FreeBSD SOC Setup: Suricata + Cowrie Honeypot dans un Jail

Table des Matières

1 Introduction

Ce guide détaille la création d'un mini-SOC léger sur FreeBSD 14.3, intégrant :

- **Suricata** (IDS/IPS) pour la détection réseau.
- **Cowrie** (honeypot SSH/Telnet) pour capturer les attaques.
- **Isolation via jails** pour la sécurité.

2 Prérequis

- FreeBSD 14.3 installé.
- Accès root ou utilisateur avec sudo.
- Internet actif pour télécharger les paquets.
- Python ≥ 3.11.

3 Mise à jour du système et installation des dépendances

3.1 Mettre à jour FreeBSD

sudo pkg update

sudo pkg upgrade

3.2 Installer les outils essentiels

sudo pkg install gcc gmake libffi pkgconf rust git python311 py311-venv bash vim htop

4 Installation et configuration de Suricata

4.1 Installer Suricata

sudo pkg install suricata

sudo sysrc suricata_enable="YES"

sudo service suricata start

4.2 Configurer Suricata pour écouter sur wlan0

Éditer /usr/local/etc/suricata/suricata.yaml :

default-rule-path: /var/lib/suricata/rules/rules

rule-files:

- emerging.rules

af-packet:

- interface: wlan0

4.3 Télécharger des règles (ex : Emerging Threats)

sudo mkdir -p /var/lib/suricata/rules/rules

cd /var/lib/suricata/rules/rules

sudo fetch https://rules.emergingthreats.net/open/suricata/emerging.rules.tar.gz

sudo tar -xvzf emerging.rules.tar.gz

sudo rm emerging.rules.tar.gz

sudo service suricata restart

5 Installation de Cowrie (Honeypot SSH/Telnet)

5.1 Cloner le dépôt Cowrie

cd /usr/local

sudo git clone https://github.com/cowrie/cowrie.git

cd cowrie

5.2 Créer un environnement virtuel Python

sudo python3.11 -m venv cowrie-env

sudo chown -R $(whoami) cowrie-env

. cowrie-env/bin/activate

5.3 Installer les dépendances Python

pip install --upgrade pip setuptools wheel

pip install -r requirements.txt

6 Configuration de Cowrie

6.1 Copier et modifier cowrie.cfg

cp etc/cowrie.cfg.dist etc/cowrie.cfg

Éditer etc/cowrie.cfg :

[honeypot]

hostname = svr04

listen_port = 2222

telnet_port = 2223

log_path = var/log/cowrie

download_path = var/lib/cowrie/downloads

6.2 Créer les dossiers de logs

mkdir -p var/log/cowrie var/lib/cowrie/downloads

chmod -R 755 var/log/cowrie var/lib/cowrie

6.3 Démarrer Cowrie

bin/cowrie start

7 Isolation avec Jails (FreeBSD)

7.1 Créer un jail pour Cowrie

sudo pkg install ezjail

sudo ezjail-admin create cowrie-jail 'lo1|127.0.0.2'

sudo ezjail-admin start cowrie-jail

sudo ezjail-admin console cowrie-jail

7.2 Préparer le jail pour Cowrie

Dans le jail, répéter les étapes 3 et 4 pour installer Cowrie.

8 Configuration du Jail pour Cowrie

8.1 Créer la structure du jail

sudo mkdir -p /opt/cowrie-jail/bin

sudo mkdir -p /opt/cowrie-jail/lib

sudo mkdir -p /opt/cowrie-jail/libexec

sudo mkdir -p /opt/cowrie-jail/etc

sudo mkdir -p /opt/cowrie-jail/usr

sudo mkdir -p /opt/cowrie-jail/var

sudo mkdir -p /opt/cowrie-jail/tmp

sudo mkdir -p /opt/cowrie-jail/dev

sudo chmod 1777 /opt/cowrie-jail/tmp

8.2 Copier les binaires et bibliothèques

sudo cp /bin/sh /opt/cowrie-jail/bin/

sudo cp /bin/ls /opt/cowrie-jail/bin/

sudo cp /bin/cat /opt/cowrie-jail/bin/

sudo cp /bin/echo /opt/cowrie-jail/bin/

sudo cp /bin/pwd /opt/cowrie-jail/bin/

sudo cp /bin/date /opt/cowrie-jail/bin/

sudo cp /bin/mkdir /opt/cowrie-jail/bin/

sudo cp /bin/rm /opt/cowrie-jail/bin/

sudo cp /usr/bin/whoami /opt/cowrie-jail/bin/

sudo cp /usr/bin/id /opt/cowrie-jail/bin/

sudo cp /usr/bin/env /opt/cowrie-jail/bin/

8.3 Copier les bibliothèques

sudo cp /lib/libc.so.7 /opt/cowrie-jail/lib/

sudo cp /lib/libutil.so.9 /opt/cowrie-jail/lib/

sudo cp /lib/libedit.so.8 /opt/cowrie-jail/lib/

sudo cp /lib/libtinfow.so.9 /opt/cowrie-jail/lib/

sudo cp /libexec/ld-elf.so.1 /opt/cowrie-jail/libexec/

8.4 Créer les périphériques dans /dev

sudo mknod /opt/cowrie-jail/dev/null c 0 0

sudo mknod /opt/cowrie-jail/dev/zero c 0 12

sudo mknod /opt/cowrie-jail/dev/tty c 0 3

sudo chmod 666 /opt/cowrie-jail/dev/null

sudo chmod 666 /opt/cowrie-jail/dev/zero

sudo chmod 666 /opt/cowrie-jail/dev/tty

8.5 Ajouter les fichiers de configuration

sudo sh -c 'echo "root:x:0:0:root:/root:/bin/sh" > /opt/cowrie-jail/etc/passwd'

sudo sh -c 'echo "root:*:0:0:root:/root:/bin/sh" > /opt/cowrie-jail/etc/master.passwd'

sudo sh -c 'echo "wheel:*:0:root" > /opt/cowrie-jail/etc/group'

8.6 Copier Cowrie dans le jail

sudo cp -r /usr/local/cowrie /opt/cowrie-jail/usr/

8.7 Tester le jail

sudo chroot /opt/cowrie-jail /bin/sh

9 Test et Validation

9.1 Vérifier que Cowrie écoute sur le port 2222

sockstat -4 -l | grep 2222

9.2 Tester une connexion SSH

ssh root@<IP_DU_SERVEUR> -p 2222

9.3 Vérifier les logs

tail -f /usr/local/cowrie/var/log/cowrie/cowrie.json

10 Arborescence finale

/usr/local/

├── cowrie/

│ ├── bin/

│ ├── cowrie-env/

│ ├── var/

│ │ ├── log/cowrie/

│ │ └── lib/cowrie/

│ └── etc/cowrie.cfg

└── suricata/

11 Notes importantes

- **Suricata :** Écoute sur wlan0 et utilise les règles Emerging Threats.
- **Cowrie** : Écoute sur les ports 2222 (SSH) et 2223 (Telnet).
- **Logs :** Centralisés dans /var/log/suricata/ et /usr/local/cowrie/var/log/cowrie/.
- **Sécurité** : Pour un lab pro, utilise une jail ou une VM dédiée.

12 Conclusion :

J'ai passé 8h à configurer ce lab, je l'ai trouvé très intéressant mais ce n'était pas une mince affaire, je vous ai donc indiqué les commandes qui ont permis au lab de fonctionner, mais si quelque chose d'inattendu se passe, je vous invite à regarder la documentation ou à demander de l'aide à un LLM, cela m'a beaucoup aidé !

Afin de ne pas mettre tous ses oeufs dans le même panier, car c'est très important en sécurité si une machine se fait compromettre, ou si elle deviens hors service, la partie SIEM (Splunk) et samba seront assurés dans une autre documentation via une autre machine.

Merci pour votre lecture !