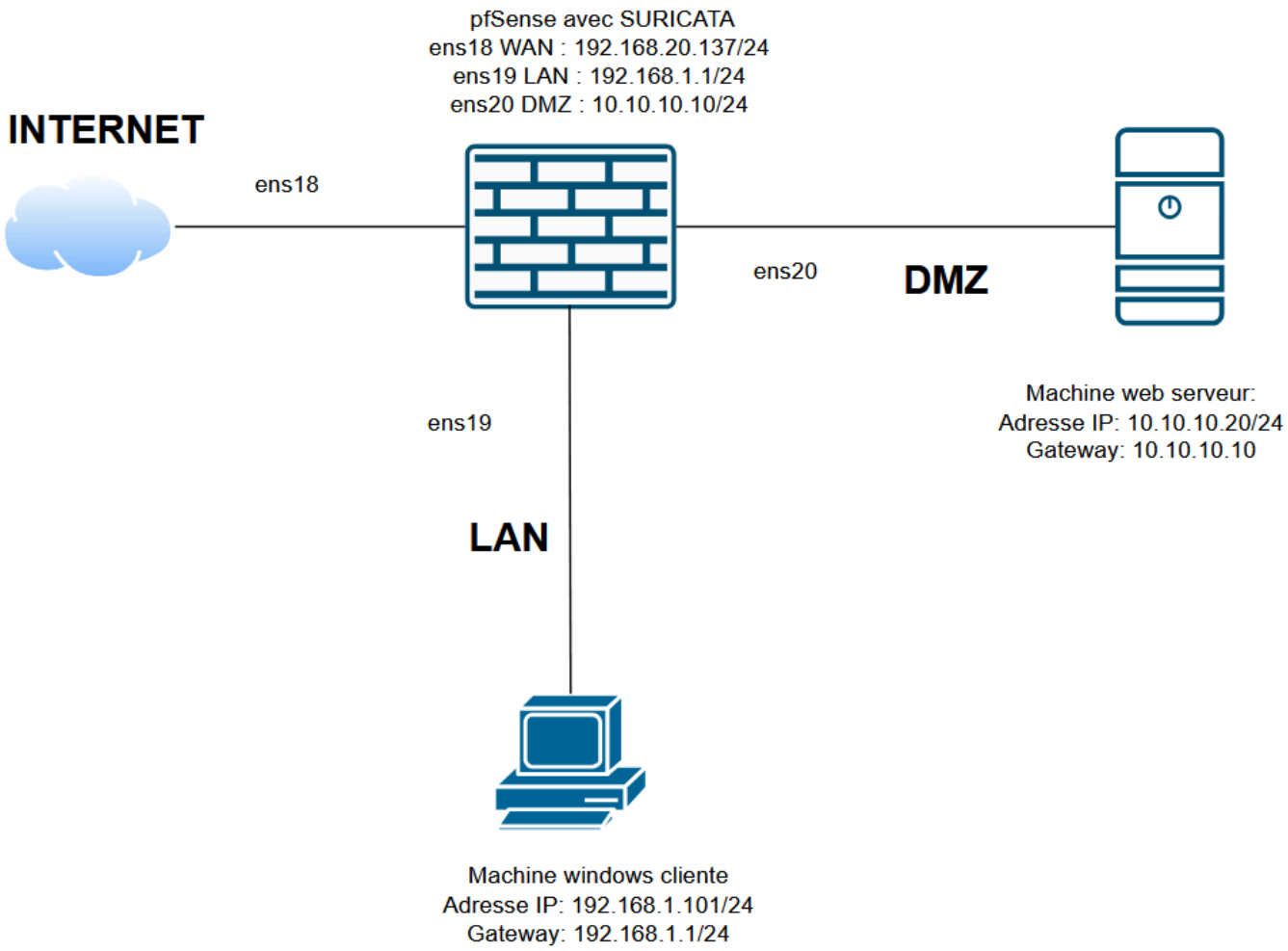
Tp-Suricata.md

TP-IDS/IPS Thomas GRZESINSKI

C’est quoi un IDS et un IPS ?

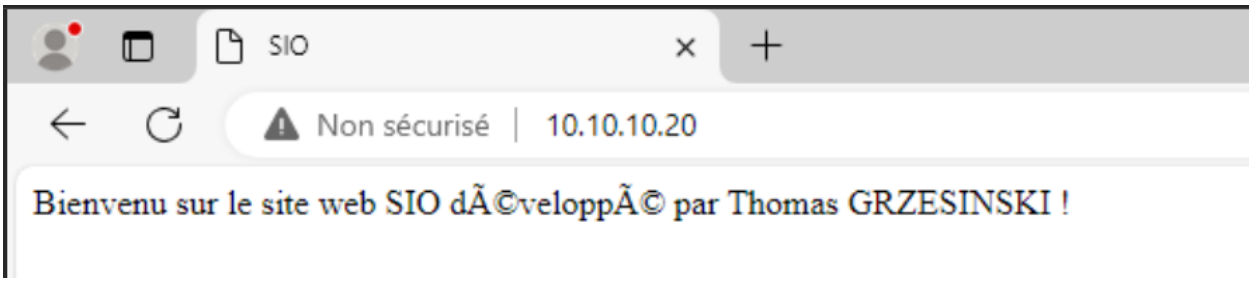
- Un IDS (Intrusion Detection System) détecte les activités suspectes sur un réseau sans les bloquer, tandis qu’un IPS (Intrusion Prevention System) agit en plus pour stopper les menaces, et nous allons travailler avec Suricata, un outil puissant qui peut fonctionner comme IDS et IPS pour analyser et sécuriser le trafic réseau.

Schéma réseau :



1 Installation et configuration de l’infrastructure

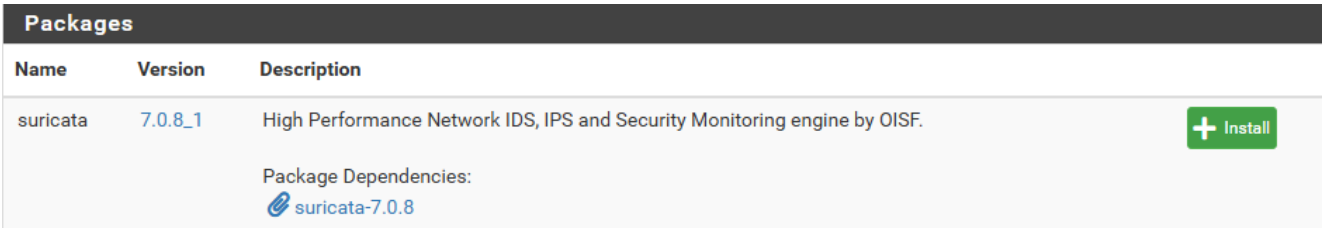
- Tuto de la mise en place du pfsense sur mon portfolio : https://thomas-portfolio.fr/TP-pfSense.pdf
- Vérification que le client accède au serveur web :



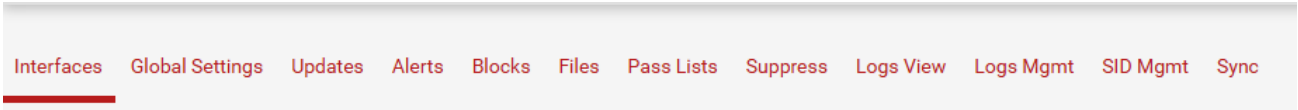
---

2 Installation et configuration de Suricata sur pfSense

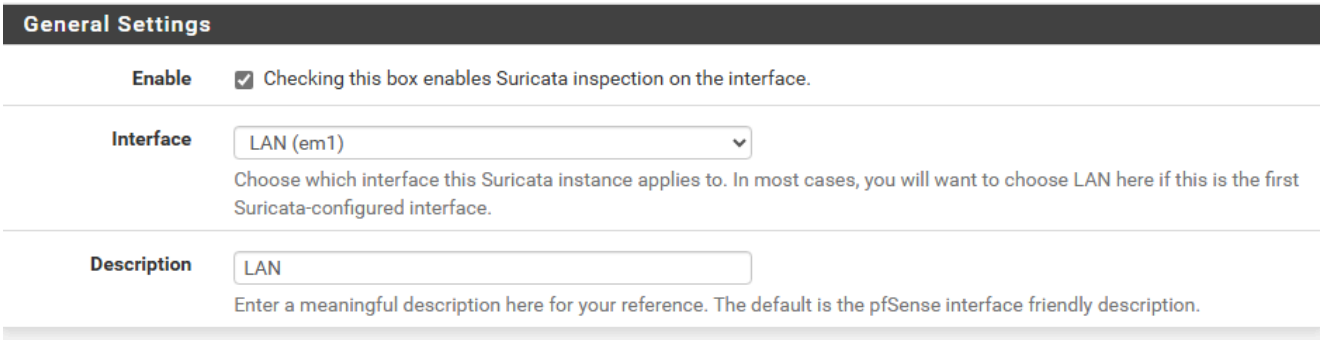
- Installation de suricata :
  - Sur pfSense aller dans Systeme -> Package manager -> Available Packages -> recherché et installé suricata :



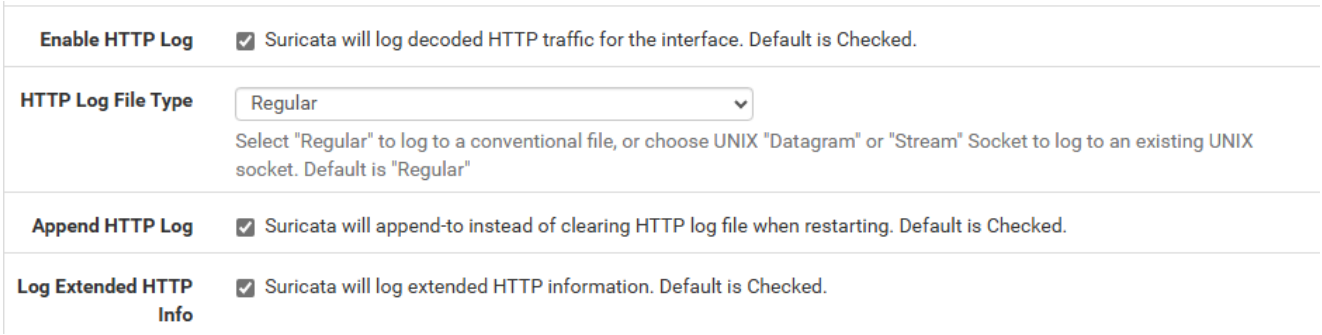
  - Une fois installé Suricata installé aller dans “Services -> Suricata”
  - Aller dans l’onglet “Global Settings”



- Nous allons donc maintenant passé à la configuration de Suricata, je vous recommande fortement de suivre ce tuto qui explique trés clairement comment le configurer : https://www.ctechmat.fr/pfsense-paquet-suricata/
- Pour la première partie de configuration je vous invite donc a suivre ce tuto : https://www.ctechmat.fr/pfsense-paquet-suricata/
- Ensuite il vous faudra cliquer sur Interfaces -> Ajouter chosir l’interface LAN



- Il vous faudra ensuite renseigné que l’on souhaite “surveillé” le protocole http qui circule



Promiscuous Mode ☑ Suricata will place the monitored interface in promiscuous mode when checked. Default is Checked.

- Une fois que ceci est fait vous pouvez aller dans “LAN Rules” et vous pouvez observé qu’une multitue de règles existent. Vous pouvez décidez de les activer ou désactiver.

Rule Signature ID (SID) Enable/Disable Overrides

Legend: Default Enabled · Enabled by user · Auto-enabled by SID Mgmt · Action/content modified by SID Mgmt · Rule action is alert · Rule contains noalert option · Default Disabled · Disabled by user · Auto-disabled by SID Mgmt

State	Action	GID	SID	Proto	Source	SPort	Destination	DPort	Message
✓	⚠	1	2260000	ip	any	any	any	any	SURICATA Applayer Mismatch protocol both directions
✓	⚠	1	2260001	ip	any	any	any	any	SURICATA Applayer Wrong direction first Data
✓	⚠	1	2260002	ip	any	any	any	any	SURICATA Applayer Detect protocol only one direction
✓	⚠	1	2260003	ip	any	any	any	any	SURICATA Applayer Protocol detection skipped
✓	⚠	1	2260004	tcp	any	any	any	any	SURICATA Applayer No TLS after STARTTLS
✓	⚠	1	2260005	tcp	any	any	any	any	SURICATA Applayer Unexpected protocol

- Une fois que vous avez configuré la base de suricata, retournez sur l’interface et activé la :

Interface Settings Overview

	Interface	Suricata Status	Pattern Match	Blocking Mode	Description	Actions
☐	LAN (em1)	✓ ↻ ◉	AUTO	DISABLED	LAN	✎ ⧉ 🗑

- Vérifier que l’IDS de Suricata est aussi actif en allant dans Status -> Services :

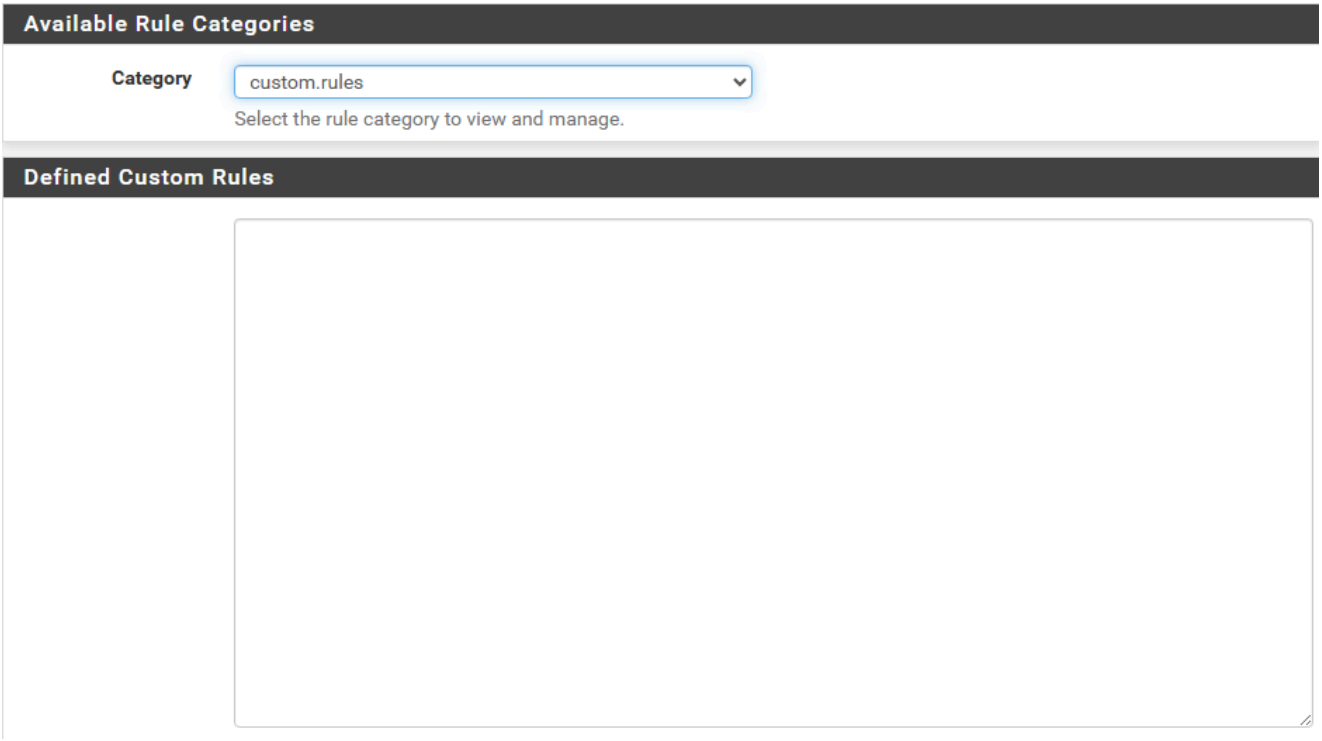
suricata	Suricata IDS/IPS Daemon	✓	↻ ◉

3 Comment fonctionne les règles sur Suricata ?

Section	Description
Format des Règles	Les signatures dans Suricata sont utilisées pour identifier et analyser des comportements suspects.
Règles Composants	Une règle se compose de 3 parties : action, en-tête, options.
Exemple de Règle	alert tcp any any -> $HOME_NET 80 (msg:"Tentative de connexion HTTP détectée"; sid:100001; rev:1;)
Action	Définit ce qui se passe lorsqu'une règle correspond (alert, pass, drop, reject).
Protocole	Spécifie le protocole concerné : TCP, UDP, ICMP, IP, ou protocoles d'application comme HTTP, DNS, FTP.
Source et Destination	Définit les adresses IP source et destination, utilisant des variables comme $HOME_NET et $EXTERNAL_NET.
Ports	Définit les ports source et destination (ex : HTTP sur port 80) avec des opérateurs (port ranges, exclusions).
Direction	Indique la direction de l'évaluation : -> (source -> destination), <-> (les deux directions).
Options de Règles	Les options de règle (msg, content, flow) sont séparées par des points-virgules et définissent les critères spécifiques.
Modificateurs de Contenu	ContenuCertains modificateurs comme les "content modifiers" ou "sticky buffers" influencent l'évaluation de la règle.
Buffers Normalisés	Les données des paquets sont "normalisées" pour éliminer les contenus anormaux, utilisées par certaines options comme http.uri.
Protocoles d'Application	Liste des protocoles d'application disponibles comme HTTP, FTP, DNS, SMB, etc. Cela dépend des paramètres dans suricata.yaml.
Exemples de Syntaxe	!1.1.1.1 (tous les IP sauf 1.1.1.1), $HOME_NET (réseau interne), [80, 81, 82] (ports 80, 81 et 82).

3.2 Création de règles de détection

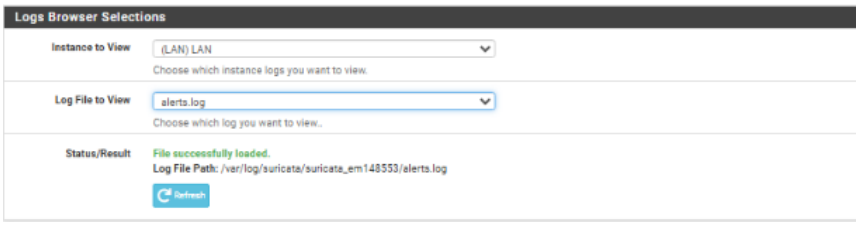
- Nous allons donc maintenant créer diffèrents règles afin d'utilisé et exploité le service
- Retournez dans Services -> Suricata -> Interfaces -> LAN - > LAN Rules
- Sur category modifié et choisissez "custom.rules" pour appliquer vos propres règles



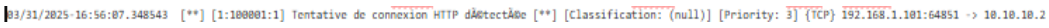
- Règle nuémero 1 : Écrire une règle pour détecter les tentatives de connexion HTTP au serveur Apache :
- Règle : alert tcp any any -> $HOME_NET 80 (msg:"Tentative de connexion HTTP détectée"; sid:100001; rev:1;)

- Explication de la règle : Cette règle Snort, « alert tcp any any -> $HOME_NET 80 (msg:“Tentative de connexion HTTP détectée”; sid:100001; rev:1;) », génère une alerte dès qu’une connexion TCP, émanant de n’importe quelle adresse et port, est tentée vers le port 80 (HTTP) d’une machine du réseau interne défini par la variable $HOME_NET, avec le message d’alerte spécifié, un identifiant unique (sid 100001) et une révision (rev:1).

- Pour vérifier si la règle fonctionne aller dans “Logs view” choisissez l’interface LAN et le fichier d’alerts.log



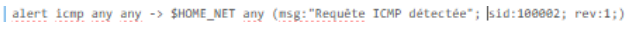
- Recherché ensuite une alerte avec votre message parmis les autres nombreuses alrtes:



- Explication de l’alerte : L’alerte affiche la date et l’heure de l’événement, ainsi que l’adresse IP de la machine source et celle de la machine distante (DMZ), en intégrant le message personnalisé souhaité et une classification précise de la gravité de l’alerte.

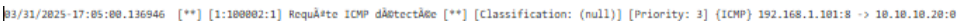
Règle nuémro 2 : Écrire une règle pour détecter les requêtes ICMP (ping) vers le serveur :

Règle : alert icmp any any -> $HOME_NET any (msg:“Requête ICMP détectée”; sid:100002; rev:1;)



- Explication de la règle : Cette règle Suricata détecte toute requête ICMP émanant de n’importe quelle adresse source vers n’importe quelle machine du réseau interne ($HOME_NET), générant ainsi une alerte avec le message “Requête ICMP détectée” et identifiée par le sid 100002 et la révision 1.

- Vérification :

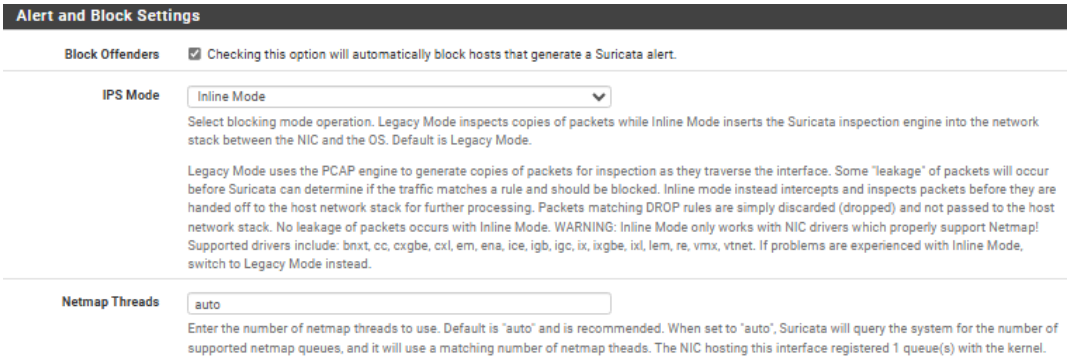


- Explication de l’alerte : L’alerte affiche la date et l’heure de l’événement, ainsi que l’adresse IP de la machine source et celle de la machine distante (DMZ), en intégrant le message personnalisé souhaité et une classification précise de la gravité de l’alerte.

4 Passage en mode IPS et tests

- Nous allons maintenant mettre en place des règles qui permettent de bloqué toutes connexions suspectes HTTP

- Nous devons d’abord allez dans LAN settings et activé le paramètre le paramètre ” Black offenders”



- Explication : Les réglages “alert” et “block” dans Suricata servent à deux objectifs complémentaires : d’une part, les alertes permettent d’identifier et de consigner des événements ou des comportements réseau suspects pour analyse, et d’autre part, les paramètres de blocage (lorsque Suricata est utilisé en mode inline) permettent d’intervenir activement en arrêtant ou en bloquant le trafic malveillant détecté, renforçant ainsi la sécurité du réseau.

- Nous pouvon maintenant mettre en place nos nouvelles règles

Règle numéro 3 : Modifier les règles pour bloquer certaines activités :

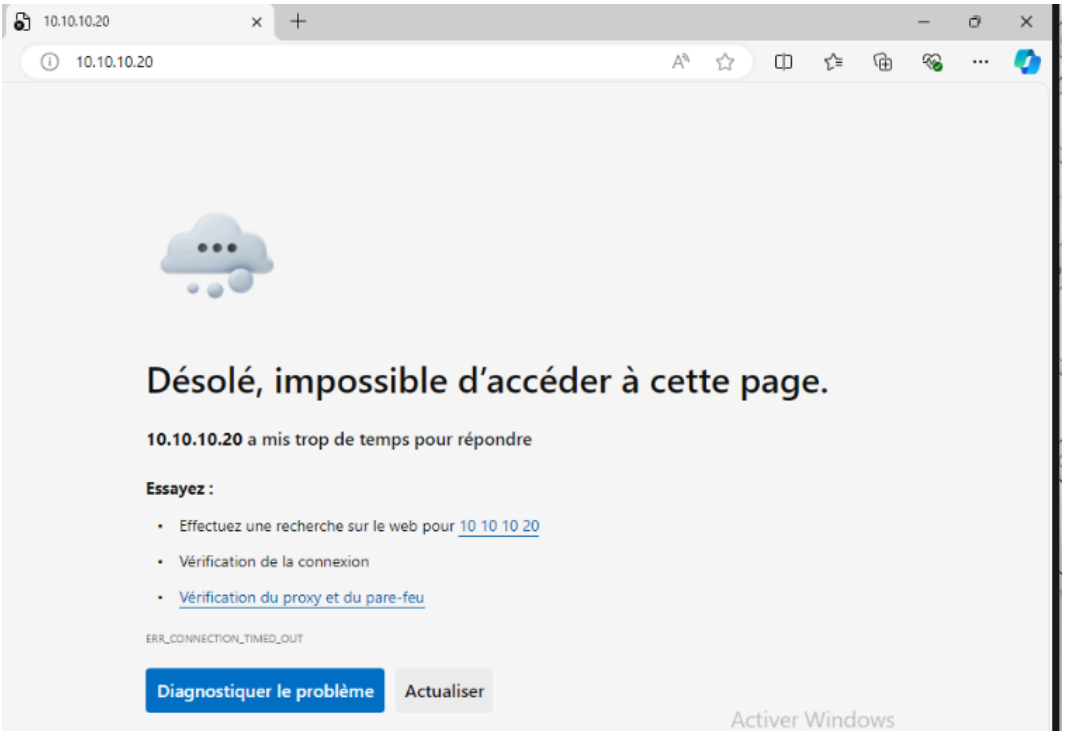
Règle : drop ip any any -> 10.10.10.20 80 (msg:“Connexion HTTP bloquée; sid 100001; priority:1;)

```
drop ip any any -> 10.10.10.20 80 (msg:"Connexion HTTP bloquée"; sid:1000001; priority:1;)
```

- Explication de la règle : Cette règle Snort, « drop ip any any -> 10.10.10.20 80 (msg:“Connexion HTTP bloquée; sid 100001; priority:1;) », bloque tout trafic IP émanant de n’importe quelle adresse et port source, destiné à l’IP 10.10.10.20 sur le port 80 (HTTP), en générant une alerte avec le message “Connexion HTTP bloquée”, identifiée par le sid 100001 et classée avec une priorité de 1.

- Vérification :

- Sur le client :



- Sur le serveur :

```
03/31/2025-17:27:07.272717  [wDrop] [**] [1:100003:1] Connexion HTTP bloquÃ©e [**] [Classification: (null)] [Priority: 3] {TCP} 192.168.1.101:64970 -> 10.10.10.20:80
```

- Explication de l’alerte : L’alerte affiche la date et l’heure de l’événement, ainsi que l’adresse IP de la machine source et celle de la machine distante (DMZ) sur le port 80, en intégrant le message personnalisé souhaité et une classification précise de la gravité de l’alerte. Tout en mettant au début de l’alerte le type de commande intégrée à la règle qui, en plus de générer une alerte, bloque activement le trafic correspondant en précisant le protocole TCP

- Avec wireshark :

```
    88 3.821781    192.168.1.101    10.10.10.20    TCP    66 [TCP Retransmission] 65099 → 80 [SYN] Seq=0 Win=64240 Len=0 MS…
    89 3.917642    192.168.1.101    10.10.10.20    TCP    66 [TCP Retransmission] 65100 → 80 [SYN] Seq=0 Win=64240 Len=0 MS…
    90 4.072842    192.168.1.101    10.10.10.20    TCP    66 [TCP Retransmission] 65101 → 80 [SYN] Seq=0 Win=64240 Len=0 MS…
```

- Explication de wireshark : Les paquets montrent des retransmissions SYN, indiquant que la machine source tente d’établir une connexion TCP sans recevoir de réponse de la destination.

- Règle nuémro 4 : Bloqué les pings

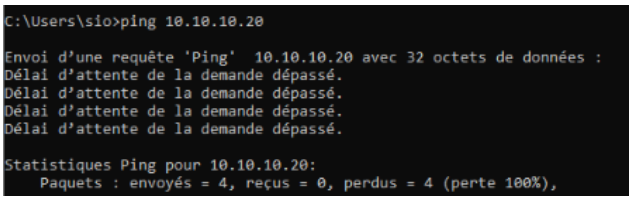
- Règle : drop icmp any any -> 10.10.10.20 any (msg:“Ping bloqué”; sid:100004; rev:1;)

```
drop icmp any any -> 10.10.10.20 any (msg:"Ping bloqué"; sid:100004; rev:1;)
```

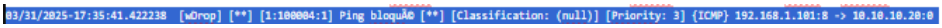
- Explication de la règle : Cette règle Snort, « drop icmp any any -> 10.10.10.20 any (msg:“Ping bloqué”; sid:100004; rev:1;) », bloque tout trafic ICMP, émanant de n’importe quelle adresse, destiné à l’IP 10.10.10.20, en générant une alerte avec le message “Ping bloqué”, identifiée par le sid 100004 et la révision 1.

- Vérification:

- Sur le client :



- Sur le serveur :

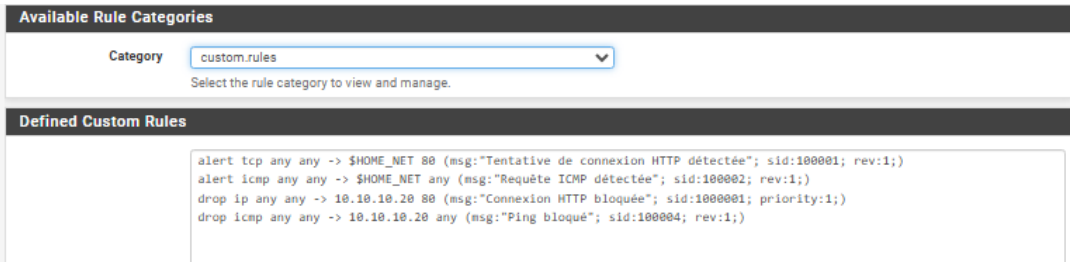


- Explication de l'alerte : L'alerte affiche la date et l'heure de l'événement, ainsi que l'adresse IP de la machine source et celle de la machine distante (DMZ) sur le port 80, en intégrant le message personnalisé souhaité et une classification précise de la gravité de l'alerte. Tout en mettant au début de l'alerte le type de commande intégrée à la règle qui, en plus de générer une alerte, bloque activement le trafic correspondant en précisant le protocole ICMP

- Avec wireshark :

- Explication de wireshark: Les paquets ICMP envoyés depuis la machine source indiquent que la tentative de ping est refusée par la destination.

5 Intégralités des règles



5.2 Amélioration des règles pour éviter les faux positifs et améliorer la détection.

- Amélioration règle numéro 1 : Limite l'alerte aux connexions HTTP établies, en filtrant spécifiquement les requêtes HTTP GET afin de réduire les faux positifs liés aux paquets non HTTP et ceux ne faisant pas partie d'une session établie.
  - Règle : alert tcp $EXTERNAL_NET any -> $HOME_NET 80 (msg:"Tentative de connexion HTTP détectée"; flow:to_server,established; http.method; content:"GET"; sid:100001; rev:2;)

- Amélioration règle numéro 2 : Limite l'alerte aux requêtes ICMP Echo Request (type 8) pour éviter de déclencher des alertes pour d'autres types ICMP comme les erreurs ou les réponses
  - Règle : alert icmp any any -> $HOME_NET any (msg:"Requête ICMP détectée"; itype:8; sid:100002; rev:2;)

- Amélioration règle numéro 3 : Cible spécifiquement une adresse IP interne et filtre les connexions HTTP vers cette IP pour éviter de bloquer toutes les connexions et limiter les faux positifs. Utilise également une action de priorité pour mieux gérer l'impact.
  - Règle : drop tcp any any -> 10.10.10.20 80 (msg:"Connexion HTTP bloquée"; sid:100003; priority:1; flow:to_server,established; http.method; content:"GET"; rev:2;)

- Amélioration règle numéro 4 : Limite la règle aux requêtes ICMP Echo Request (type 8) et cible une IP spécifique pour éviter de bloquer tous les pings et se concentrer sur les tentatives potentiellement malveillantes.
  - Règle : drop icmp any any -> 10.10.10.20 any (msg:"Ping bloqué"; sid:100004; itype:8; rev:2;)