

Lab 02

Cybersecurity 3

Détection d'une attaque SSH avec Wazuh, conçu pour un déploiement facile et reproductible

Environnement Dockerisé

- Wazuh Manager + Dashboard + Indexer (stack standard Docker)
- Wazuh Agent sur Ubuntu Dockerisé
- Simulation d'une attaque SSH brute-force via un fichier auth.log

Etape 1. Docker Compose : Wazuh Manager + Agent

Créez un fichier docker-compose.yml :

```
version: '3.8'
```

```
services:
```

```
  wazuh-manager:
```

```
    image: wazuh/wazuh
```

```
    hostname: wazuh-manager
```

```
    ports:
```

```
      - "1514:1514"
```

```
      - "1515:1515"
```

```
      - "55000:55000"
```

```
      - "443:443"
```

```
    volumes:
```

```
      - wazuh_data:/var/ossec/data
```

```
  wazuh-agent:
```

```
    image: ubuntu:22.04
```

```
    hostname: wazuh-agent
```

```
    depends_on: [wazuh-manager]
```

```
    volumes:
```

```
- ./auth.log:/var/log/auth.log:ro
```

command: >

```
bash -c "
```

```
apt-get update &&
```

```
apt-get install -y curl apt-transport-https gnupg &&
```

```
curl -s https://packages.wazuh.com/key/GPG-KEY-WAZUH | apt-key add - &&
```

```
echo 'deb https://packages.wazuh.com/4.x/apt/ stable main' > /etc/apt/sources.list.d/wazuh.list &&
```

```
apt-get update &&
```

```
apt-get install -y wazuh-agent &&
```

```
sed -i 's/manager_address:./manager_address: wazuh-manager/' /etc/ossec.conf &&
```

```
service wazuh-agent start &&
```

```
tail -f /var/log/auth.log
```

```
"
```

volumes:

wazuh_data:

Etape 2. Fichier auth.log simulé (40 tentatives SSH)

Créez un fichier auth.log à côté :

```
#!/bin/bash
```

```
cat <<EOF > auth.log
```

```
EOF
```

```
for i in {1..40}; do
```

```
    echo "Jul 21 12:00:0$i ubuntu sshd[12345]: Failed password for invalid user user$i from 192.168.0.$((i%5+1)) port $((5000+i)) ssh2" >> auth.log
```

```
done
```

```
echo "Jul 21 12:01:00 ubuntu sshd[12345]: Accepted password for validuser from 192.168.0.1 port 5000 ssh2" >> auth.log
```

Etape 3. Démarrage du docker-compose

```
docker-compose up -d
```

- wazuh-manager expose le dashboard sur le port 443 (<https://localhost>)
- wazuh-agent monte et "joue" auth.log

Etape 4. Créer une règle Wazuh locale

Dans le container wazuh-manager :

```
docker exec -it <container_name> bash
```

```
cat <<EOF >> /var/ossec/etc/rules/local_rules.xml
```

```
<group name="sshd,">
```

```
<rule id="100001" level="10">
```

```
<if_sid>5710</if_sid>
```

```
<field name="srcip">.*</field>
```

```
<frequency>5</frequency>
```

```
<timeframe>60</timeframe>
```

```
<description>Plus de 5 tentatives SSH échouées en 1 minute</description>
```

```
</rule>
```

```
</group>
```

```
EOF
```

```
service wazuh-manager restart
```

- if_sid=5710 correspond aux logs SSH échoués
- Alerte déclenchée dès 6 tentatives en moins d'une minute
[documentation.wazuh.com+15documentation.wazuh.com+15m.youtube.com+15medium.com+4medium.com+4documentation.wazuh.com+4documentation.wazuh.com+3medium.com+3medium.com+3github.com+1medium.com+1](https://documentation.wazuh.com/1.5/documentation.wazuh.com+15m.youtube.com+15medium.com+4medium.com+4documentation.wazuh.com+4documentation.wazuh.com+3medium.com+3medium.com+3github.com+1medium.com+1)

Etape 5. Vérification de l'alerte sur le dashboard

- Accédez au dashboard (<https://localhost>)
- Allez à Security Events → recherchez l'ID 100001
- Capturez l'écran contenant l'alerte (livrable)

Pour exporter l'alerte en JSON depuis l'API :

```
curl -k -u admin:SecretPassword "https://localhost:55000/alerts?rule_id=100001" > alert.json
```

Livrables attendus

1. Capture d'écran du dashboard montrant l'alerte déclenchée
2. Fichier local_rules.xml avec la règle personnalisée
3. Rapport sur l'incident simulé : description, timeline des logs, IPs sources