TP Suricata

Table des matières

[TP Suricata 1](#_Toc132303146)

[**Suricata** 2](#_Toc132303147)

[Installation 2](#_Toc132303148)

[Configuration 3](#_Toc132303149)

[Mode IDS 4](#_Toc132303150)

[Mode service 5](#_Toc132303151)

[Mode IDS (optionnel) 6](#_Toc132303152)

# **Suricata**

Suricata est une sonde N.I.D.S / N.I.P.S facile d’installation et d’utilisation.

Dans ce TP, nous verrons comment installer Suricata sur une machine Debian et les premiers paramètres de cette sonde.

## Installation

#### VM

Dans un premier temps il faut créer un VM Debian avec une interface graphique.

Avec au minimum :

6 GB de RAM

20 GB

2 interfaces réseau.

1 connexion à internet.

#### Suricata

Pour télécharger Suricata :

sudo apt-get install suricata

Vérifiez que l’installation de Suricara s’est déroulée correctement :

Suricata – V

Vous devriez avoir le résultat suivant :

« This is Suricata version 6.0.1 RELEASE »

## Configuration

Le fichier de configuration de Suricata est le suivant :

vi /etc/suricata/suricata.yaml

Dans ce fichier, ce que vous pouvez immédiatement modifier est la variable : HOME\_NET

En remplaçant les ips donnés en exemple par celle de votre/vos réseaux.

Exemple : HOME\_NET : « [10.13.0.0/26] »

Le dossier comportant les fichiers de règles de Suricata se trouve ici :

*/var/lib/suricata/rules/*

Vous pouvez normalement y trouver un certain nombre de fichier de règles importé par Suricata. Ce sont les fichiers de base permettant de lancer Suricata avec une grande variété de règle.

Si on lance Suricata sans préciser le fichier de règle, Suricata va utiliser le fichier suivant : */var/lib/suricata/rules/suricata.rules*

Vérifier qu’il existe sinon créez-le :

vi /var/lib/suricata/rules/suricata.rules

Puis dans ce fichier créez une règle simple pour détecter tous les pings :

Alert icmp any any -> any any (msg : « ICMP Ping Detected”; sid:1000001; rev:1;)

Sauvegargez le fichier.

## Mode IDS

Vous pouvez à présent lancer la commande suivante pour lancer Suricata :



* -c : pour indiquer le fichier de configuration
* -i : pour indiquer sur quelle interface suricata va écouter
* -S : pour spécifier le fichier de règle pour Suricata (et ne pas récupérer les règles indiquées dans le fichier de configuration)
* -D : lance suricata en mode démon

Deux fichiers vont alors se créer :

*/var/log/suricata/eve.json* : fichier structuré et très volumineux comprenant tous les agissements sur le réseau que Suricata va détecter.

*/var/log/suricata/fast.log* : C’est le fichier qui va contenir toutes les alertes liés aux règles qu’on a indiqué à Suricata dans le fichier suricata.rules.

Pour vérifier si Suricata c’est correctement lancé, ouvrez un second terminal et tapez :

sudo suricata –T –c /etc/suricata/suricata.yaml

Cette commande affiche les instances de Suricata en cours d’exécution.

Si la commande vous indique en effet que Suricata fonctionne correctement, testez votre règle :

Toujours avec ce second terminal essayer de ping google.

Puis vérifiez dans le fichier le fichier d’alerte :

cat /var/log/suricata/fast.log

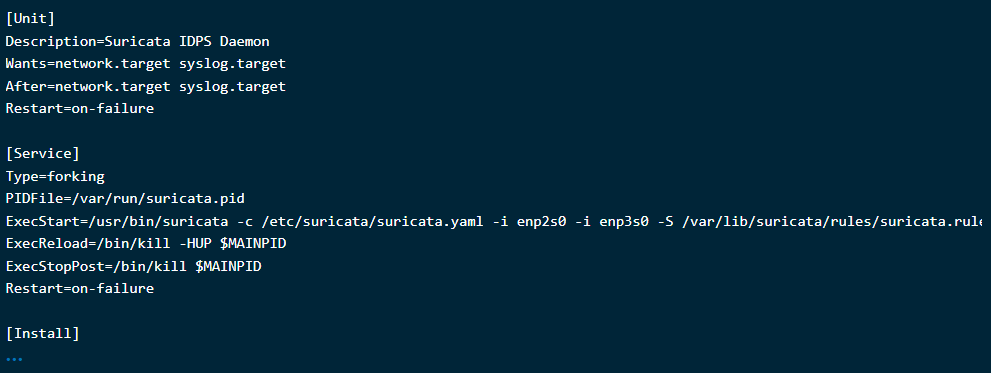
Vous devriez y retrouver les alertes de vos pings.

## Mode service

Vous pouvez ajouter Suricata en tant que service afin de le lancer ou même voir son statut plus facilement.

Pour cela vous devez créer le fichier : */etc/systemd/system/suricata.service*

Exemple :



**Attention :** La valeur ExecStart correspond à la commande que le system service va lancer avec la commande « start ». Il convient donc de modifier cette commande en fonction de votre environnement (interface, commande, rules, etc)

A présent, activez le service :



Enfin démarrez le :



On peut également le redémarrer :



L’arrêter :



Ou bien vérifier son statut :



## Mode IDS (optionnel)

Il se peut que vous souhaitez configurer Suricata pour agir en tant que IPS (Intervention Prevention System). C’est-à-dire que Suricata ne sera plus seulement capable de détecter les packets parcourant le réseau, mais également de bloquer, rejeter ou droper certains packets, potentiellement malveillant.

Avec un IPS on ajoute une couche de protection active au sein de son réseau, cependant cela peut également bloquer ou ralentir la production au sein de l’entreprise si l’IPS n’a pas été configuré avec soin car certains packets primordiaux pourraient être bloqués par erreur. C’est pour cela que je vous conseille d’être méticuleux et attentifs si vous décidez d’utiliser un IPS.

Suricata utilise NFQueue pour interagir comme IPS. NFQ est un filtreur réseau (netfilter) qui marche avec des tables ip (iptables).

Dans un premier temps, vérifier bien que suricata a été configuré et installé avec NFQ.

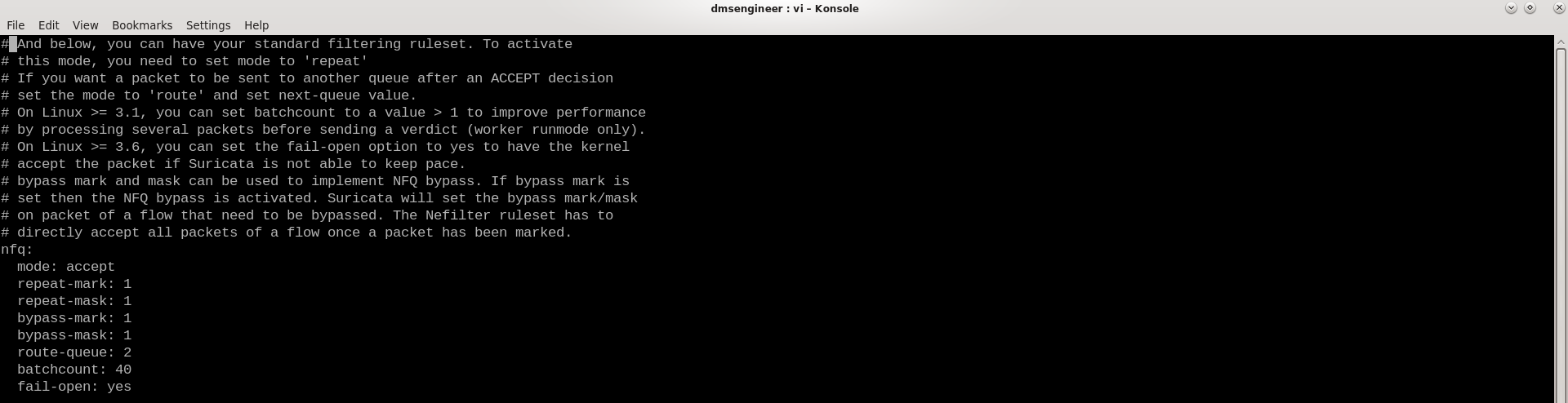
Tapez la commande :

suricata –build-info

Vérifiez ensuite sous la catégorie « Suricata Configuration » que le « NFQueue support » est coché « yes ».

Une fois le NFQueue validé Suricata, il faut légèrement modifier le ficher */etc/suricata/suricata.yaml*:

Trouver la ligne « nfq » et décommentez la ligne. Également les lignes doivent ressembler à ceci :



Pour cela il faut commencer par configurer des iptables NFQueue. Voici un exemple :

sudo iptables -I INPUT -p tcp -j NFQUEUE

sudo iptables -I OUTPUT -p tcp -j NFQUEUE

Ces commandes vont créer des iptables qui va superviser tous les paquets entrant et sortant en TCP. NFQueue va capturer tous les paquets qu’il va détecter, il va les stocker dans une « queue » qui va ensuite les transmettre à Suricata qui va devoir les filtrer en fonction de ces règles configurées.

De base c’est la queue 0 qui va être sélectionné. On peut préciser celle que l’on souhaite avec l’argument : --queue-num \*. (\* Étant le nombre désiré).

**Attention** : Tant que Suricata n’a pas autorisé les paquets, NFQueue empêche leur transmission. Par exemple si vous ne lancer pas Suricata avec alors que ces iptables sont actives, NFQueue bloquera toutes les communications en TCP car il restera en attente de la validation de Suricata, même si ce dernier n’est pas activé.

Cependant il existe un paramètre avec NFQueue qui permet de lui indiquer de laisser les paquets si Suricata ne réponds pas : --queue-bypass***.*** Ce permets d’empêcher de bloquer tout votre réseau si jamais Suricata rencontre un problème. Cependant cela signifie également que si jamais quelqu’un parvient à éteindre Suricata, votre réseau ne sera plus filtré.

Vous pouvez également indiqué à NFQueue que vous souhaitez filtrer un protocole en particulier par exemple :

sudo iptables -I INPUT -p tcp --sport 80 -j NFQUEUE