Manuel Technique : Déploiement d'une Solution SOC avec OSSEC et Zabbix

1. Description de la Solution et Rôle dans un SOC

Ce document détaille la mise en place d'une solution de supervision de sécurité (SOC) basée sur l'intégration de Zabbix et OSSEC. L'objectif est de combiner la supervision des performances et de la disponibilité des systèmes avec la détection d'intrusions et l'analyse de logs de sécurité.

Zabbix :

Agit comme une plateforme de supervision centralisée. Il collecte des métriques sur l'état de santé des serveurs (CPU, RAM, réseau, espace disque) et offre une interface de visualisation (tableaux de bord, graphiques) pour suivre les performances en temps réel. Dans ce contexte, il sert également à visualiser les alertes de sécurité remontées par OSSEC.

OSSEC :

Fonctionne comme un Système de Détection d'Intrusion basé sur l'Hôte (HIDS). Il analyse en temps réel les journaux système, vérifie l'intégrité des fichiers, détecte les rootkits et peut déclencher des réponses actives (comme le blocage d'une adresse IP) lors de la détection d'une activité malveillante.

➔ L'intégration de ces deux outils permet de créer une chaîne de sécurité complète : OSSEC détecte les menaces au niveau de l'hôte, et Zabbix centralise la visualisation de ces alertes aux côtés des métriques de performance, offrant aux analystes SOC une vue corrélée des événements de sécurité et de leur impact sur l'infrastructure.

2. Environnement et Prérequis

La solution est déployée sur une machine virtuelle unique qui héberge tous les composants.

2.1 Configuration de la Machine Virtuelle

Caractéristique	Spécification Minimale	Spécification Recommandée
Système	Ubuntu Server 22.04 LTS	Ubuntu Server 22.04 LTS
RAM	4 GB	8 GB
CPU	2 cœurs	4 cœurs
Stockage	50 GB	50 GB
Adresse IP	192.168.56.110 (statique)	(Adapter selon le réseau)

3. Étapes d'Installation et de Configuration

3.1 Préparation de l'Environnement Serveur

Étape 3.1.1 : Mise à jour et Installation des Outils de Base

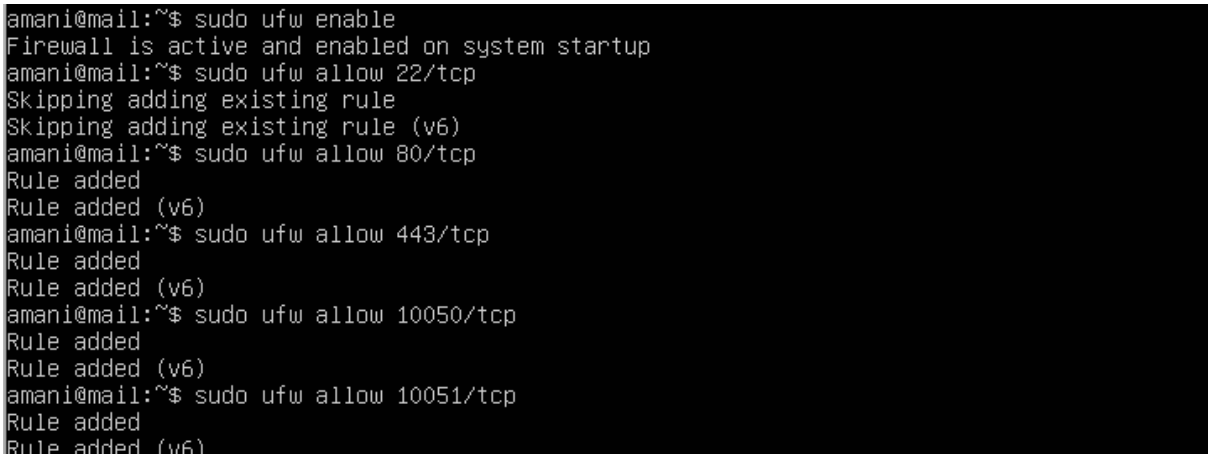
La première étape consiste à mettre à jour le système et à installer les paquets essentiels.

```
# Mise à jour du système
sudo apt update && sudo apt upgrade -y

# Installation des outils essentiels
sudo apt install -y curl wget git vim python3 python3-pip software-properties-common apt-transport-https

# Configuration du fuseau horaire
sudo timedatectl set-timezone Europe/Paris
```

Étape 3.1.2 : Configuration du Pare-feu

Le pare-feu UFW (Uncomplicated Firewall) est activé pour sécuriser le serveur. Les ports nécessaires au fonctionnement de SSH, des serveurs web et de Zabbix sont ouverts.

```
# Activation du pare-feu
sudo ufw enable

# Ouverture des ports
sudo ufw allow 22/tcp   # SSH
sudo ufw allow 80/tcp   # HTTP
sudo ufw allow 443/tcp  # HTTPS
sudo ufw allow 10050/tcp # Zabbix Agent
sudo ufw allow 10051/tcp # Zabbix Server
```



Capture d'écran de la configuration du pare-feu : L'image du terminal montre l'exécution successive des commandes `sudo ufw enable` et `sudo ufw allow` pour chaque port, confirmant l'ajout des règles pour les protocoles TCP sur les ports 22, 80, 443, 10050 et 10051.

3.2 Installation de Zabbix 7.0

Étape 3.2.1 : Ajout du Dépôt Zabbix

Pour installer Zabbix, il est nécessaire de télécharger et d'installer le paquet du dépôt officiel.

Télécharger le package de dépôt Zabbix 7.0
wget
https://repo.zabbix.com/zabbix/7.0/ubuntu/pool/main/z/zabbix-release/zabbix-release_latest_7.0+ubuntu22.04_all.deb

```
root@amani:/home/amani# wget https://repo.zabbix.com/zabbix/7.0/ubuntu/pool/main/z/za
bbix-release/zabbix-release_latest_7.0+ubuntu22.04_all.deb
--2025-05-10 22:30:42--  https://repo.zabbix.com/zabbix/7.0/ubuntu/pool/main/z/zabbix
-release/zabbix-release_latest_7.0+ubuntu22.04_all.deb
Resolving repo.zabbix.com (repo.zabbix.com)... 178.128.6.101, 2604:a880:2:d0::2062:d0
01
Connecting to repo.zabbix.com (repo.zabbix.com)|178.128.6.101|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 8288 (8,1K) [application/octet-stream]
Saving to: 'zabbix-release_latest_7.0+ubuntu22.04_all.deb'

zabbix-release_latest 100%[====================>]   8,09K  --.-KB/s    in 0s

2025-05-10 22:30:46 (101 MB/s) - 'zabbix-release_latest_7.0+ubuntu22.04_all.deb' save
d [8288/8288]
```

# Installer le dépôt
sudo dpkg -i zabbix-release_latest_7.0+ubuntu22.04_all.deb

```
root@amani:/home/amani# dpkg -i zabbix-release_latest_7.0+ubuntu22.04_all.deb
Sélection du paquet zabbix-release précédemment désélectionné.
(Lecture de la base de données... 74805 fichiers et répertoires déjà installés.)
Préparation du dépaquetage de zabbix-release_latest_7.0+ubuntu22.04_all.deb ...
Dépaquetage de zabbix-release (1:7.0-2+ubuntu22.04) ...
Paramétrage de zabbix-release (1:7.0-2+ubuntu22.04) ...
```

# Mettre à jour la liste des paquets
sudo apt update

```
root@amani:/home/amani# apt update
Réception de :1 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Atteint :2 http://tn.archive.ubuntu.com/ubuntu jammy InRelease
Réception de :3 http://tn.archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
Réception de :4 https://repo.zabbix.com/zabbix-tools/debian-ubuntu jammy InRelease [2
```

```
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
54 paquets peuvent être mis à jour. Exécutez « apt list --upgradable » pour les voir
```

Captures d'écran du processus : Les captures du terminal illustrent le téléchargement réussi du fichier `.deb` avec `wget`, son installation avec `dpkg -i`, et la mise à jour des sources `apt` qui inclut désormais le nouveau dépôt Zabbix.

Étape 3.2.2 : Installation du Serveur de Base de Données MySQL

Zabbix nécessite une base de données pour stocker les métriques et sa configuration. MySQL est utilisé dans ce déploiement.

```
# Installation de MySQL Server et Client
sudo apt install -y mysql-server mysql-client
```

```
root@amani:/home/amani# apt install mysql-server
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les paquets supplémentaires suivants seront installés :
  libcgi-fast-perl libcgi-pm-perl libclone-perl libencode-locale-perl libfcgi-bin
  libfcgi-perl libfcgi0ldbl libhtml-parser-perl libhtml-tagset-perl
```

```
# Lancement du script de sécurisation
sudo mysql_secure_installation
```

Lors de l'exécution de `mysql_secure_installation`, les choix suivants ont été faits pour la démonstration :

- Validation du composant de mot de passe: **Oui**
- Niveau de politique de mot de passe : **0 (LOW)**
- Changer le mot de passe root : **Non** (mot de passe vide conservé)
- Supprimer les utilisateurs anonymes : **Oui**
- Interdire la connexion root à distance : **Oui**
- Supprimer la base de données de test : **Oui**
- Recharger les tables de privilèges : **Oui**

Étape 3.2.3 : Création de la Base de Données pour Zabbix

Une base de données et un utilisateur dédiés sont créés pour Zabbix.

```
# Connexion à MySQL
sudo mysql -u root -p

# Commandes SQL à exécuter
CREATE DATABASE zabbix CHARACTER SET utf8mb4 COLLATE utf8mb4_bin;
CREATE USER 'zabbix'@'localhost' IDENTIFIED BY 'Zabbix25272000***';
GRANT ALL PRIVILEGES ON zabbix.* TO 'zabbix'@'localhost';
SET GLOBAL LOG_BIN_TRUST_FUNCTION_CREATORS = 1;
FLUSH PRIVILEGES;
quit;
```

```
root@amani:/home/amani# mysql -uroot -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.42-0ubuntu0.22.04.1 (Ubuntu)

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

```
mysql> create database zabbix character set utf8mb4 collate utf8mb4_bin;
Query OK, 1 row affected (0,07 sec)

mysql> create user zabbix@localhost identified by '25272000DhA***';
Query OK, 0 rows affected (0,06 sec)

mysql> grant all privileges on zabbix.* to zabbix@localhost;
Query OK, 0 rows affected (0,03 sec)

mysql> set global log_bin_trust_function_creators = 1;
Query OK, 0 rows affected, 1 warning (0,00 sec)

mysql> quit
Bye
```

Capture d'écran des commandes SQL : Le terminal MySQL affiche l'exécution réussie de chaque commande : `CREATE DATABASE`, `CREATE USER`, `GRANT`, `SET GLOBAL`, et `quit`.

Étape 3.2.4 : Installation des Composants Zabbix

Installation du serveur Zabbix, de l'interface web (frontend) et de l'agent local.

# Installation des paquets Zabbix
sudo apt install -y zabbix-server-mysql zabbix-frontend-php zabbix-apache-conf zabbix-sql-scripts zabbix-agent

```
root@amani:/home/amani# apt install zabbix-server-mysql zabbix-frontend-php zabbix-ap
ache-conf zabbix-sql-scripts zabbix-agent
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les paquets supplémentaires suivants seront installés :
  apache2 apache2-bin apache2-data apache2-utils bzip2 fontconfig-config
  fonts-dejavu fonts-dejavu-core fonts-dejavu-extra fping libapache2-mod-php
  libapache2-mod-php8.1 libaprl libaprutill libaprutill-dbd-sqlite3
  libaprutill-ldap libdeflate0 libevent-extra-2.1-7 libevent-pthreads-2.1-7
  libfontconfig1 libgd3 libjbig0 libjpeg-turbo8 libjpeg8 libltdl7 liblua5.3-0
  libmodbus5 libmysqlclient21 libodbc2 libonig5 libopenipmi0 libsensors-config
  libsensors5 libsnmp-base libsnmp40 libtiff5 libwebp7 libxpm4 mailcap mime-support
  mysql-client mysql-client-8.0 mysql-client-core-8.0 mysql-common php-bcmath
```

Import du schéma initial de la base de données
zcat /usr/share/zabbix-sql-scripts/mysql/server.sql.gz | mysql -uzabbix -p zabbix

```
root@amani:/home/amani# zcat /usr/share/zabbix-sql-scripts/mysql/server.sql.gz | mysq
l --default-character-set=utf8mb4 -uzabbix -p zabbix
Enter password:
```

# Désactivation de la contrainte sur les fonctions après l'import
sudo mysql -u root -e "SET GLOBAL LOG_BIN_TRUST_FUNCTION_CREATORS = 0;"

```
root@amani:/home/amani# mysql -uroot -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 16
Server version: 8.0.42-0ubuntu0.22.04.1 (Ubuntu)

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> set global log_bin_trust_function_creators = 0;
Query OK, 0 rows affected, 1 warning (0,00 sec)

mysql> quit
Bye
```

Étape 3.2.5 : Configuration et Démarrage des Services Zabbix

Le fichier de configuration du serveur Zabbix doit être édité pour inclure le mot de passe de la base de données.

1. **Éditer le fichier `/etc/zabbix/zabbix_server.conf`** :

```
root@amani:/home/amani# nano /etc/zabbix/zabbix_server.conf
```

```
  GNU nano 6.2                 /etc/zabbix/zabbix_server.conf
# DBUser=

DBUser=zabbix

### Option: DBPassword
#       Database password.
#       Comment this line if no password is used.
#
# Mandatory: no
# Default:
DBPassword=25272000DhA***
```

Capture d'écran de la configuration : L'éditeur de texte `nano` montre le fichier `zabbix_server.conf` avec la ligne `DBPassword` correctement configurée.

2. **Redémarrer et activer les services** :

```
root@amani:/home/amani# systemctl restart zabbix-server zabbix-agent apache2
root@amani:/home/amani# systemctl enable zabbix-server zabbix-agent apache2
Synchronizing state of zabbix-server.service with SysV service script with /lib/systemd/system
d-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable zabbix-server
Synchronizing state of zabbix-agent.service with SysV service script with /lib/systemd/systemd
-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable zabbix-agent
Synchronizing state of apache2.service with SysV service script with /lib/systemd/systemd-sysv
-install.
Executing: /lib/systemd/systemd-sysv-install enable apache2
Created symlink /etc/systemd/system/multi-user.target.wants/zabbix-server.service → /lib/syste
md/system/zabbix-server.service.
```

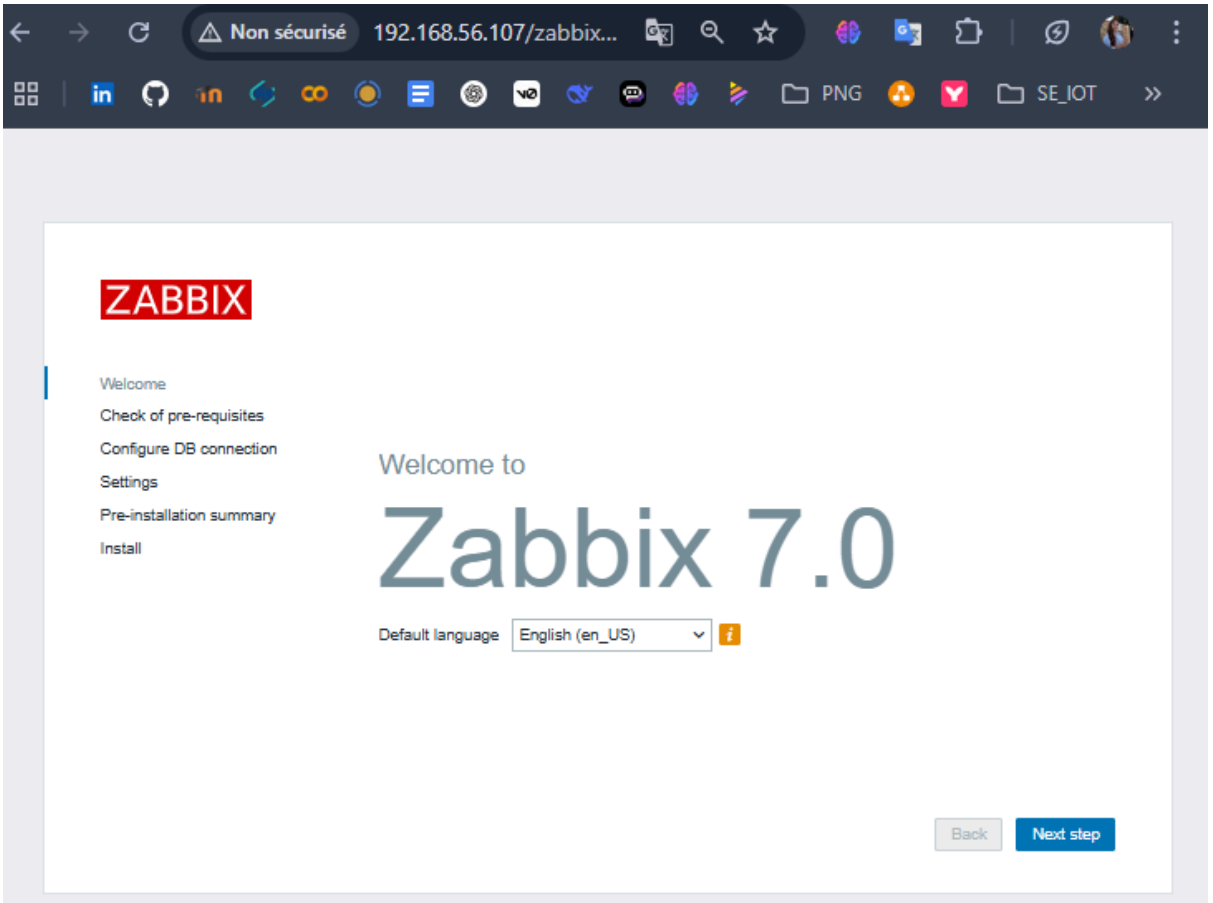
Étape 3.2.6 : Configuration via l'Interface Web

L'installation se finalise via l'interface web de Zabbix.

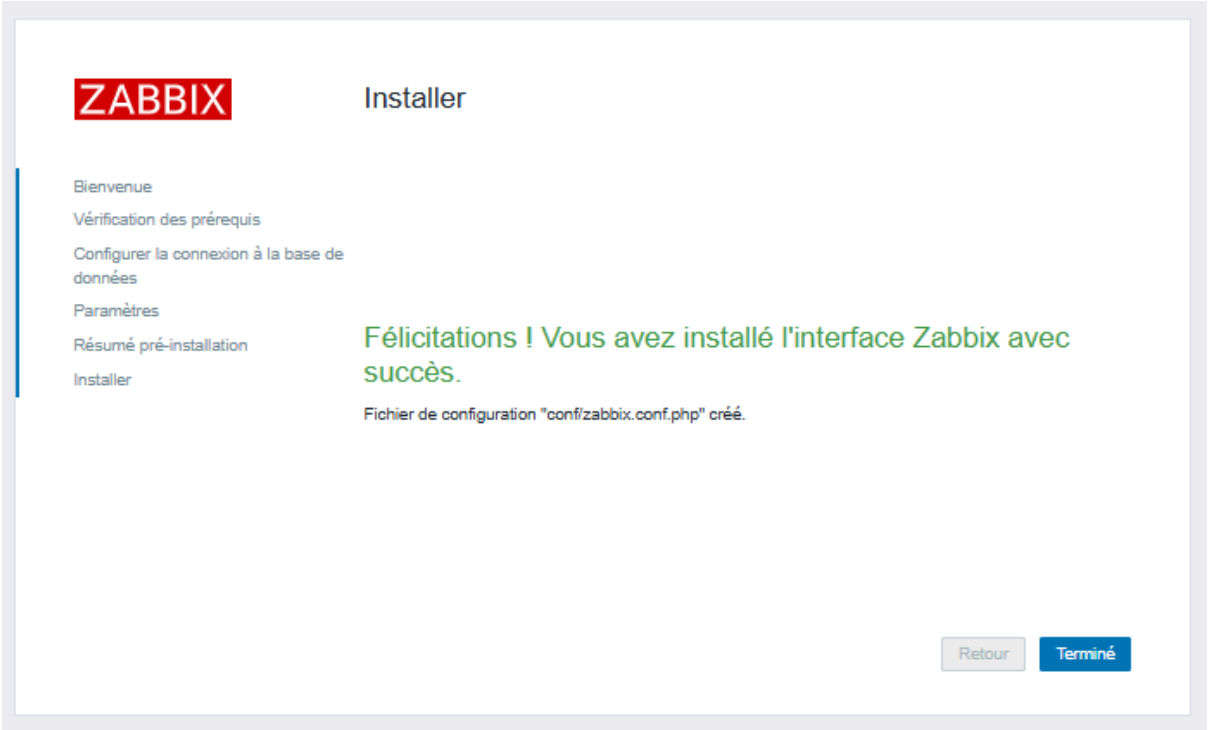
1. Accéder à l'URL `http://192.168.56.110/zabbix` (ou l'IP du serveur) dans un navigateur.
2. Suivre les étapes de l'assistant d'installation, en fournissant les informations de connexion à la base de données configurées précédemment.
3. L'assistant vérifie les prérequis, configure la connexion et finalise l'installation.

Captures d'écran de l'interface web :

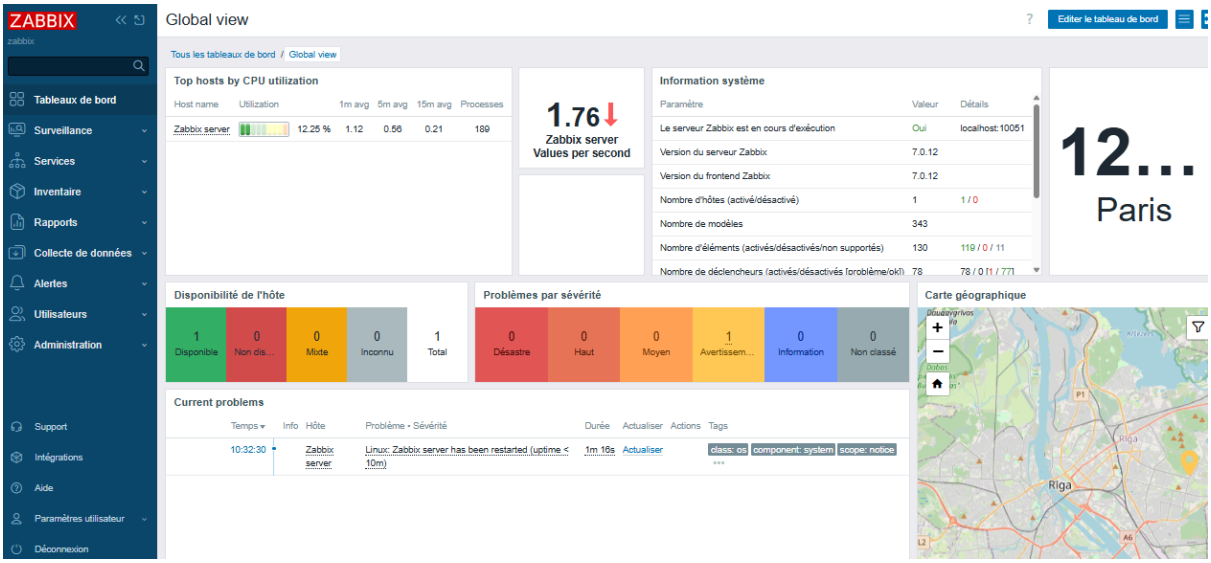
- La première capture montre l'écran d'accueil de l'assistant d'installation "Welcome to Zabbix 7.0".



- La seconde capture affiche le message de succès : "Félicitations ! Vous avez installé l'interface Zabbix avec succès."



- Une fois connecté, le tableau de bord global de Zabbix est visible, affichant des informations système, l'état des hôtes et les problèmes actuels.



3.3 Installation d'OSSEC 3.7.0

Étape 3.3.1 : Installation des Dépendances

OSSEC doit être compilé depuis les sources, ce qui requiert des outils de développement.

```
# Mettre à jour la liste des paquets
sudo apt update

# Installer les dépendances de compilation et outils de développement
sudo apt install -y build-essential make libssl-dev libpcre2-dev zlib1g-dev gcc g++
libevent-dev
```

```
amani@mail:~$ sudo apt install -y build-essential make libssl-dev libpcre2-dev zlib1g-dev
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
build-essential est déjà la version la plus récente (12.9ubuntu3).
build-essential passé en « installé manuellement ».
make est déjà la version la plus récente (4.3-4.1build1).
make passé en « installé manuellement ».
libpcre2-dev est déjà la version la plus récente (10.39-3ubuntu0.1).
libssl-dev est déjà la version la plus récente (3.0.2-0ubuntu1.20).
zlib1g-dev est déjà la version la plus récente (1:1.2.11.dfsg-2ubuntu9.2).
zlib1g-dev passé en « installé manuellement ».
0 mis à jour, 0 nouvellement installés, 0 à enlever et 142 non mis à jour.
```

```
amani@mail:~$ sudo apt install -y gcc g++ libevent-dev
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
g++ est déjà la version la plus récente (4:11.2.0-1ubuntu1).
g++ passé en « installé manuellement ».
gcc est déjà la version la plus récente (4:11.2.0-1ubuntu1).
gcc passé en « installé manuellement ».
Les NOUVEAUX paquets suivants seront installés :
  libevent-2.1-7 libevent-dev libevent-openssl-2.1-7
0 mis à jour, 3 nouvellement installés, 0 à enlever et 142 non mis à jour.
Il est nécessaire de prendre 442 ko dans les archives.
```

Étape 3.3.2 : Téléchargement et Compilation

Téléchargement des sources d'OSSEC et extraction de l'archive.

```
# Se déplacer dans le répertoire temporaire
cd /tmp

# Télécharger l'archive OSSEC 3.7.0
wget -O ossec-hids-3.7.0.tar.gz
https://github.com/ossec/ossec-hids/archive/refs/tags/3.7.0.tar.gz
```

```
amani@mail:/tmp$ wget -O ossec-hids-3.7.0.tar.gz https://github.com/ossec/ossec-hids/archive/r
efs/tags/3.7.0.tar.gz
--2026-01-10 10:50:05--  https://github.com/ossec/ossec-hids/archive/refs/tags/3.7.0.tar.gz
Resolving github.com (github.com)... 140.82.121.3
Connecting to github.com (github.com)|140.82.121.3|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://codeload.github.com/ossec/ossec-hids/tar.gz/refs/tags/3.7.0 [following]
--2026-01-10 10:50:06--  https://codeload.github.com/ossec/ossec-hids/tar.gz/refs/tags/3.7.0
Resolving codeload.github.com (codeload.github.com)... 140.82.121.10
Connecting to codeload.github.com (codeload.github.com)|140.82.121.10|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [application/x-gzip]
Saving to: 'ossec-hids-3.7.0.tar.gz'

ossec-hids-3.7.0.tar.gz     [<=>                                ]   2,40M   360KB/s    in 18s

2026-01-10 10:50:26 (133 KB/s) - 'ossec-hids-3.7.0.tar.gz' saved [2518737]
```

Extraire l'archive
tar -xvzf ossec-hids-3.7.0.tar.gz

```
amani@mail:/tmp$ tar -xvzf ossec-hids-3.7.0.tar.gz
ossec-hids-3.7.0/
ossec-hids-3.7.0/.gitignore
ossec-hids-3.7.0/.travis.yml
ossec-hids-3.7.0/BUGS
ossec-hids-3.7.0/CHANGELOG.md
ossec-hids-3.7.0/CONFIG
ossec-hids-3.7.0/CONTRIBUTORS
ossec-hids-3.7.0/Dockerfile
ossec-hids-3.7.0/INSTALL
```

Se déplacer dans le répertoire des sources
cd ossec-hids-3.7.0

Étape 3.3.3 : Installation Interactive

Lancement du script d'installation interactif.

sudo ./install.sh

Les choix suivants sont effectués pendant l'installation :

- Type d'installation : `local` (serveur et agent sur la même machine).
- Répertoire d'installation : `/var/ossec` (par défaut).
- Notification par e-mail : `n` (non).
- Démon de vérification d'intégrité : `y` (oui).
- Moteur de détection de rootkit : `y` (oui).
- Réponse active : `y` (oui).
- Réseaux autorisés : `192.168.0.0/16` (par défaut).
- Réponse active `firewall-drop` : `y` (oui).
- Ajout d'IP à la liste blanche : `n` (non).
- Syslog distant (port 514 UDP) : `n` (non).

Étape 3.3.4 : Démarrage et Vérification

Une fois l'installation terminée, les services OSSEC sont démarrés.

# Démarrer OSSEC
sudo /var/ossec/bin/ossec-control start

```
amani@mail:/tmp/ossec-hids-3.7.0$ sudo /var/ossec/bin/ossec-control start
Starting OSSEC HIDS v3.7.0...
2026/01/10 10:56:57 ossec-maild: INFO: E-Mail notification disabled. Clean Exit.
Started ossec-maild...
Started ossec-execd...
Started ossec-analysisd...
Started ossec-logcollector...
Started ossec-syscheckd...
Started ossec-monitord...
Completed.
```

# Vérifier le statut des services
sudo /var/ossec/bin/ossec-control status

```
amani@mail:/tmp/ossec-hids-3.7.0$ sudo /var/ossec/bin/ossec-control status
ossec-monitord is running...
ossec-logcollector is running...
ossec-syscheckd is running...
ossec-analysisd is running...
ossec-maild not running...
ossec-execd is running...
```

# Consulter les logs en temps réel
sudo tail -f /var/ossec/logs/ossec.log

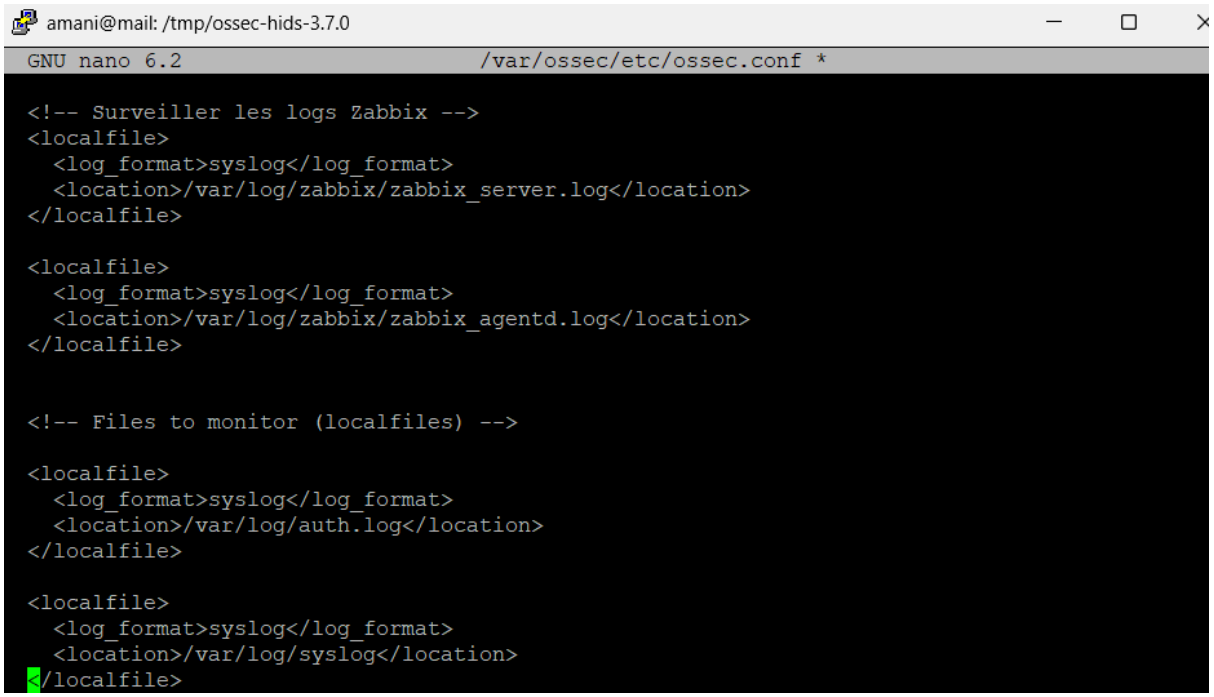
```
amani@mail:/tmp/ossec-hids-3.7.0$ sudo tail -f /var/ossec/logs/ossec.log
2026/01/10 10:57:04 ossec-logcollector(1950): INFO: Analyzing file: '/var/log/syslog'.
2026/01/10 10:57:04 ossec-logcollector(1950): INFO: Analyzing file: '/var/log/dpkg.log'.
2026/01/10 10:57:04 ossec-logcollector(1950): INFO: Analyzing file: '/var/log/apache2/error.log'.
2026/01/10 10:57:04 ossec-logcollector(1950): INFO: Analyzing file: '/var/log/apache2/access.log'.
2026/01/10 10:57:04 ossec-logcollector: INFO: Monitoring output of command(360): df -P
2026/01/10 10:57:04 ossec-logcollector: INFO: Monitoring full output of command(360): netstat -tan
|grep LISTEN |egrep -v '(127.0.0.1| ::1)' | sort
2026/01/10 10:57:04 ossec-logcollector: INFO: Monitoring full output of command(360): last -n 5
2026/01/10 10:57:04 ossec-logcollector: INFO: Started (pid: 22142).
2026/01/10 10:58:07 ossec-syscheckd: INFO: Starting syscheck scan (forwarding database).
2026/01/10 10:58:07 ossec-syscheckd: INFO: Starting syscheck database (pre-scan).
```

Capture d'écran de l'état des services : Le terminal affiche le statut "running" pour `ossec-monitor`, `ossec-logcollector`, `ossec-syscheckd`, `ossec-analysisd` et `ossec-execd`. Le service `ossec-maild` est affiché comme "not running", ce qui est normal car les notifications par email ont été désactivées.

Étape 3.3.5 : Configuration Supplémentaire

Pour intégrer OSSEC et Zabbix, OSSEC doit être configuré pour surveiller les fichiers de logs de Zabbix ainsi que les logs système critiques.

1. **Éditer le fichier `/var/ossec/etc/ossec.conf`** et ajouter les blocs `<localfile>` suivants :



2. **Redémarrer OSSEC pour appliquer les changements** :

   sudo /var/ossec/bin/ossec-control restart

```
amani@mail:/tmp/ossec-hids-3.7.0$ sudo /var/ossec/bin/ossec-control restart
Killing ossec-monitord ..
Killing ossec-logcollector ..
Killing ossec-syscheckd ..
Killing ossec-analysisd ..
ossec-maild not running ..
Killing ossec-execd ..
OSSEC HIDS v3.7.0 Stopped
Starting OSSEC HIDS v3.7.0...
2026/01/10 11:02:57 ossec-maild: INFO: E-Mail notification disabled. Clean Exit.
Started ossec-maild...
Started ossec-execd...
Started ossec-analysisd...
Started ossec-logcollector...
Started ossec-syscheckd...
Started ossec-monitord...
Completed.
amani@mail:/tmp/ossec-hids-3.7.0$ sudo /var/ossec/bin/ossec-control status
ossec-monitord is running...
ossec-logcollector is running...
ossec-syscheckd is running...
ossec-analysisd is running...
ossec-maild not running...
ossec-execd is running...
```

4. Cas d'Usage et Scénarios de Test

4.1 Intégration par Réponse Active

Pour que les alertes Zabbix déclenchent une action dans OSSEC, une règle personnalisée et une configuration de réponse active sont mises en place.

1. **Créer une règle OSSEC pour les alertes Zabbix** : Éditer `/var/ossec/rules/local_rules.xml` et ajouter :

```
 GNU nano 6.2              /var/ossec/rules/local_rules.xml *

<group name="local,zabbix_alerts,">
  <rule id="100002" level="10">
    <if_sid>500</if_sid>
    <match>Zabbix alert</match>
    <description>Alerte de sécurité critique venant de Zabbix</description>
  </rule>
</group>
```

2. **Lier l'alerte à une réponse active** : Éditer `/var/ossec/etc/ossec.conf` et ajouter un bloc `<active-response>` qui déclenche la commande `firewall-drop` lorsque la règle `100002` est activée.

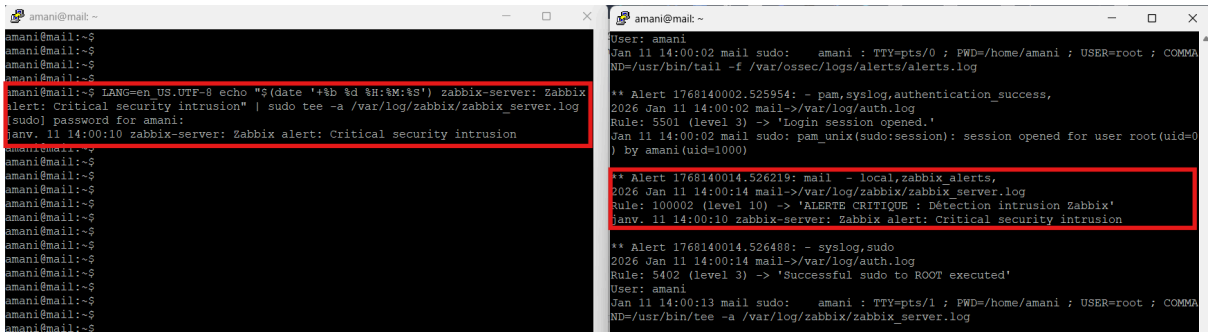
```
 GNU nano 6.2                  /var/ossec/etc/ossec.conf *

  <active-response>
    <!-- Firewall Drop response. Block the IP for
       - 600 seconds on the firewall (iptables,
       - ipfilter, etc).
      -->
    <command>firewall-drop</command>
    <location>local</location>
    <level>6</level>
    <rules_id>100001</rules_id>
    <timeout>600</timeout>
  </active-response>
```

4.2 Scénario 1 : Preuve de Concept (Injection de Log)

Ce test vise à vérifier que la chaîne de détection fonctionne en simulant une alerte critique.

1. **Injecter un log de sécurité critique** dans le fichier surveillé par OSSEC :

2. **Vérifier la détection par OSSEC** en surveillant le fichier d'alertes :



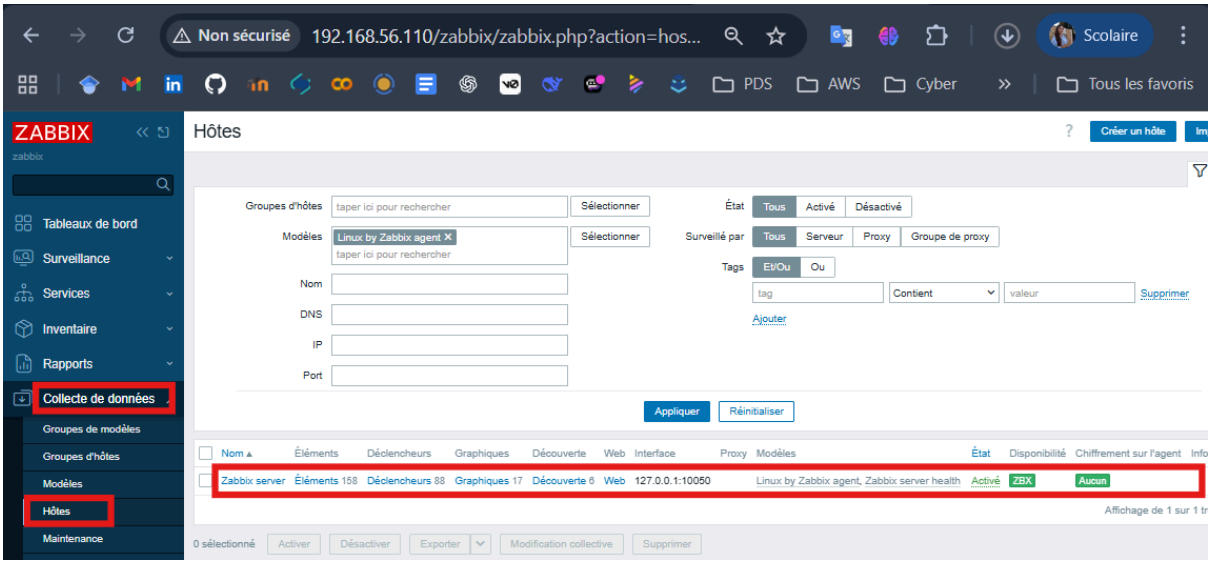
Captures d'écran de l'alerte : Les captures montrent deux fenêtres de terminal. À gauche, la commande d'injection de log est exécutée. À droite, le fichier `alerts.log` affiche immédiatement une nouvelle alerte. L'analyse de l'alerte montre :

- **Règle déclenchée** : `Rule: 100002 (level 10)`
- **Description** : `ALERTE CRITIQUE : Détection intrusion Zabbix`
- **Log original** : `Jan 11 14:37:24 zabbix-server: Zabbix alert: Critical intrusion.`

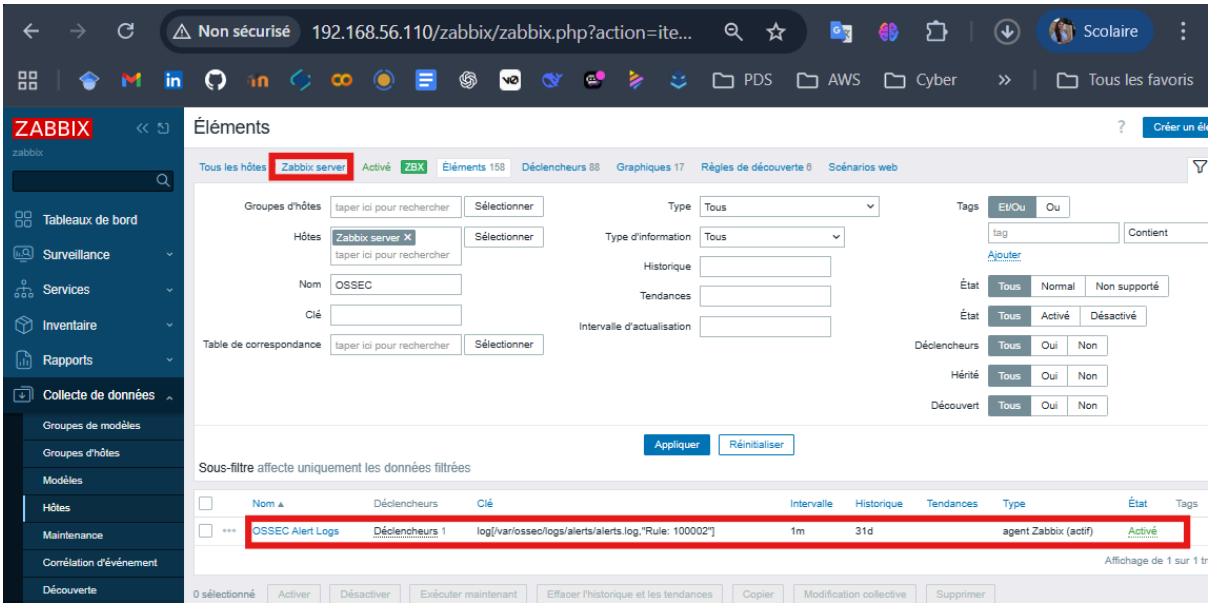
4.3 Visualisation des Alertes dans Zabbix

Pour remonter les alertes OSSEC dans l'interface Zabbix, un élément (`item`) et un déclencheur (`trigger`) sont créés.

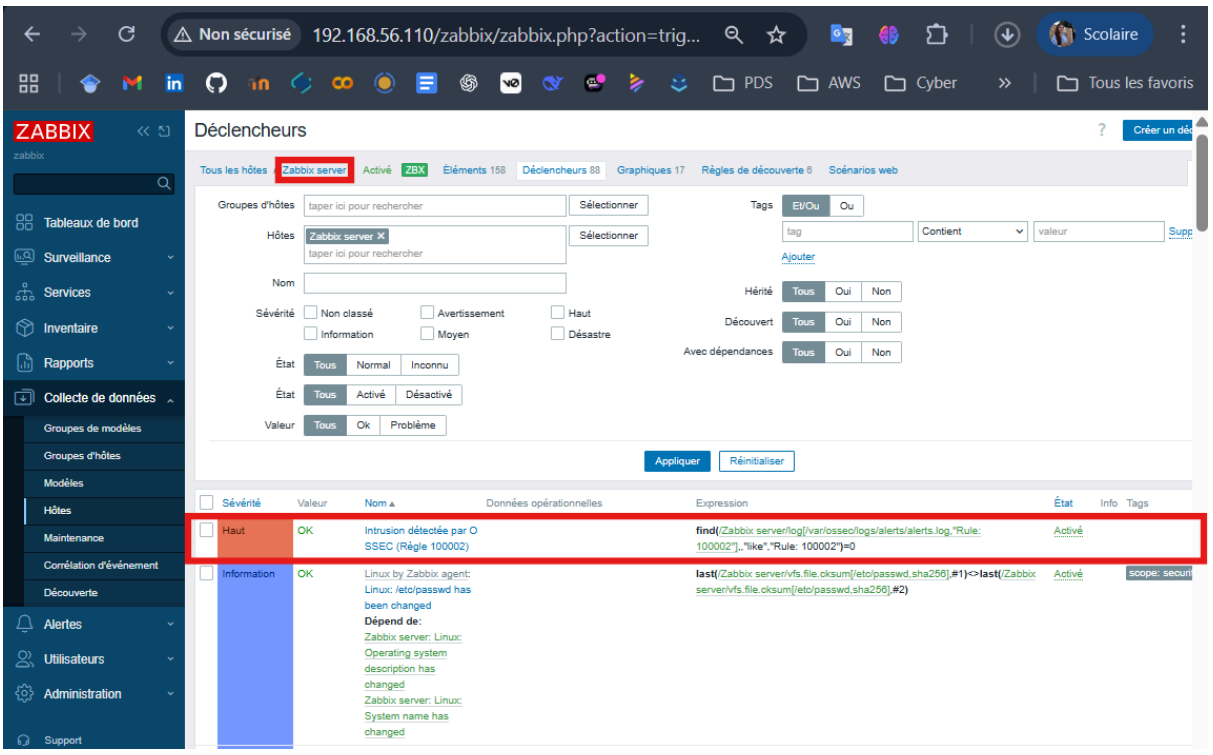
1. **Création de l'élément** :
    - Naviguer vers `Collecte de données` -> `Hôtes`.
    - Sélectionner l'hôte `Zabbix server` et aller dans l'onglet `Éléments`.



- Créer un nouvel élément nommé `OSSEC Alert Logs`.
- Clé : `log[/var/ossec/logs/alerts/alerts.log,"Rule: 100002"]`
- Type d'information : `Log`
- Intervalle d'actualisation : `1m`

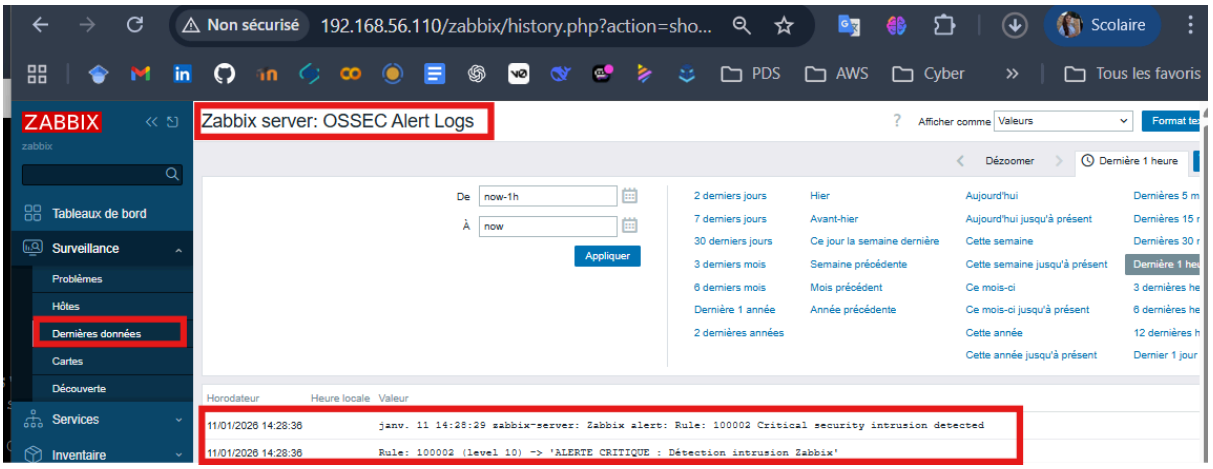


2. **Création du déclencheur** :
   - Dans l'onglet `Déclencheurs` de l'hôte.
   - Créer un nouveau déclencheur nommé `Intrusion détectée par OSSEC (Règle 100002)`.
   - Sévérité : `Haut`.
   - Expression : `find(/Zabbix server/log[/var/ossec/logs/alerts/alerts.log,"Rule: 100002"],,"like","Rule: 100002")>0`



Captures d'écran de la configuration Zabbix :

- Une capture montre la création de l'élément `OSSEC Alert Logs`.
- Une autre montre la configuration du déclencheur `Intrusion détectée`.
- Enfin, une capture de la section `Dernières données` affiche les logs d'alerte OSSEC remontés, confirmant que la valeur `Rule: 100002 (level 10) -> 'ALERTE CRITIQUE : ...'` est bien collectée par Zabbix.



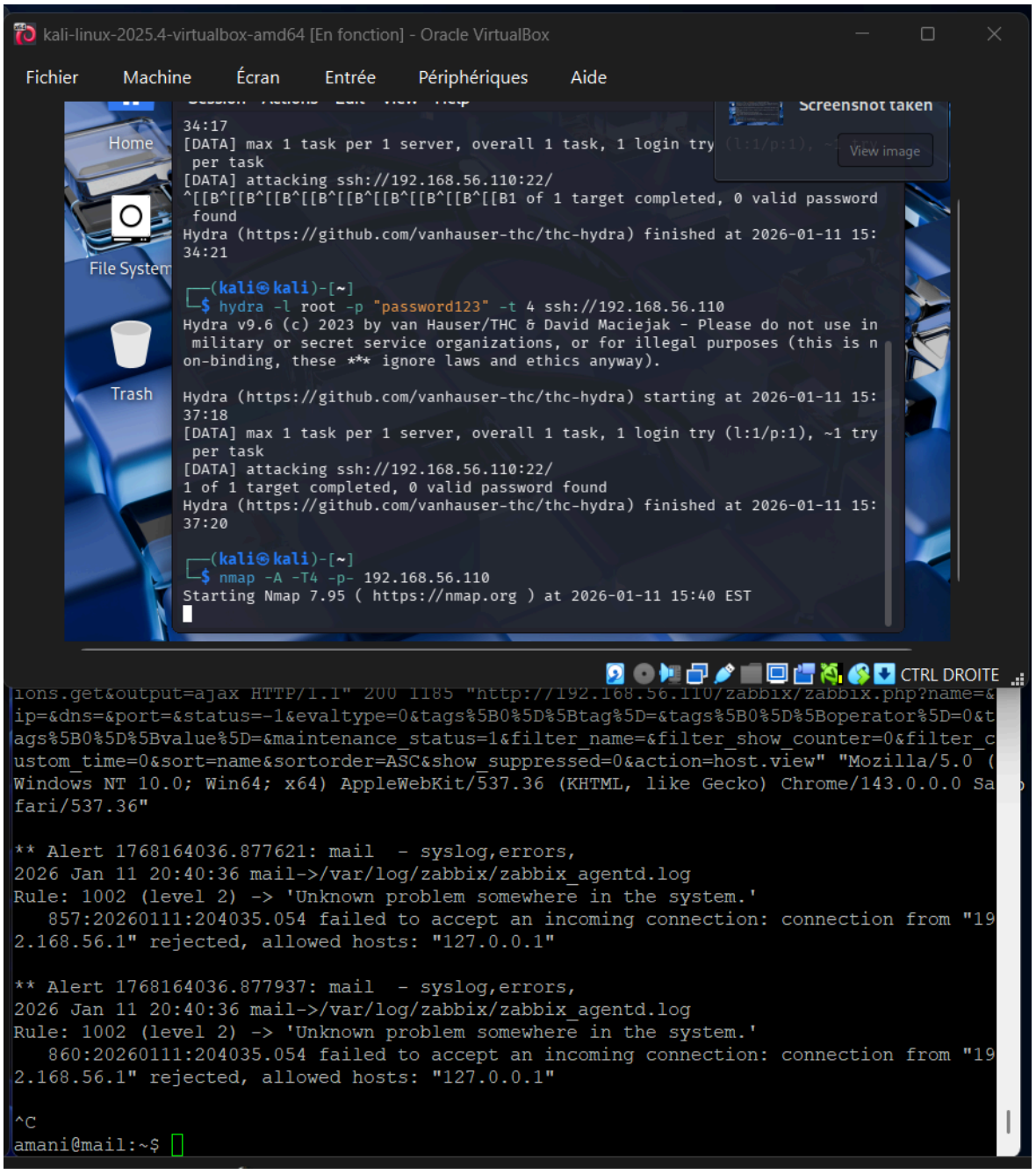
4.4 Scénario 2 : Simulations d'Attaques depuis Kali Linux

Des attaques sont lancées depuis une machine Kali Linux pour tester les capacités de détection en conditions réelles.

Attaque 1 : Scan de Ports Agressif avec Nmap

Un scan complet des ports est lancé pour identifier les services ouverts.

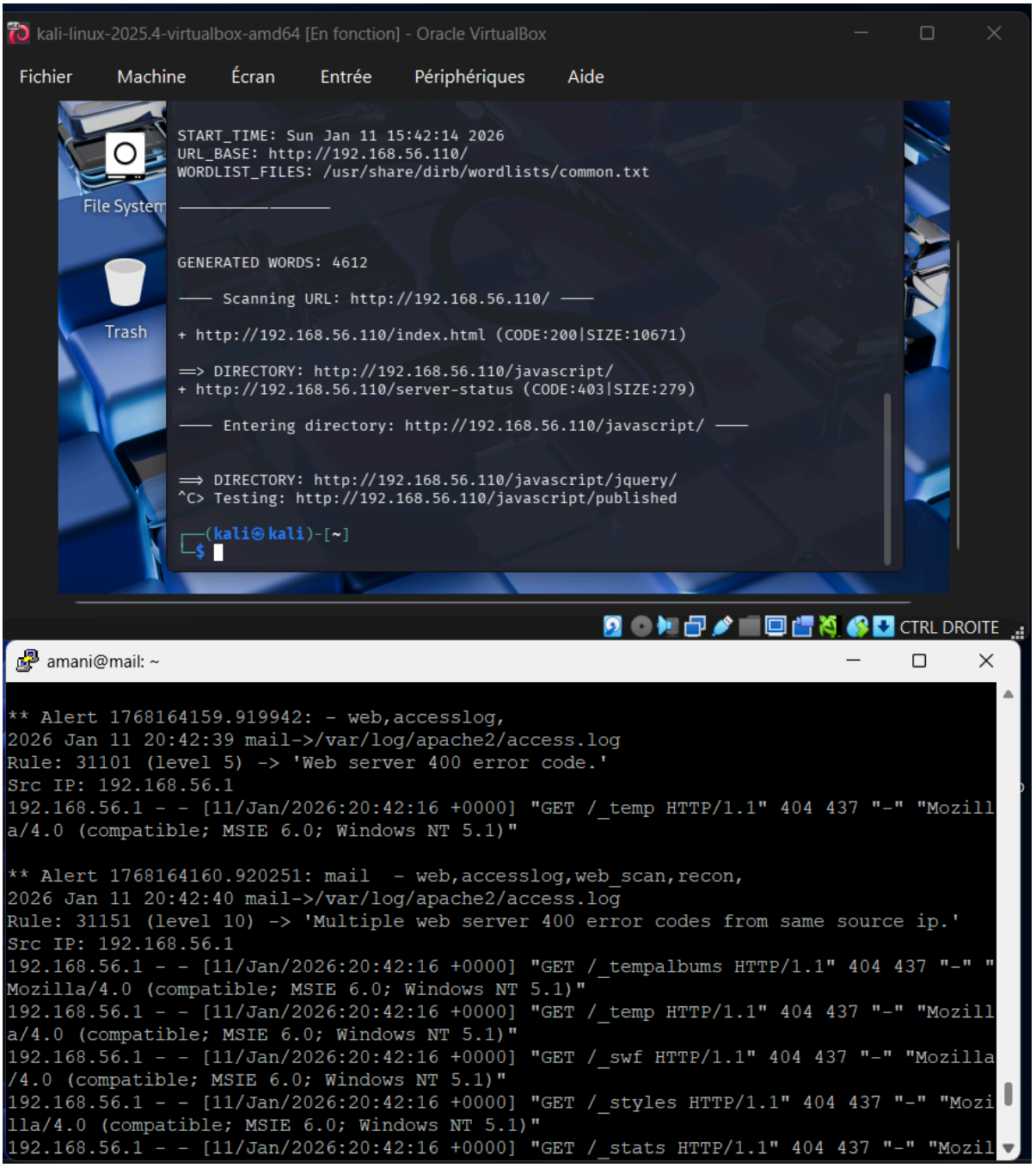
- **Commande sur Kali :**nmap -A -T4 -p- 192.168.56.110
- **Détection :** OSSEC et Zabbix (via les logs de l'agent) détectent une activité réseau anormale et de multiples tentatives de connexion échouées, générant des alertes.



Attaque 2 : Scan de Répertoires Web avec Dirb

Cette attaque cherche des répertoires web cachés, générant de nombreuses erreurs 404.

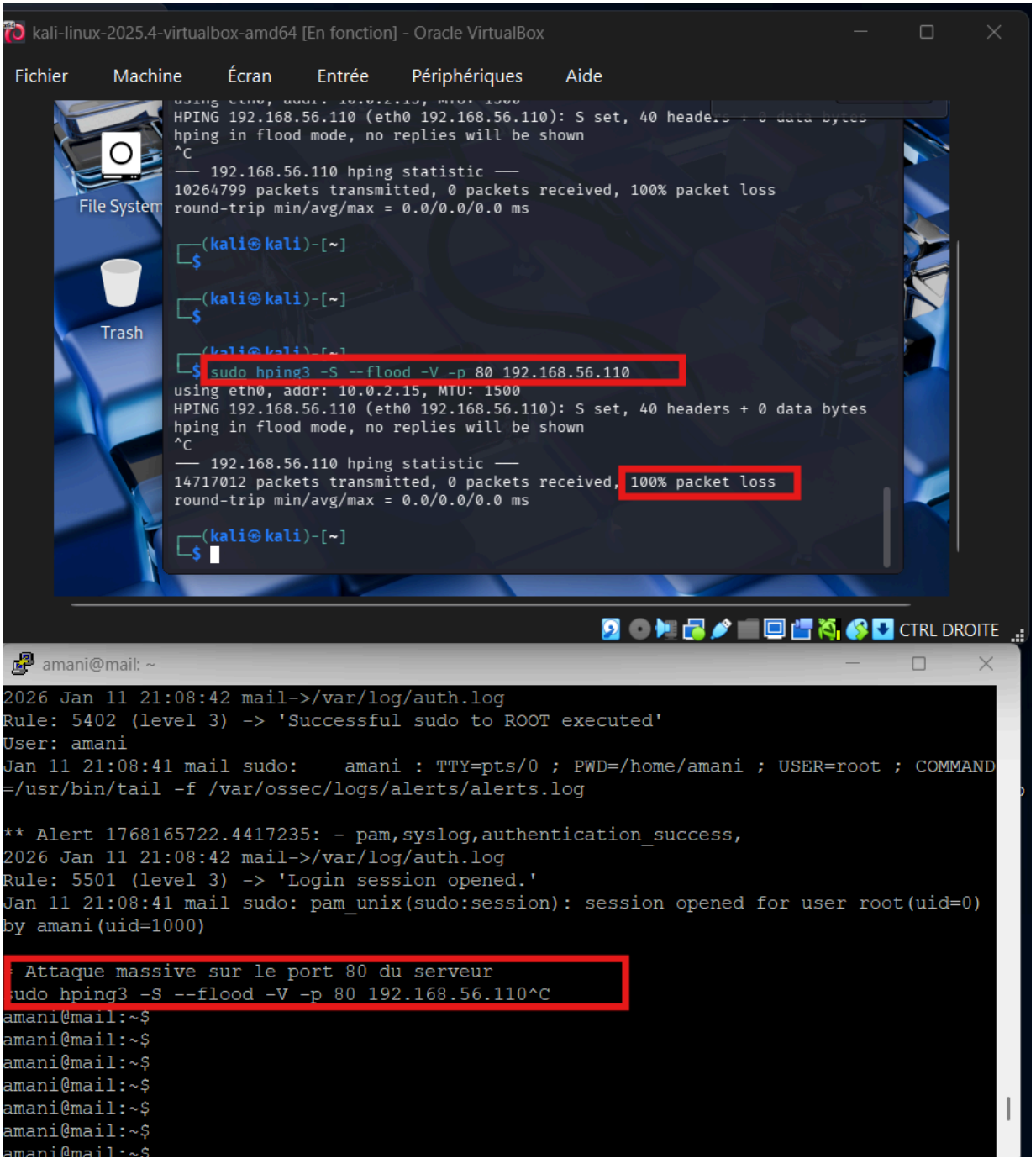
- **Commande sur Kali :** `dirb http://192.168.56.110`
- **Détection :** OSSEC, qui surveille `access.log` d'Apache, détecte un grand nombre d'erreurs 404 provenant de la même IP et déclenche une alerte de type "Web scanning". **Capture d'écran de la détection :** Le terminal du serveur SOC affiche des alertes OSSEC avec la règle `31151 (level 10)` -> `'Multiple web server 400 error codes from same source ip.'`



Attaque 3 : Déni de Service (DoS) avec hping3

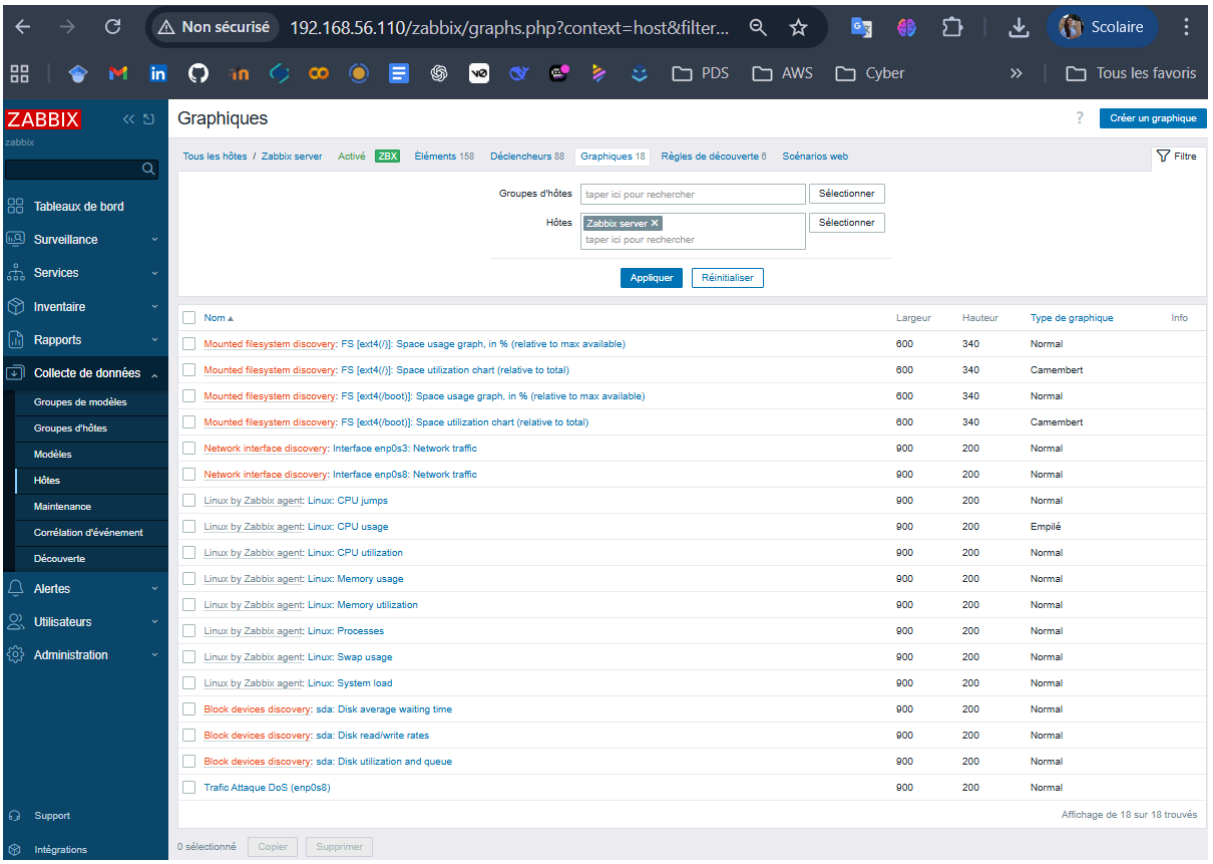
Une attaque de type SYN Flood est simulée pour saturer le serveur.

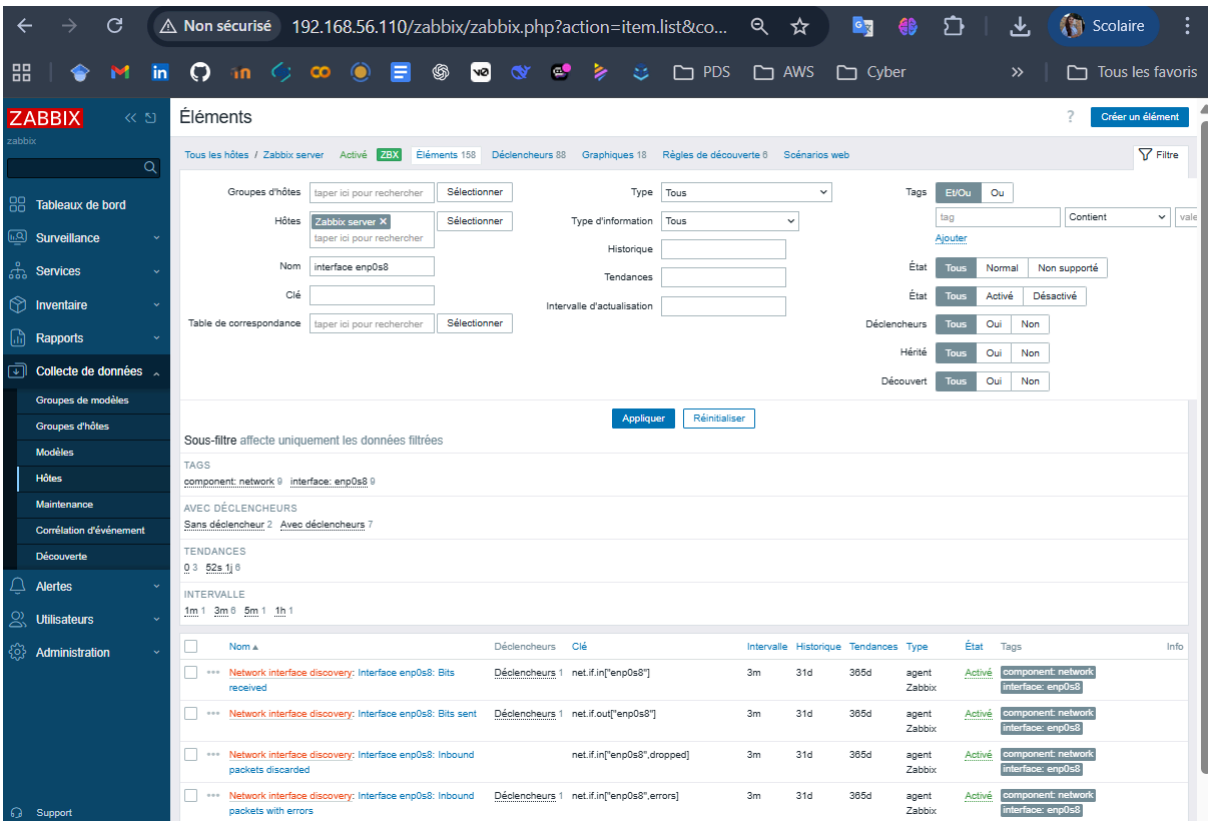
- **Commande sur Kali :** `sudo hping3 -S --flood -V -p 80 192.168.56.110`
- **Détection et Corrélation :**
  - **OSSEC** détecte une activité réseau anormale et peut générer des alertes sur les logs système.

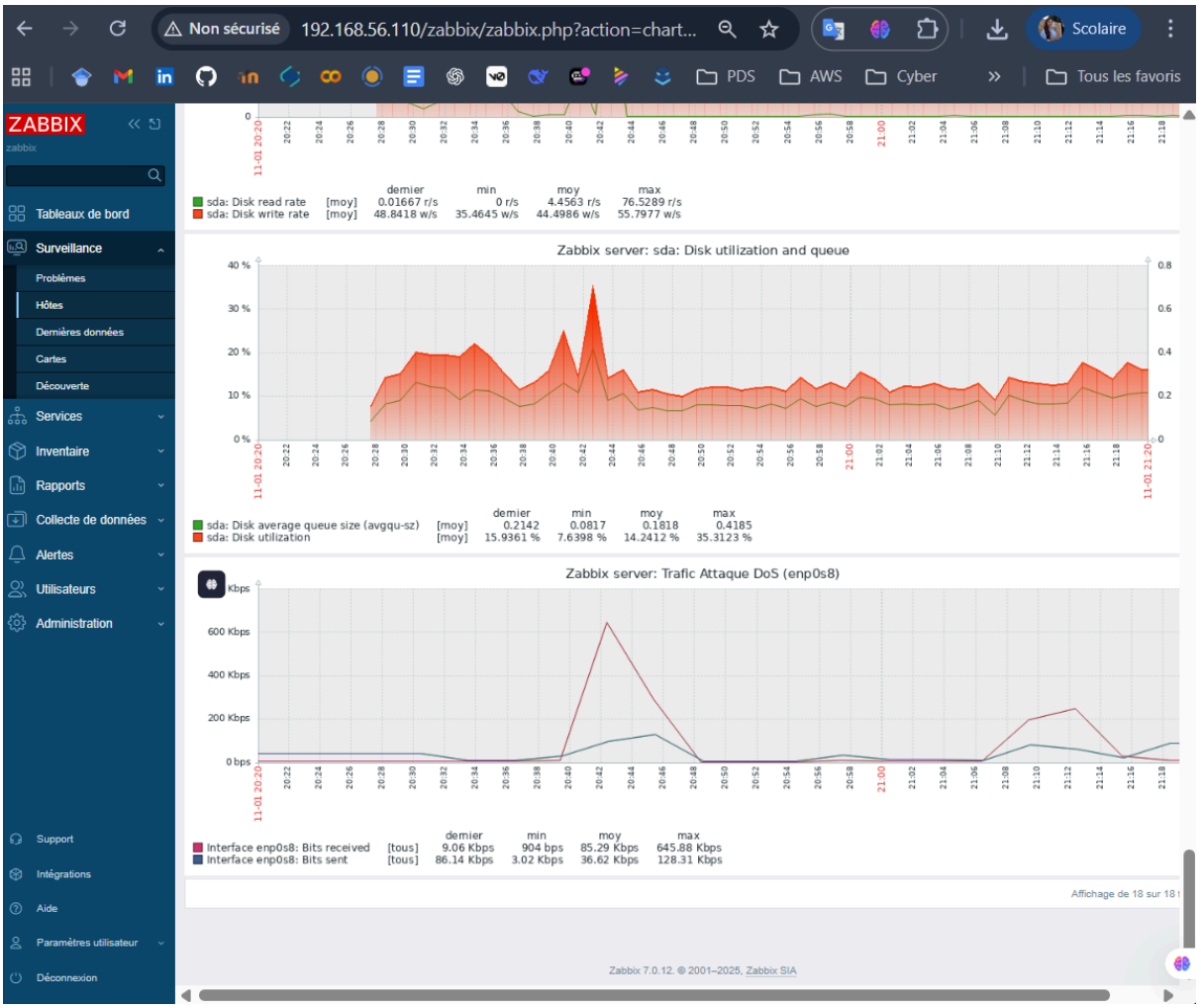


  - **Zabbix** fournit la vue la plus claire de l'impact :

Les graphiques montrent une augmentation spectaculaire de la charge CPU (`Linux: CPU utilization`), de l'utilisation de la mémoire (`Linux: Memory usage`) et surtout du trafic réseau entrant (`Network interface enp0s8: Bits received`).







Captures d'écran de l'impact : Les graphiques Zabbix montrent un pic très net sur le trafic réseau ("Traffic Attaque DoS") et sur l'utilisation des disques ("Disk utilization and queue") au moment de l'attaque, démontrant la capacité de la solution à corréler un événement de sécurité avec son impact sur la performance.