AMOSSYS

Open Information Security Foundation

Cible de sécurité CSPN

Produit Suricata version 6.0.8

Catégorie « Détection d'intrusion »

Référence : CSPN-ST-Suricata-1.02

Date : le 29/09/2022

Code interne : ANS020-04-AVT01

Siège : Immeuble Le Ouessant • Bâtiment B • 11 rue Maurice Fabre • 35000 Rennes • France • **www.amossys.fr**
SIRET : 493 348 890 00051 • **NAF** : 6202 A • RCS Rennes B 493 348 890 • SAS au capital de 38.000 Euros

FICHE D'EVOLUTIONS

Révision	Date	Description	Rédacteur(s)
1.00	23/03/2022	Création du document	P. GAUTIER
1.01	02/05/2022	Modification de la version du produit	M. VOGT
	13/09/2022	Prise en compte de la dernière version (6.0.6) et des remarques de l'ANSSI	J. LEMETEYER
1.02	29/09/2022	Prise en compte de la dernière version (6.0.8)	J. LEMETEYER

Ce document a été validé par l'ANSSI.

SOMMAIRE

1. INTRODUCTION 4
1.1. Objet du document 4
1.2. Identification du produit 4
1.3. Références 4
2. DESCRIPTION DU PRODUIT 5
2.1. Description générale 5
2.2. Principe de fonctionnement 5
2.3. Description des dépendances 7
2.4. Description de l'environnement technique de fonctionnement 7
2.4.1. Matériel compatible ou dédié 7
2.4.2. Système d'exploitation retenu 7
2.5. Périmètre de l'évaluation 7
2.5.1. Périmètre 7
2.5.2. Plateforme d'évaluation 8
3. PROBLEMATIQUE DE SECURITE 9
3.1. Description des utilisateurs typiques 9
3.2. Description des biens sensibles 9
3.3. Description des hypothèses sur l'environnement 10
3.4. Description des menaces 10
3.5. Description des fonctions de sécurité 11
3.6. Matrices de couvertures 11
3.6.1. Menaces et biens sensibles 11
3.6.2. Menaces et fonctions de sécurité 12

1. INTRODUCTION

1.1. OBJET DU DOCUMENT

Ce document est réalisé dans le cadre de l'évaluation, selon le schéma CSPN[1] promu par l'ANSSI[2], du produit « Suricata » développé par la société **Open Information Security Foundation**.

La TOE[3] considérée est Suricata dans sa dernière version au démarrage de l'évaluation.

Ce document est soumis au contrôle technique et qualité d'**AMOSSYS** ainsi qu'à la validation de l'ANSSI, en tant que commanditaire. Les mises à jour de ce document sont effectuées par l'équipe projet d'**AMOSSYS**.

1.2. IDENTIFICATION DU PRODUIT

Éditeur	**Open Information Security Foundation**
Lien vers l'organisation	https://oisf.net
Nom commercial du produit	Suricata
Numéro de la version évaluée	Dernière version à jour (6.0.8, au moment de la rédaction du présent document)
Catégorie du produit	Détection d'intrusion

1.3. REFERENCES

Pour l'établissement de la présente cible de sécurité, les documents suivants ont été consultés par le rédacteur :

- la documentation accessible sur le site de l'éditeur : https://suricata-ids.org/docs/;
- la documentation présente sur le site readthedocs.io : https://suricata.readthedocs.io/en/suricata-6.0.4/;
- la documentation présente sur le site readthedocs.io : https://suricata.readthedocs.io/en/suricata-6.0.6/;
- la documentation présente sur le site readthedocs.io : https://suricata.readthedocs.io/en/suricata-6.0.8/.

[1] Certification de Sécurité de Premier Niveau
[2] Agence Nationale de la Sécurité des Systèmes d'Information
[3] *Target Of Evaluation*

2. DESCRIPTION DU PRODUIT

2.1. DESCRIPTION GENERALE

Suricata est un système de détection d'intrusions réseau (« *Network Intrusion Detection System* ») *open source*, disponible pour les systèmes d'exploitation de type Unix (dont Linux, FreeBSD et OpenBSD), qui analyse le trafic réseau à la recherche de toute activité suspecte, en se basant sur des règles de détection.

Suricata peut être utilisé en dérivation, dans ce cas-là, son rôle est passif : il est uniquement chargé d'analyser des flux réseaux qui lui sont soumis et de générer des alertes. Dans le cas où il est en mode « inline », Suricata a la possibilité de bloquer les flux réseaux. Il s'agit alors d'un IPS.

Suricata implémente un langage de signature complet pour identifier des menaces connues, des violations de règles ou encore des comportements malveillants par l'analyse du trafic qu'il inspecte.

Le produit dispose des fonctionnalités suivantes :

- Haute performance par la gestion du *multi-threading*, l'utilisation des GPU (accélération graphique) et le support de l'accélération matérielle à travers PF_RING et AF_PACKET ;
- la détection automatique et l'analyse des protocoles :
  - décodage de paquets IPv4/6, TCP, UDP, SCTP, ICMP, GRE, Ethernet, PPP, PPPoE, Raw, SLL, VLAN, QINQ, PLS, ERSPAN ;
  - décodage des couches applicatives HTTP, SSL/TLS, FTP, TFTP, SMB, SMTP, SSH, DNS, DCERPC, Modbus, ENIP/CIP, DNP3, NFS, NTP, DHCP, KRB5, IKEv2 ;
- NSM : journalisation des requêtes HTTP, journalisation et analyse des échanges SSL/TLS, stockage des certificats, extraction de fichiers, etc. ;
- l'utilisation de la bibliothèque HTP indépendante pour le moteur HTTP ;
- la prise en charge des formats d'entrée standards ;
- la prise en charge de nombreux formats de sortie notamment pour les alertes (JSON, syslog, fast.log, Unified2, Prelude, PCAP, etc.) ;
- la décompression des archives GZIP ;
- la gestion de variables pour la sauvegarde d'information des flux ;
- un système d'identification rapide d'adresse IP.

2.2. PRINCIPE DE FONCTIONNEMENT

Fonctionnement général

Suricata est un moteur IDS/IPS utilisant des jeux de règles pour surveiller le trafic réseau et fournir des alertes à l'administrateur quand un événement suspect intervient.

La gestion du jeu de signatures ou encore les interfaces de consultation des alertes ne font pas partie de son champ d'action. Pour cela, le moteur a été développé pour être compatible avec les composants de sécurité réseau existants et s'intègre dans le SI comme tout autre produit de type IDS/IPS.

La Figure 1 présente un cas typique d'utilisation de Suricata.

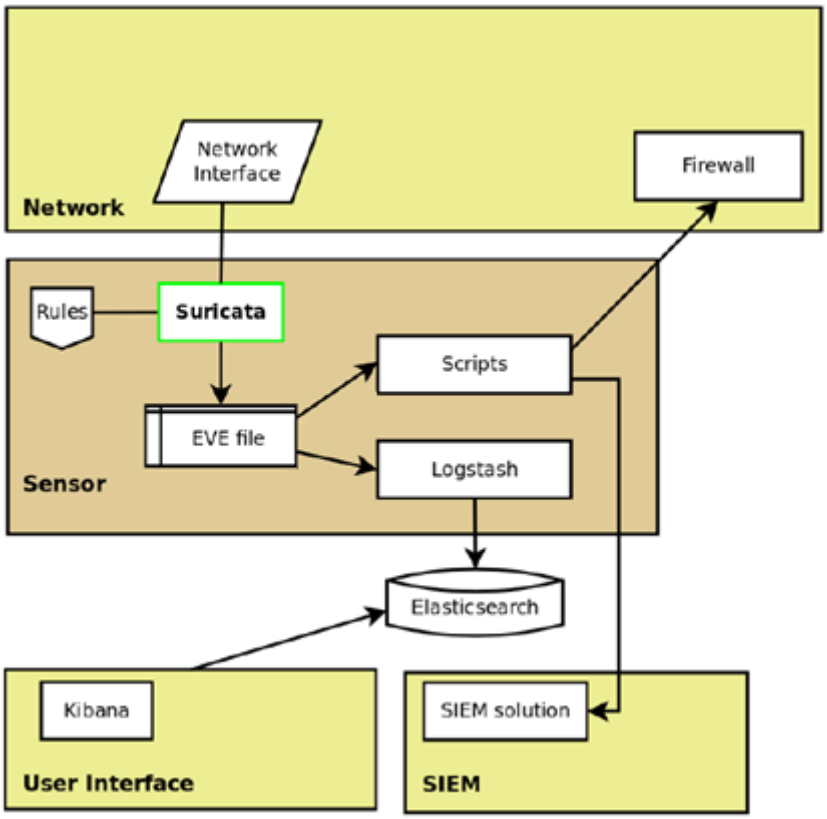

Figure 1 – Exemple d'intégration de Suricata

Sur cette figure les flèches permettent de montrer les relations entre les composants. Cela met en évidence le fait que Suricata n'est pas responsable de la réaction suite à une détection d'une menace et ne dispose pas d'interface graphique. En effet, le produit prend en entrée les flux présents sur son interface réseau et les analyse au regard de règles qui lui ont été définies. En sortie, un fichier JSON (EVE.json) est créé. Dans le cas présenté ci-dessus, celui-ci est ensuite fourni à une pile ELK[4] afin d'être analysé par un exploitant.

Installation

Sous Linux, le produit s'installe et se configure en ligne de commande. Après installation, il est nécessaire de configurer des règles pour que l'outil fonctionne.

S'il est possible de télécharger des règles prédéfinies et de les installer manuellement, il est aussi possible d'utiliser un programme annexe permettant de simplifier cette tâche. L'outil officiel pour la gestion des règles **Suricata** est *Suricata-Update*.

D'autres programmes comme *Pulled Pork* et *Oinkmaster* permettent également de récupérer automatiquement des ensembles de règles prédéfinies mises à jour (disponible par exemple à cette adresse :

https://rules.emergingthreats.net/open/suricata-<version>/emerging.rules.tar.gz).

La configuration de Suricata se fait au moyen d'un fichier au format YAML qui contient l'ensemble des directives de configuration dont la liste des fichiers de signatures à utiliser.

[4] https://www.elastic.co/fr/elk-stack

2.3. DESCRIPTION DES DEPENDANCES

Les bibliothèques nécessaires au bon fonctionnement du produit sont :

- **libpcre3** : librairie de fonctions utilisant la même syntaxe et sémantique que Perl ;
- **libpcap** : capture des paquets réseaux ;
- **libnet** : API générique réseau ;
- **libyaml** : parseur et émetteur YAML ;
- **zlib1g** : bibliothèque implémentant la méthode de compression gzip et PKZIP
- **Rust :** à partir de la version 5, la prise en charge de Rust est nécessaire.

Par défaut, Suricata fonctionne en tant qu'IDS. Pour fonctionner en tant qu'IPS, de nouvelles librairies doivent être installées :

- **libnfnetlink** : la bibliothèque de bas niveau *libnfnetlink* permet les communications noyau/espace utilisateur liées à *netfilter*. Elle fournit une infrastructure de messages générique pour les sous-systèmes *netfilter* intégrés au noyau (comme nfnetlink_log, nfnetlink_queue et nfnetlink_conntrack) et leurs utilisateurs et/ou outils de gestion en espace utilisateur.
- **libnetfilter-queue** : librairie userland, requise par *Libnfnetlink*, fournissant une API pour les paquets qui ont été mis en attente par le noyau packet filter.

Des outils tels que *make*, *automake* ou *autoconf* sont nécessaires pour la compilation et l'installation du produit.

2.4. DESCRIPTION DE L'ENVIRONNEMENT TECHNIQUE DE FONCTIONNEMENT

2.4.1. Matériel compatible ou dédié

Suricata peut être installé sur une plateforme munie d'un système Ubuntu, Debian, CentOS, Fedora, OpenSuse, FreeBSD, Mac OS X ou Windows.

La plateforme doit disposer au minimum d'une interface réseau. Il est par contre recommandé d'utiliser deux interfaces : une capturant les paquets et l'autre dédiée à l'administration du produit.

2.4.2. Système d'exploitation retenu

Le système d'exploitation retenu pour la réalisation de l'étude est un système Debian 10 à jour des correctifs de sécurité, au démarrage de l'évaluation.

2.5. PERIMETRE DE L'EVALUATION

2.5.1. Périmètre

L'évaluation portera sur la dernière version stable de **Suricata** sur Debian 10 et **se focalisera sur la fonctionnalité de détection d'intrusions**. Dans ce contexte, **Suricata** est uniquement responsable de la capture des flux, de leur reconstruction protocolaire et de l'extraction des données.

L'analyse s'articulera surtout autour de :

- la fonctionnalité de reconstruction protocolaire ; les protocoles suivants sont alors considérés : IPv4, IPv6, ICMP et TCP ;

- la fonctionnalité d'extraction des métadonnées; les protocoles suivants sont alors considérés : DNS et TLS. Pour cette évaluation, les versions Rust des analyseurs seront considérées. L'objectif est de vérifier la bonne extraction des métadonnées par les dissecteurs.

L'installation du produit se fera en suivant le guide d'installation du produit. Notamment, l'évaluateur mettra en œuvre les **mécanismes de durcissement présents dans la documentation**.

Bien qu'une fonctionnalité de notification **et journalisation** soit présente, l'analyse de cette fonction se limitera à vérifier qu'elle existe.

Suricata dispose d'un mécanisme d'exécution de script Lua. Or, cette fonctionnalité est seulement offerte aux administrateurs du produit qui sont considérés de confiance. C'est pourquoi cette fonctionnalité n'est pas intégrée à la cible de sécurité.

La fonctionnalité d'*offloading* de la carte d'acquisition est désactivée.

2.5.2. Plateforme d'évaluation

La Figure 2 présente une architecture qui pourra être mise en place pour l'évaluation. Celle-ci est composée principalement :

- de serveurs ;
- de postes clients requêtant les serveurs ;
- de Suricata positionné en dérivation des flux entre les serveurs et les postes clients.

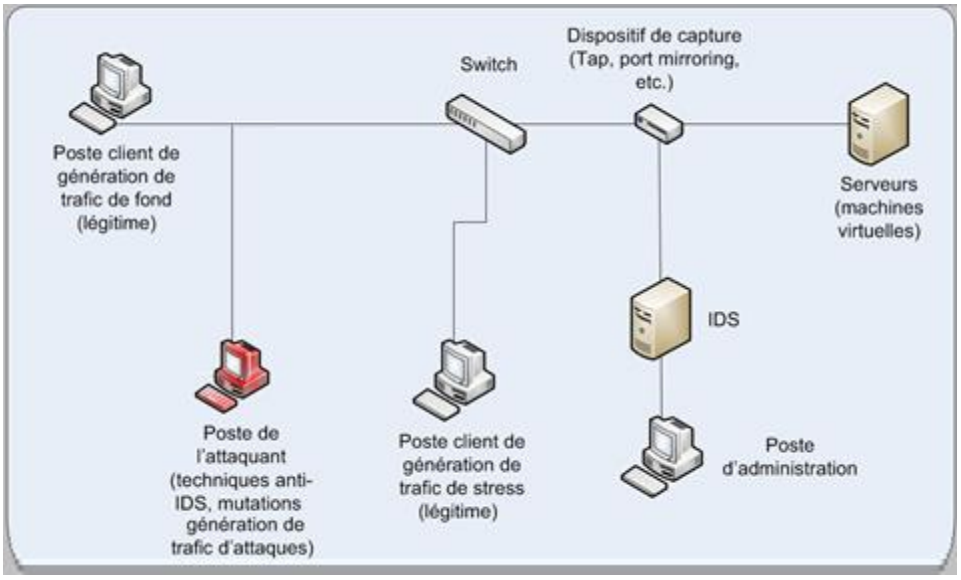

Figure 2 – Exemple de plateforme

Suricata sera compilé sans le support de la bibliothèque imagemagick. La méthode d'acquisition sera suffisante pour traiter les flux entrants**. L'évaluateur se basera ensuite sur les recommandations de configuration fournies par la documentation officielle du produit.**

3. PROBLEMATIQUE DE SECURITE

3.1. DESCRIPTION DES UTILISATEURS TYPIQUES

Par définition, les utilisateurs concernent les personnes et services applicatifs qui interagissent avec le produit évalué.

Les rôles suivants doivent être pris en considération dans le cadre de l'évaluation de sécurité :

- **l'administrateur**. Dans cette évaluation, il a été choisi de considérer que l'administrateur système et l'administrateur fonctionnel de **Suricata** sont les mêmes.
- **L'utilisateur légitime** présent sur le réseau. Cet utilisateur génère du trafic qui sera redirigé vers la TOE.

3.2. DESCRIPTION DES BIENS SENSIBLES

Par définition, un bien sensible est une donnée (ou fonction) jugée comme ayant de la valeur par la TOE. Sa valeur est estimée selon des critères de sécurité (aussi appelés besoins de sécurité) : disponibilité, intégrité, confidentialité et authenticité.

Les biens à protéger sont les suivants :

- B1.CONFIGURATION ET BASE DE REGLES / SIGNATURES

Les fichiers de configuration de l'IDS ainsi que les règles et signatures définies.

Besoin de sécurité : disponibilité, intégrité

- B2.FLUX RESEAU

Les flux réseau devant être analysés par Suricata. Ceux-ci sont reconstitués par Suricata, avant analyse.

Besoin de sécurité : intégrité, disponibilité

- B3.METADONNEES RESEAU

Suricata analyse les flux afin d'en extraire des métadonnées réseau.

Besoin de sécurité : intégrité, disponibilité

- B4.JOURNAUX ET ALERTES

Des alertes basées sur les signatures de détection sont remontées par l'IDS afin d'être traitées par un exploitant. En outre, les actions mises en œuvre par le produit sont consignées dans les journaux.

Besoin de sécurité : intégrité, disponibilité, confidentialité

Biens sensibles	Disponibilité	Intégrité	Confidentialité
Configuration et base de règles / signatures	√	√	
Flux réseau	√	√	
Métadonnées réseau	√	√	

Biens sensibles	Disponibilité	Intégrité	Confidentialité
Journaux et alertes	√	√	√

Remarque : Le bien sensible B1 ne fait pas l'objet de besoin en confidentialité. En effet, il est considéré que ce n'est pas à la TOE d'assurer ce besoin. Le cas échéant, la confidentialité est assurée via la mise en œuvre d'une architecture sécurisée réseau sécurisée (par exemple, un TAP assurant une fonctionnalité de diode serait alors présente).

3.3. DESCRIPTION DES HYPOTHESES SUR L'ENVIRONNEMENT

Par définition, les hypothèses sont des déclarations portant sur le contexte d'emploi de la TOE ou de son environnement.

Les hypothèses sur l'environnement de la TOE suivantes doivent être considérées :

- H1.SYSTEME-EXPLOITATION

Le système d'exploitation, support de la TOE, met en œuvre des mécanismes de protection adéquats (confinement, contrôle d'accès, etc.) paramétrés et configurés selon les règles de l'état de l'art. De plus, il est à jour des correctifs en vigueur au moment de l'installation, sain et exempt de virus, chevaux de Troie, etc.

- H2.LOCAUX

Les serveurs hébergeant le produit se trouvent dans des locaux sécurisés dont l'accès est contrôlé et restreint au personnel autorisé.

- H3.INSTALLATION-INTEGRE

Les outils nécessaires à l'installation de la TOE sont intègres.

- H4.ADMINISTRATEURS

Les administrateurs de l'IDS sont des personnes de confiance. Ils sont formés pour administrer et configurer le produit.

3.4. DESCRIPTION DES MENACES

Pour cette évaluation un seul type d'agent de menace est à considérer : un attaquant présent sur le réseau et en capacité de générer du trafic qui passera par l'interface de capture de Suricata. Cet attaquant ne dispose pas de droits particuliers.

Les menaces qui portent sur les biens sensibles de la TOE sont les suivantes :

- M1.CONTOURNEMENT RECONNAISSANCE SIGNATURES

Un attaquant contourne le système de reconnaissance de signatures dans le but de réaliser une attaque indétectable par la TOE.

- M2.DENI DE SERVICE

Un attaquant parvient à corrompre les données utiles à la TOE afin de perturber son bon fonctionnement.

- M3.PRISE DE CONTROLE

Un attaquant réussit à obtenir des accès en lecture et/ou écriture sur le serveur sur lequel est installée la TOE.

3.5. DESCRIPTION DES FONCTIONS DE SECURITE

Par définition, les fonctions de sécurité sont l'ensemble des mesures techniques et mécanismes mis en œuvre dans la TOE pour protéger de façon proportionnée les biens sensibles de la TOE contre les menaces identifiées.

Les fonctions de sécurité essentielles de la TOE sont les suivantes :

- F1.INNOCUITE

Le produit ne dispose pas de fonctionnalité portant atteinte au réseau protégé. De plus, il dispose d'une architecture ou de mécanismes d'autoprotection lui permettant de ne pas être un point d'entrée sur le réseau pour un attaquant.

- F2.AUTOPROTECTION

Les fonctionnalités métiers (telle que la reconstruction des flux protocolaires) de la TOE ne doivent pas créer de vulnérabilité sur la TOE

- F3.EXTRACTION DES METADONNEES

Après avoir reconstruit les flux au niveau transport, Suricata dispose de *parsers* par protocole applicatif. Ceux-ci lui permettent d'extraire des métadonnées qui seront par la suite confrontées aux règles de détection afin d'identifier des menaces, ou journalisées.

- F4.JOURNALISATION ET NOTIFICATION

Lorsqu'une menace est détectée, une alerte est remontée par le produit. De plus, une trace de l'ensemble des événements est journalisée.

3.6. MATRICES DE COUVERTURES

3.6.1. Menaces et biens sensibles

La matrice suivante présente la couverture des menaces sur les biens sensibles (les lettres "D", "I", "C" et "A" représentent respectivement les besoins de Disponibilité, Intégrité, Confidentialité et Authenticité) :

	Configuration et base de règles / signatures	Flux réseau	Métadonnées réseau	Journaux et alertes
Contournement reconnaissance signatures				**DI**
Déni de service	**DI**		**D**	**D**
Prise de contrôle	**DCI**	**I**	**IC**	**IC**

Tableau 1 - Couverture des biens sensibles par les menaces

3.6.2. Menaces et fonctions de sécurité

La matrice suivante présente la couverture des menaces par les fonctions de sécurité :

	Innocuité	Autoprotection	Extraction des métadonnées	Journalisation et notification
Contournement reconnaissance signatures				√
Déni de service		√	√	√
Prise de contrôle	√	√	√	√

Tableau 2 - Couverture des menaces par les fonctions de sécurité

Fin du document