Comprendre les règles Snort3

Table des matières

Introduction

Ce document décrit les règles de **Snort3** moteur de la gamme Cisco **Secure Firewall Threat Defense (FTD)**.

Conditions préalables

Exigences

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- Cisco **Secure Firewall Threat Defense (FTD)**
- **Intrusion Prevention System (IPS)**
- **Snort2** syntaxe

Licences

Aucune condition de licence spécifique, la licence de base est suffisante et les fonctionnalités mentionnées sont incluses dans le moteur **Snort** dans le FTD et dans les versions open source de **Snort3**.

Composants utilisés

Les informations contenues dans ce document sont basées sur les versions de matériel et de logiciel suivantes :

- Cisco **Secure Firewall Threat Defense (FTD)**, Cisco **Secure Firewall Management Center (FMC)** version 7.0+

avec **Snort3**.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. Si votre réseau est en ligne, assurez-vous de bien comprendre l'incidence possible des commandes.

Informations générales

Snort est le moteur Cisco IPS capable d'analyser le trafic en temps réel et de consigner les paquets.

Snort peut analyser les protocoles, rechercher du contenu et détecter les attaques.

Snort3 est une version mise à jour du Snort2 IPS avec une nouvelle architecture logicielle qui améliore les performances, la détection, l'évolutivité et la convivialité.

Règles Snort3

Ils utilisent ce format LUA pour créer le **Snort3** règles plus faciles à lire, écrire et vérifier.

Actions de règle

Cette nouvelle version modifie les actions de règle. Les nouvelles définitions sont les suivantes :

- **Pass**: arrêt de l'évaluation des règles suivantes par rapport au paquet
- **Alert**: Générer un événement uniquement
- **Block**: abandon du paquet, blocage de la session restante
- **Drop**: abandonner le paquet uniquement
- **Rewrite**: obligatoire si l'option remplace est utilisée
- **React**: page Envoyer une réponse de bloc HTML
- **Reject**: Injecter TCP RST ou ICMP inaccessible

Anatomie des règles

L'anatomie est :

L'en-tête de règle contient l'action, le protocole, le ou les réseaux source et de destination, ainsi que le ou les ports.

Dans **Snort3**, l'en-tête de règle peut être l'une des options suivantes :

- En-tête de règle de service

```
<iline" lang="lua">alert http ( msg:"Alert HTTP rule"; flow:to_client,established;
content:"evil", nocase; sid:1000001; )
```

- En-tête de règle de fichier

```
alert file ( msg: “Alert File example”; file_data; content:”malicious_stuff”; sid:1000006; )
```

- En-tête de règle conventionnelle

```
alert tcp $EXTERNAL_NET any -> $HOME_NET $HTTP_PORTS ( msg:"Alert HTTP rule";
flow:to_client,established; content:"evil", nocase; sid:1000001; )
```

Fonctionnalités des règles

Voici quelques-unes des nouvelles fonctionnalités :

- Espace blanc arbitraire (chaque option sur sa propre ligne)

```
alert tcp $EXTERNAL_NET any -> $HOME_NET $HTTP_PORTS ( msg:"Alert TCP rule";
flow:to_client,established; content:"evil", nocase; sid:1000000; )
```

- l'utilisation cohérente des `ressources` et `;`

```
content:"evil", offset 5, depth 4, nocase;
```

- Les réseaux et les ports sont facultatifs

```
alert http ( Rule body )
```

- Ajoute d'autres tampons rémanents (la liste n'est pas complète)

```
http_uri http_raw_uri http_header http_raw_header http_trailer http_raw_trailer http_cookie
http_raw_cookie http_true_ip http_client_body http_raw_body http_method http_stat_code
http_stat_msg http_version http2_frama_header script_data raw_data
```

- Commentaires de style C

```
alert http ( msg:"Alert HTTP rule"; /* I can write a comment here */ ... )
```

- Mot clé Remark (rem)

```
alert http ( msg:"Alert HTTP rule"; flow:to_client,established; rem:"Put comments in the rule
anywhere"; content:"evil", nocase; sid:1000001; )
```

- Apids, mots clés

```
alert tcp $HOME_NET any -> $EXTERNAL_NET any ( msg:"Alert on apps"; appids:"Google, Google
Drive"; content:"evil", nocase; sid:1000000; )
```

- sd_pattern pour le filtrage des données sensibles
- Mot clé Regex avec utilisation de la technologie hyperflex
- Le mot clé Service remplace les métadonnées

Exemples

Exemple avec en-tête de service http et tampon rémanent http_uri

Tâche : Écrire une règle qui détecte le mot **malicious** dans l'URI HTTP.

Solution :

```
alert http ( msg:"Snort 3 http_uri sticky buffer"; flow:to_server,established; http_uri;
content:"malicious", within 20; sid:1000010; )
```

Exemple avec en-tête de service de fichiers

Tâche : écrire une règle qui détecte les fichiers PDF.

Solution :

```
alert file ( msg:"PDF File Detected"; file_type: "PDF"; sid:1000008; )
```

Liens connexes

Téléchargement des règles Snort et du logiciel IDS

Github

À propos de cette traduction
Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.