Despliegue y configuración automática de R-Snort Collector

Deian Orlando Petrovics

Abril 2025

1 Introducción

Este documento describe de forma profesional, detallada y precisa todos los pasos llevados a cabo para desplegar el sistema de captura automática de alertas de Snort 3 y métricas del sistema en una Raspberry Pi 5, utilizando como bases MariaDB y un script Python permanente gestionado mediante `systemd`.

2 Preparación del entorno

- Instalación de MariaDB:

```
sudo apt update
sudo apt install mariadb-server mariadb-client
```

- Creación del usuario y base de datos:

```
sudo mysql
CREATE DATABASE rsnort_webapp;
CREATE USER 'snortuser'@'localhost' IDENTIFIED BY 'contraseña_segura';
GRANT ALL PRIVILEGES ON rsnort_webapp.* TO 'snortuser'@'localhost';
FLUSH PRIVILEGES;
EXIT;
```

- Creación de tablas:

```
USE rsnort_webapp;
CREATE TABLE alerts (...);
CREATE TABLE system_metrics (...);
CREATE TABLE users (...);
CREATE TABLE rules (...);
CREATE TABLE service_actions (...);
CREATE TABLE email_verifications (...);
```

(Se crearon las tablas necesarias para la WebApp de administración.)

3 Desarrollo del script R-Snort Collector

- Se desarrolló un script `r-snort-collector.py` en Python 3, con las siguientes características:
  - Captura continua de métricas del sistema (CPU, RAM, temperatura, disco).
  - Lectura de `/opt/snort/logs/live/alert`$json.txtparacapturaralertas.Inserciónautomáticadedatosen Maria$
  - Manejo de errores y reconexión automática.
- Se instalaron dependencias necesarias:

```
sudo apt install python3-pip
pip3 install psutil pymysql
```

- Se creó el directorio final de trabajo:

```
sudo mkdir -p /opt/r-snort_webapp
sudo cp -r /home/snort-lab/r-snort_webapp/* /opt/r-snort_webapp/
sudo chown -R snort:snort /opt/r-snort_webapp/
```

- Se configuró el script para leer variables de entorno (contraseña segura) desde un archivo:

```
sudo nano /etc/r-snort_collector.env
RSNORT_DB_PASSWORD=contraseña_segura
sudo chmod 600 /etc/r-snort_collector.env
```

4 Creación del servicio systemd

- Se creó el servicio `r-snort_collector.service` :

```
sudo nano /etc/systemd/system/r-snort_collector.service
```

Contenido del servicio:

```
[Unit]
Description=r-snort_collector: Captura de alertas y métricas en MariaDB
After=network.target mariadb.service

[Service]
EnvironmentFile=/etc/r-snort_collector.env
Environment="PYTHONUNBUFFERED=1"
WorkingDirectory=/opt/r-snort_webapp
ExecStart=/usr/bin/python3 /opt/r-snort_webapp/r-snort_collector.py
Restart=always
User=snort
Group=snort

[Install]
WantedBy=multi-user.target
```

Se activó y lanzó el servicio:

```
sudo systemctl daemon-reload
sudo systemctl enable r-snort_collector
sudo systemctl start r-snort_collector
```

5 Pruebas y verificación

- Confirmación de inserción continua de alertas y métricas:

```
mysql -u snortuser -p rsnort_webapp
SELECT COUNT(*) FROM alerts;
SELECT COUNT(*) FROM system_metrics;
```

- Validación de estructura y contenido de alertas:

```
SELECT * FROM alerts ORDER BY timestamp DESC LIMIT 5;
```

- Supervisión del servicio:

```
sudo systemctl status r-snort_collector
journalctl -u r-snort_collector -f
```

6 Notas importantes

- Se aseguró que Snort estuviera siempre activo capturando eventos.
- Se comprobó que la interfaz de red estuviera activa tras cualquier reboot.
- El diseño permite que todo el sistema sea paquetizado en un futuro `.deb` para instalación automática.
- La seguridad de la contraseña de base de datos está mantenida via archivo de entorno con permisos restringidos.