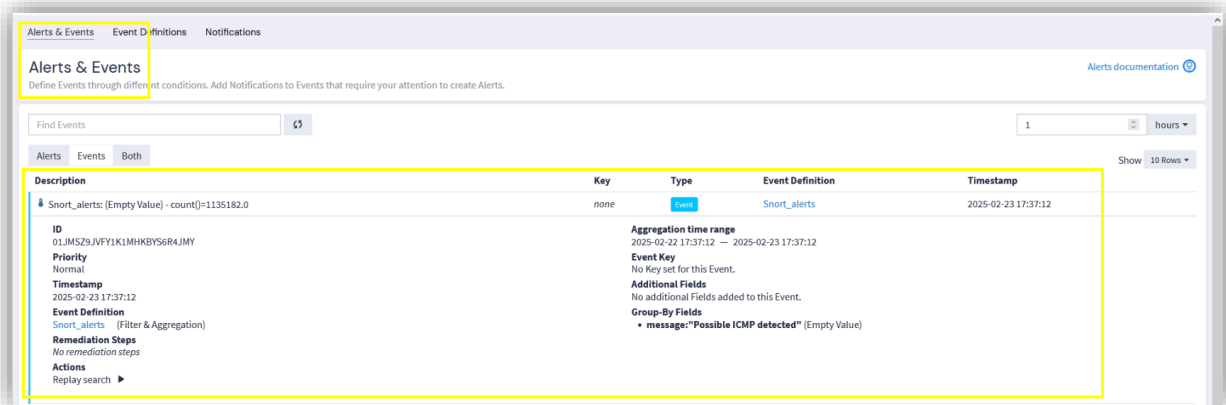
HTML HTML Source Text RAW

Event Definition	
Title	Snort_alerts
Description	
Type	aggregation-v1

Event	
Alert Replay	<a href="http://192.168.10.3:9000/alerts/01JMSZFY4355DW8Y7JPMX8Q8KF/replay-search">http://192.168.10.3:9000/alerts/01JMSZFY4355DW8Y7JPMX8Q8KF/replay-search</a>
Timestamp	2025-02-23T17:39:07.954Z
Message	Snort_alerts: (Empty Value) - count()=1291730.0
Source	srv-graylog1.adrr.local
Key	
Priority	2
Alert	true
Timestamp Processing	2025-02-23T17:39:07.954Z
Timerange Start	2025-02-22T17:39:07.954Z
Timerange End	2025-02-23T17:39:07.954Z
Source Streams	[67b87eae41eab03585202b2a, 000000000000000000000001]
Fields	

Les alertes sont également disponibles dans le menu « **Alerts et events** » ou plus de détail sont disponible :





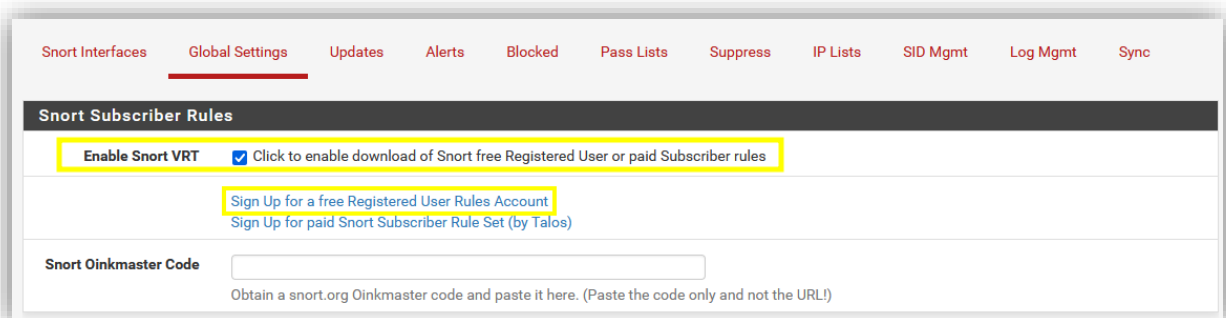
Il est aussi possible de rejouer l'alerte afin de remonter tous les événements associés à cette alerte.

Graylog étant un outil très complet, il est possible de modifier toutes les alertes, stream, notifications, dashboard... à souhait afin d'avoir les informations les plus précises possible. Cependant un certain temps de configuration sera nécessaire afin d'arriver au résultat les plus optimisés possible car cela implique la création de nouveau extractor (que l'on peut télécharger via la market place Graylog ou les créer manuellement).

## 2.6 Configuration des règles « Snort Community »

Maintenant que nous avons qualifié notre POC nous allons pouvoir mettre en place les règles Snort Community qui regroupent énormément de règles créées par la communauté pour les attaques les plus courantes (cela évite la création manuelle de toutes les règles qui demande un certain temps).

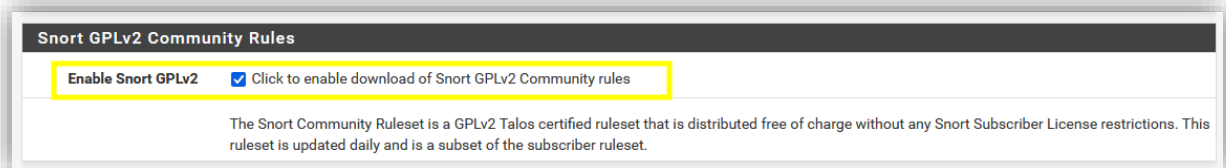
Nous éditons donc notre interface puis nous allons dans le menu « **Snort -> Global Settings** » pour configurer les options suivantes :



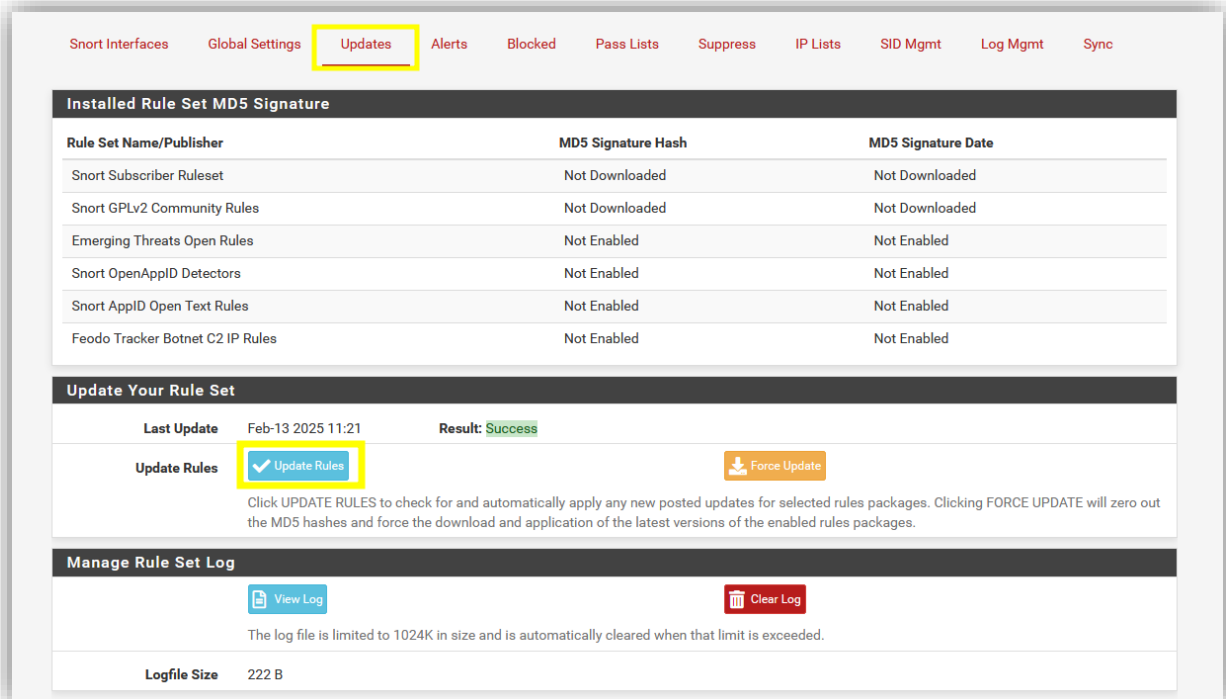
En cliquant sur « **Sign up a free ...** » nous sommes redirigées vers le site de Snort afin de récupérer un token pour télécharger les règles communautaires.

Une fois notre compte créé dans le menu « **Account -> Oinkcode** » nous retrouvons notre token à renseigner dans Pfsense :

Nous activons ensuite les règles communautaires :



Nous mettons à jour les règles via le menu « **Updates** » :



Dans le menu de l'interface puis « **WAN Categories** » nous retrouvons toutes les règles à activer au besoin :

WAN Settings
WAN Categories
WAN Rules
WAN Variables
WAN Preprocs
WAN IP Rep
WAN Logs

### Automatic Flowbit Resolution

**Resolve Flowbits** ☒ If checked, Snort will auto-enable rules required for checked flowbits. Default is Checked.  
 Snort will examine the enabled rules in your chosen rule categories for checked flowbits. Any rules that set these dependent flowbits will be automatically enabled and added to the list of files in the interface rules directory.

### Snort Subscriber IPS Policy Selection

**Use IPS Policy** ☐ If checked, Snort will use rules from one of three pre-defined IPS policies in the Snort Subscriber rules. Default is Not Checked.  
 Selecting this option disables manual selection of Snort Subscriber categories in the list below, although Emerging Threats categories may still be selected if enabled on the Global Settings tab. These will be added to the pre-defined Snort IPS policy rules from the Snort VRT.

### Select the rulesets (Categories) Snort will load at startup

▲ - Category is auto-enabled by SID Mgmt conf files  
● - Category is auto-disabled by SID Mgmt conf files

Select All Unselect All Save

**Enable** Ruleset: Snort GPLv2 Community Rules

☐ Snort GPLv2 Community Rules (Talos certified)

Enable	Ruleset: ET Open Rules	Enable	Ruleset: Snort Text Rules	Enable	Ruleset: Snort SO Rules	Snort OPENAPPID rules are not enabled.
<input type="checkbox"/>	emerging-activex.rules	<input type="checkbox"/>	snort_app-detect.rules	<input type="checkbox"/>	snort_browser-chrome.so.rules	
<input type="checkbox"/>	emerging-attack_response.rules	<input type="checkbox"/>	snort_attack-responses.rules	<input type="checkbox"/>	snort_browser-ie.so.rules	
<input type="checkbox"/>	emerging-botcc.portgrouped.rules	<input type="checkbox"/>	snort_backdoor.rules	<input type="checkbox"/>	snort_browser-other.so.rules	
<input type="checkbox"/>	emerging-botcc.rules	<input type="checkbox"/>	snort_bad-traffic.rules	<input type="checkbox"/>	snort_browser-webkit.so.rules	
<input type="checkbox"/>	emerging-chat.rules	<input type="checkbox"/>	snort_blacklist.rules	<input type="checkbox"/>	snort_exploit-kit.so.rules	
<input type="checkbox"/>	emerging-ciarmy.rules	<input type="checkbox"/>	snort_botnet-cnc.rules	<input type="checkbox"/>	snort_file-executable.so.rules	
<input type="checkbox"/>	emerging-compromised.rules	<input type="checkbox"/>	snort_browser-chrome.rules	<input type="checkbox"/>	snort_file-flash.so.rules	
<input type="checkbox"/>	emerging-current_events.rules	<input type="checkbox"/>	snort_browser-firefox.rules	<input type="checkbox"/>	snort_file-image.so.rules	
<input type="checkbox"/>	emerging-deleted.rules	<input type="checkbox"/>	snort_browser-ie.rules	<input type="checkbox"/>	snort_file-java.so.rules	

Pour notre exemple, nous allons activer « **emerging-scan rules** » afin d’alerter en cas de scan de port ouvert sur le réseau par exemple :

<input type="checkbox"/> emerging-pop3.rules	<input type="checkbox"/> snort_file-office.rules	<input type="checkbox"/> snort_protocol-ftp.so.rules
<input type="checkbox"/> emerging-retired.rules	<input type="checkbox"/> snort_file-other.rules	<input type="checkbox"/> snort_protocol-voip.so.rules
<input type="checkbox"/> emerging-rpc.rules	<input type="checkbox"/> snort_file-pdf.rules	<input type="checkbox"/> snort_pua-p2p.so.rules
<input type="checkbox"/> emerging-scada.rules	<input type="checkbox"/> snort_finger.rules	<input type="checkbox"/> snort_server-iis.so.rules
<input checked="" type="checkbox"/> emerging-scan.rules	<input type="checkbox"/> snort_ftp.rules	<input type="checkbox"/> snort_server-mail.so.rules
<input type="checkbox"/> emerging-shellcode.rules	<input type="checkbox"/> snort_icmp-info.rules	<input type="checkbox"/> snort_server-mysql.so.rules
<input type="checkbox"/> emerging-smtp.rules	<input type="checkbox"/> snort_icmp.rules	<input type="checkbox"/> snort_server-oracle.so.rules
<input type="checkbox"/> emerging-snmp.rules	<input type="checkbox"/> snort_imap.rules	<input type="checkbox"/> snort_server-other.so.rules
<input type="checkbox"/> emerging-sql.rules	<input type="checkbox"/> snort_indicator-compromise.rules	<input type="checkbox"/> snort_server-webapp.so.rules
<input type="checkbox"/> emerging-telnet.rules	<input type="checkbox"/> snort_indicator-obfuscation.rules	
<input type="checkbox"/> emerging-tftp.rules	<input type="checkbox"/> snort_indicator-scan.rules	
<input type="checkbox"/> emerging-tor.rules	<input type="checkbox"/> snort_indicator-shellcode.rules	
<input type="checkbox"/> emerging-trojan.rules	<input type="checkbox"/> snort_info.rules	
<input type="checkbox"/> emerging-user_agents.rules	<input type="checkbox"/> snort_local.rules	
<input type="checkbox"/> emerging-voip.rules	<input type="checkbox"/> snort_malware-backdoor.rules	
<input type="checkbox"/> emerging-web_client.rules	<input type="checkbox"/> snort_malware-cnc.rules	
<input type="checkbox"/> emerging-web_server.rules	<input type="checkbox"/> snort_malware-other.rules	
<input type="checkbox"/> emerging-web_specific_apps.rules	<input type="checkbox"/> snort_malware-tools.rules	
<input type="checkbox"/> emerging-worm.rules	<input type="checkbox"/> snort_misc.rules	
	<input type="checkbox"/> snort_multimedia.rules	
	<input type="checkbox"/> snort_mysql.rules	
	<input type="checkbox"/> snort_netbios.rules	
	<input type="checkbox"/> snort_nttp.rules	
	<input type="checkbox"/> snort_oracle.rules	
	<input type="checkbox"/> snort_os-linux.rules	
	<input type="checkbox"/> snort_os-mobile.rules	

Puis sur notre Kali nous allons faire un NMAP et constater sur Graylog l'alerte remontée.

All Messages

timestamp

2025-02-20 14:11:51.000

Received by

PFsense UDP on e48c999f / srv-graylog1.adrar.local

Stored in index

graylog\_1

Routed into streams

Default Stream

source

FW1

message

snort[18682]: [1:2009582:3] ET SCAN NMAP -sS window 1024 [Classification: Attempted Information Leak] [Priority: 2] [TCP] 192.168.10.50:56389 -> 192.168.10.2:53

application\_name

snort

facility

security/authorization

facility\_num

4

level

1

timestamp

2025-02-20 14:11:51.000

FW1

Activate Windows

Go to Settings to activate Windows.

Attention : Beaucoup de règles sont disponibles, il est nécessaire de faire un tri et d'activer celle qui nous semble les plus pertinentes afin de ne pas surcharger notre SIEM de logs qui ne sont pas pertinents.

### 3. Conclusion

Dans le cadre de ce projet, nous avons mis en place une solution de sécurisation du réseau pour l'ADRAR en intégrant un service NIDS avec Snort sur le pare-feu Pfsense. Cette solution permet de détecter les attaques de type Brute Force FTP, DDoS ICMP et ARP Spoofing mais aussi les scans réseau... tout en générant des alertes pour une surveillance proactive.

Afin d'améliorer la gestion des événements de sécurité, nous avons également déployé un serveur Graylog pour la centralisation et l'analyse des logs. Grâce à son interface intuitive et ses capacités de filtrage avancées, l'ADRAR pourra mieux visualiser les tentatives d'intrusion et réagir efficacement en cas d'incident.

À terme, cette infrastructure sécurisée et évolutive permettra à l'ADRAR de mieux protéger son réseau contre les menaces et d'assurer une supervision efficace des événements de sécurité.

## 4. Annexes

Installation et configuration de Snort sur Pfsense : <https://iritt.medium.com/setting-up-snort-for-network-monitoring-on-pfsense-for-your-cybersecurity-lab-f724e48d6221>

Installation et configuration de Graylog : <https://www.it-connect.fr/tuto-graylog-sur-debian-centraliser-et-analyser-logs/>

Créer des règles Snort personnalisées : <https://cyvatar.ai/write-configure-snort-rules/>

Marketplace Graylog : <https://community.graylog.org/c/marketplace/31>

Recommandation ANSSI SIEM [https://cyber.gouv.fr/sites/default/files/2022/01/anssi-guide-recommandations\\_securite\\_architecture\\_systeme\\_journalisation.pdf](https://cyber.gouv.fr/sites/default/files/2022/01/anssi-guide-recommandations_securite_architecture_systeme_journalisation.pdf)