



De la Supervision à la Réponse Active

Intégration de Zabbix & OSSEC : Preuve de Concept pour un SOC

Introduction des Outils : Le Surveillant et le Gardien



Zabbix - Le Surveillant

Rôle : Supervision des performances et de la disponibilité des systèmes.

Points Forts : Collecte de métriques (CPU, RAM, réseau), tableaux de bord personnalisables, système d'alertes flexible.

Le Défi : La surveillance des performances (Zabbix) et la détection de sécurité (OSSEC) opèrent souvent en silos.



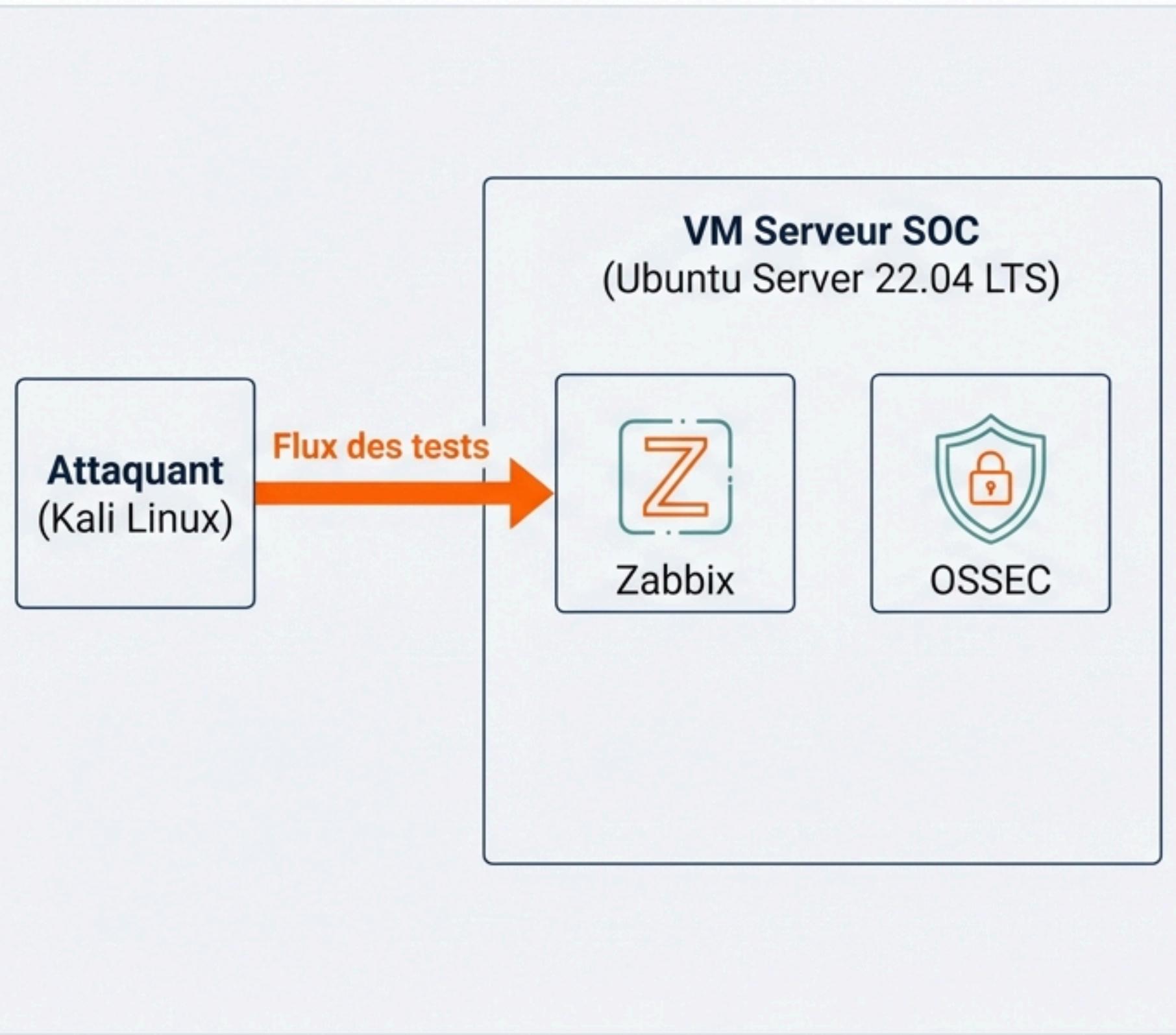
OSSEC - Le Gardien

Rôle : Système de Détection d'Intrusion Basé sur l'Hôte (HIDS).

Points Forts : Analyse de logs, contrôle d'intégrité des fichiers, détection de rootkits, réponse active.

L'Objectif : Fusionner ces deux mondes pour obtenir une vue unifiée et déclencher des réponses de sécurité basées sur des événements corrélés.

Architecture de la Solution et Environnement de Test



Spécifications Techniques Clés

- **OS** : Ubuntu Server 22.04 LTS
- **RAM** : 4 Go
- **CPU** : 2 cœurs
- **Stockage** : 50 Go
- **IP Serveur** : 192.168.56.110

Prérequis Essentiels

Configuration du pare-feu (UFW) pour les ports critiques :

- 22 (SSH)
- 80/443 (Web)
- 10050/10051 (Zabbix)

```
aman@kali:~$ sudo ufw enable
Firewall is active and enabled on system startup
aman@kali:~$ sudo ufw allow 22/tcp
Skipping adding existing rule
Skipping adding existing rule (v6)
aman@kali:~$ sudo ufw allow 80/tcp
Rule added
Rule added (v6)
aman@kali:~$ sudo ufw allow 443/tcp
Rule added
Rule added (v6)
aman@kali:~$ sudo ufw allow 10050/tcp
Rule added
Rule added (v6)
aman@kali:~$ sudo ufw allow 10051/tcp
Rule added
Rule added (v6)
```

Étape 1 : Mise en Place du Surveillant (Zabbix 7.0)

Flux de travail en 4 points clés :

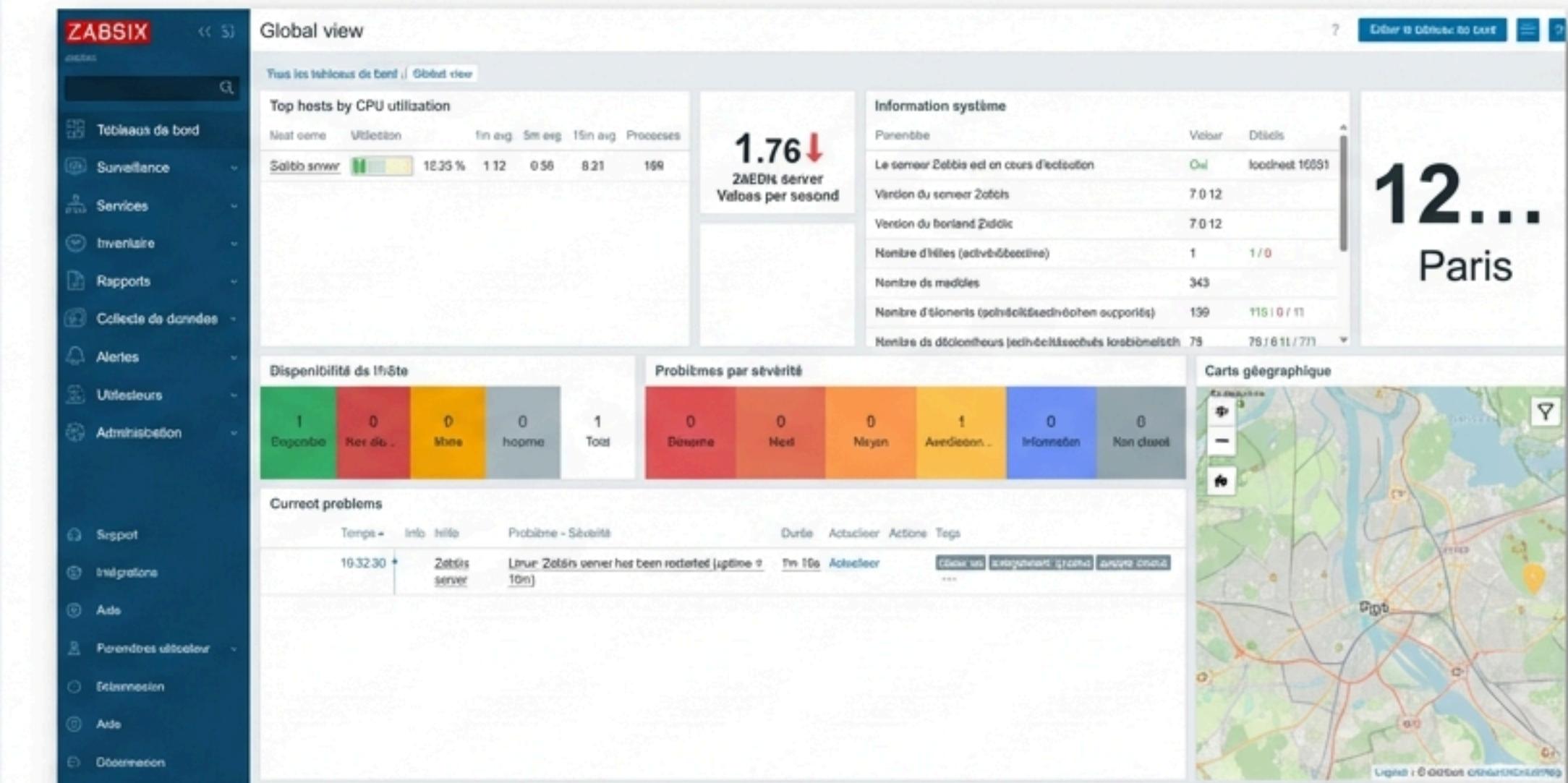


Installation des Dépôts : Ajout du référentiel officiel Zabbix.

Configuration de la Base de Données : Installation de MySQL et création d'une base et d'un utilisateur dédiés ('zabbix').

Installation des Composants : Déploiement de 'zabbix-server-mysql', 'zabbix-frontend-php' et 'zabbix-agent'.

Configuration Finale : Liaison du serveur Zabbix à sa base de données et configuration via l'interface web.



Étape 2 : Déploiement du Gardien (OSSEC 3.7.0)

Processus d'Installation Synthétisé

- ⚙️ **Dépendances** : Installation des outils de compilation (`build-essential`, `gcc`, etc.).
- ⬇️ **Compilation** : Téléchargement des sources depuis GitHub et compilation.
- **Installation Interactive (`./install.sh`)** :
 - **Choix Crucial** : Mode d'installation `local` (serveur et agent sur la même machine).
 - **Activation des Fonctions Clés** :
 - Contrôle d'intégrité des fichiers : [y]
 - Détection de rootkits : [y]
 - Réponse active : [y] <--- ⚡**
 - Activation du `firewall-drop` : [y]

Légende : Démarrage des services

```
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.
```

Légende : Vérification du statut

```
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-exscd is running...
```

L'Intégration : Créer le Pont entre Zabbix et OSSEC

Objectif : Configurer OSSEC pour qu'il surveille les journaux de Zabbix et réagisse à des événements spécifiques.

1. L'Écoute (ossec.conf)

Ajout de la surveillance des fichiers de log de Zabbix.

```
<!-- Surveiller les logs Zabbix -->
<localfile>
    <log_format>syslog</log_format>
    <location>/var/log/zabbix/zabbix_serv
er.log</location>
</localfile>
```

Analyse

2. La Règle (local_rules.xml)

Création d'une règle personnalisée (ID `100002`) pour identifier une alerte de sécurité critique provenant de Zabbix.

```
<rule id="100002" level="10">
    <match>Zabbix alert</match>
    <description>Alerte de sécurité
critique venant de Zabbix</description>
</rule>
```

Test de l'Intégration : La Preuve de Concept (PoC)

1. Action (Injection)

```
Terminal × + - □ ×  
amani@mail:~$  
amani@mail:~$ echo "Zabbix alert: Critical security intrusion"  
| sudo tee -a /var/log/zabbix/zabbix_server.log  
Zabbix alert: Critical security intrusion  
amani@mail:~$
```

1. Action (Injection)

2. Réaction (Détection)

```
Terminal × + - □ ×  
amani@mail:~$ tail -f /var/ossec/logs/alerts/alerts.log  
Jan 11 13:37:41 zabbix-server/zabbix_server.log  
** Alert 1768138661.457966: mail - local,zabbix_alerts, 2026  
Jan 11 13:37:41 mail->/var/log/zabbix/zabbix_server.log  
Rule: 100002 (level 10) -> 'ALERTE CRITIQUE : Détection  
intrusion Zabbix'  
Jan 11 13:37:40 zabbix-server: Zabbix alert: Critical security  
intrusion  
Jan 11 13:37:40 zabbix-server: Zabbix-server in alerts_server.  
■
```

2. Réaction (Détection)

Résultat : La Chaîne Fonctionne

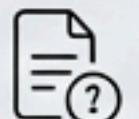
- Analyse de l'alerte :
 - Fichier surveillé : /var/log/zabbix/zabbix_server.log
 - Règle déclenchée : 100002 (level 10)
 - Description : 'Alerte de sécurité critique venant de Zabbix'

Boucler la Boucle : Visualisation de l'Alerte OSSEC dans Zabbix

Méthode



- Création d'un Élément (Item) :** Zabbix est configuré pour lire le fichier d'alertes d'OSSEC (/var/ossec/logs/alerts/alerts.log) et y chercher les alertes correspondant à notre règle.



- Création d'un Déclencheur (Trigger) :** Un déclencheur est défini pour passer à l'état 'Problème' si une entrée contenant 'Rule: 100002' est trouvée dans l'élément.

The screenshot shows the Zabbix interface with the 'Triggers' tab selected. A new trigger is being added, with the name 'Intrusion détectée par OSSEC (Règle: 100002)' and the condition 'final[Zabbix server|log/var/ossec/logs/alerts/alerts.log;Rule: 100002]'. The 'Active' status is checked.

Configuration du Déclencheur

The screenshot shows the 'Last Data' collection page. It lists two data points: 'Demibrés dérobés' at 11/01/2028 14:28:38 and 'Alerts' at 11/01/2028 14:35:38. Both entries are marked as 'Série'.

Données Collectées

Conclusion : Le système est maintenant capable de détecter un événement de sécurité (OSSEC) et de le présenter comme un problème de supervision (Zabbix).

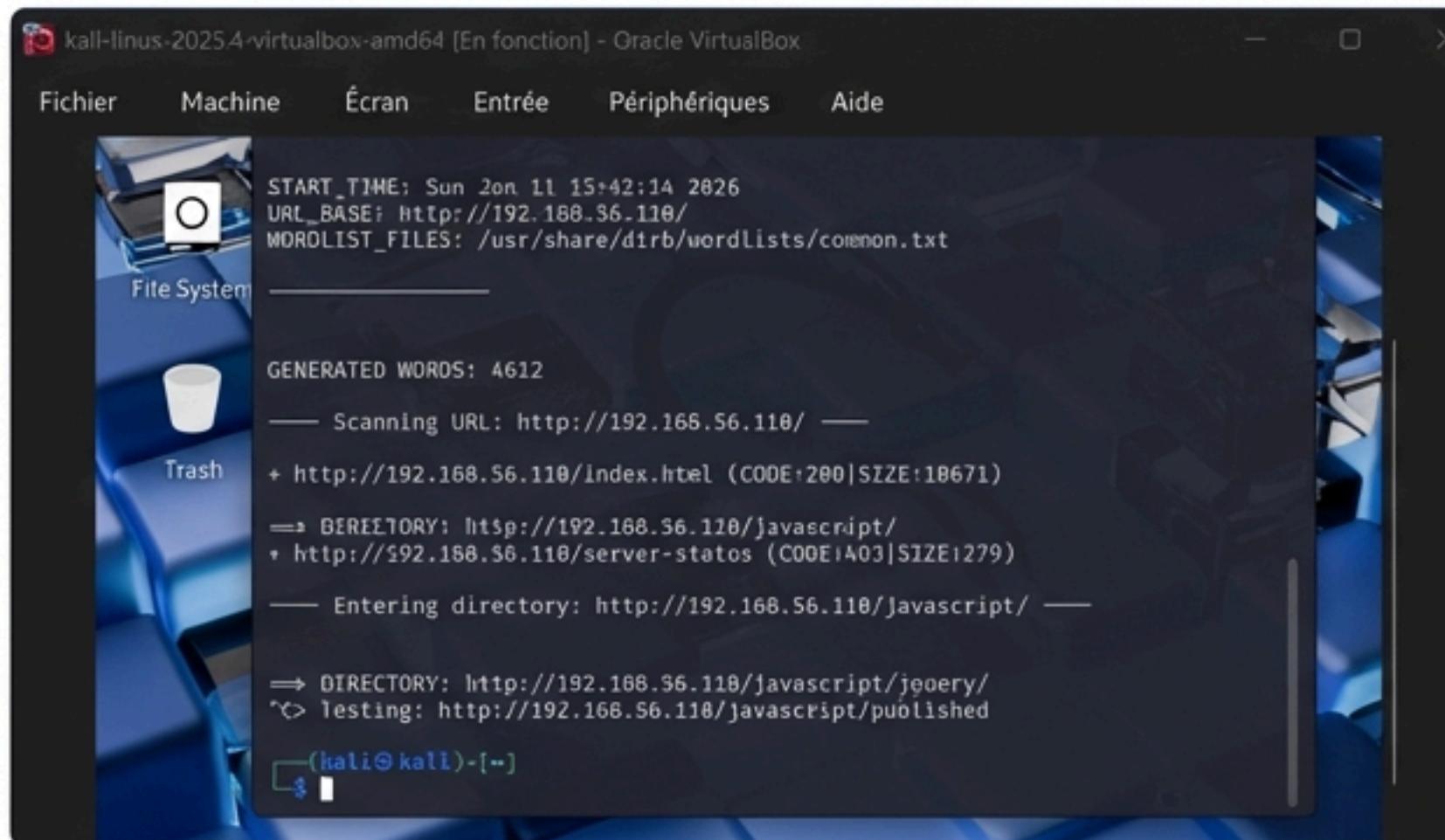
Tests Réels : Le Scan de Répertoires Web (Dirb)

Scénario d'Attaque

Outil : `Dirb` sur Kali Linux.

Objectif de l'attaquant : Découvrir des répertoires ou des fichiers cachés sur le serveur web.

Commande : `dirb http://192.168.56.110`



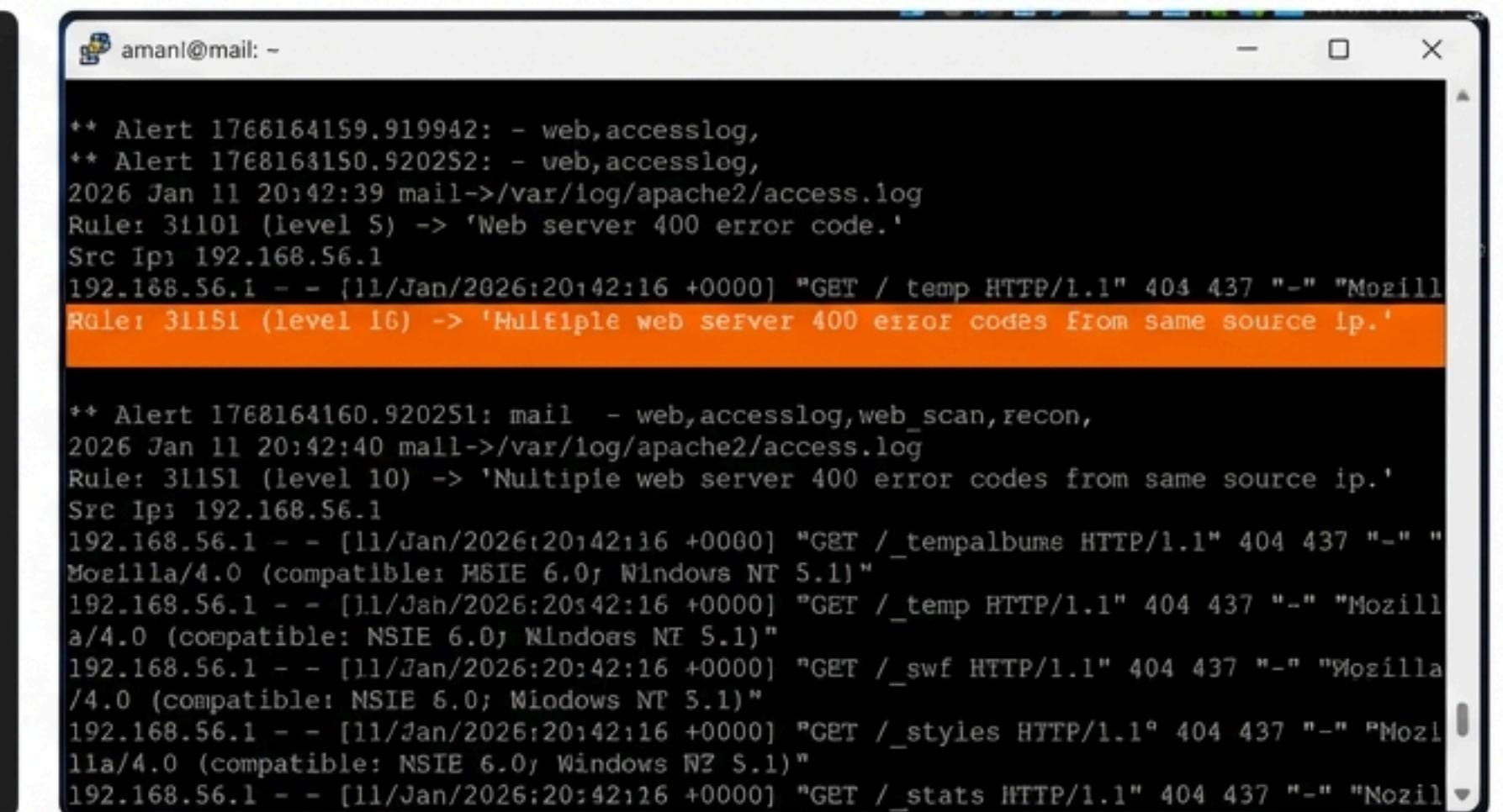
```
kali@kali:~$ dirb http://192.168.56.110/ [En fonction] - Oracle VirtualBox
Fichier Machine Écran Entrée Périphériques Aide
File System
Trash
START_TIME: Sun Jan 11 15:42:14 2026
URL_BASE: http://192.168.56.110/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt
GENERATED WORDS: 4612
— Scanning URL: http://192.168.56.110/ —
+ http://192.168.56.110/index.html (CODE:200|SIZE:18671)
==> DIRECTORY: https://192.168.56.110/javascript/
+ http://192.168.56.110/server-status (CODE:403|SIZE:1279)
— Entering directory: http://192.168.56.110/Javascript/ —
==> DIRECTORY: http://192.168.56.110/javascript/jquery/
^> testing: http://192.168.56.110/javascript/published
(kali㉿kali)-[..]
```

Console de l'Attaquant (Kali Linux)

Mécanisme de Détection

L'attaque génère une vague d'erreurs HTTP 404 dans les logs Apache.

OSSEC, via sa règle native 'ID 31151', détecte l'accumulation d'erreurs 404 depuis la même IP.



```
amani@mail: ~
** Alert 1766164159.919942: - web,accesslog,
** Alert 1766164150.920252: - web,accesslog,
2026 Jan 11 20:42:39 mail->/var/log/apache2/access.log
Rule: 31101 (level S) -> 'Web server 400 error code.'
Src Ip: 192.168.56.1
192.168.56.1 - - [11/Jan/2026:20:42:16 +0000] "GET / temp HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible: NSIE 6.0; Windows NT 5.1)"
Rule: 31151 (level 1G) -> 'Multiple web server 400 error codes From same source ip.'

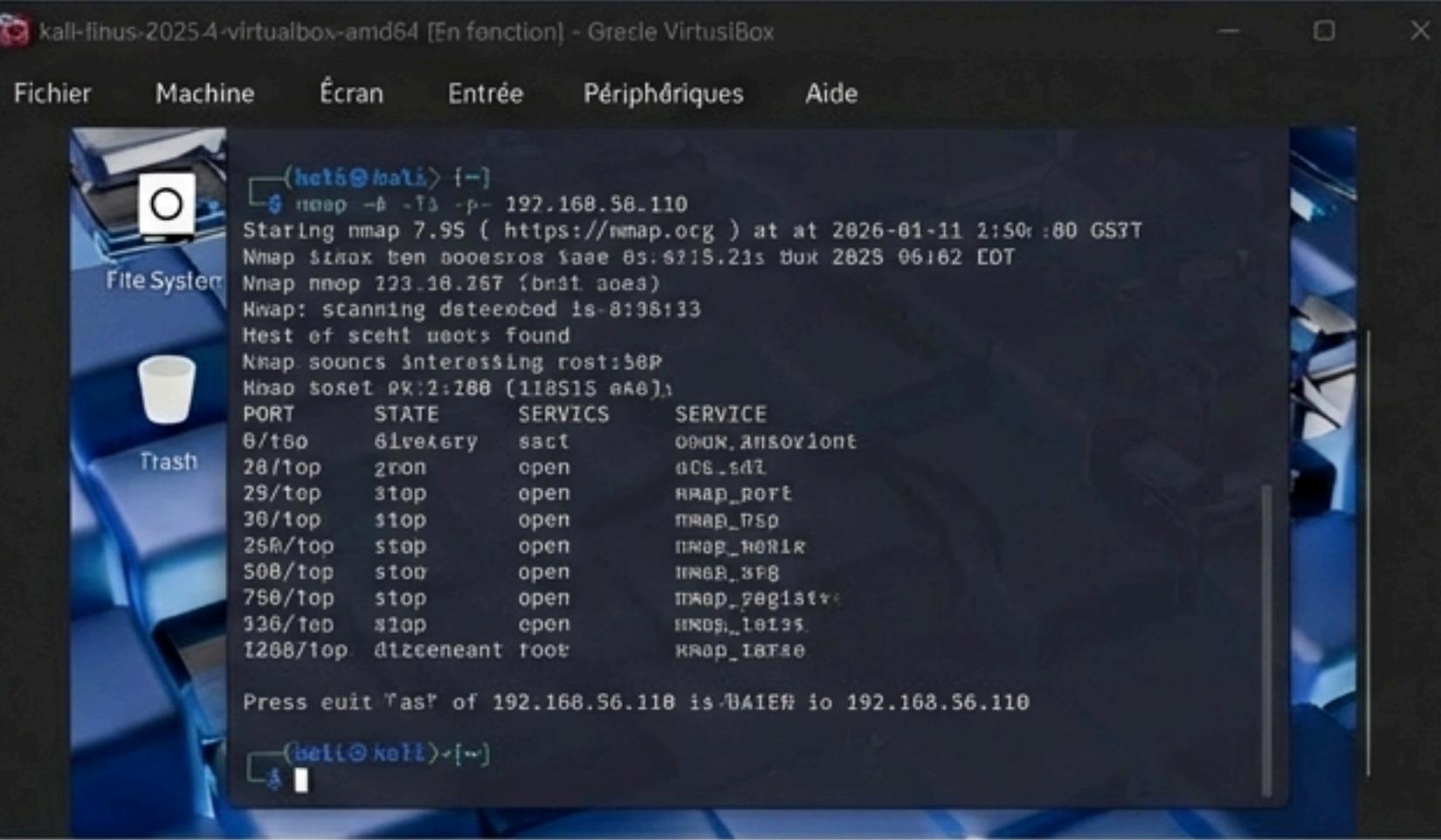
** Alert 1768164160.920251: mail - web,accesslog,web_scan,recon,
2026 Jan 11 20:42:40 mail->/var/log/apache2/access.log
Rule: 31151 (level 10) -> 'Multiple web server 400 error codes from same source ip.'
Src Ip: 192.168.56.1
192.168.56.1 - - [11/Jan/2026:20:42:16 +0000] "GET /_tempalbume HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible: NSIE 6.0; Windows NT 5.1)"
192.168.56.1 - - [11/Jan/2026:20:42:16 +0000] "GET /_temp HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible: NSIE 6.0; Windows NT 5.1)"
192.168.56.1 - - [11/Jan/2026:20:42:16 +0000] "GET /_swf HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible: NSIE 6.0; Windows NT 5.1)"
192.168.56.1 - - [11/Jan/2026:20:42:16 +0000] "GET /_styles HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible: NSIE 6.0; Windows NT 5.1)"
192.168.56.1 - - [11/Jan/2026:20:42:16 +0000] "GET /_stats HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible: NSIE 6.0; Windows NT 5.1)"
```

Alerte sur le Serveur SOC (OSSEC)

Tests Réels : Le Scan de Ports Agressif (Nmap)

Scénario d'Attaque

- **Outil** : `Nmap` sur Kali Linux.
- **Objectif de l'attaquant** : Identifier tous les services et ports ouverts sur le serveur SOC.
- **Commande** : `nmap -A -T4 -p- 192.168.56.110`



```
(kali㉿kali)-[~]
└─$ nmap -A -T4 -p- 192.168.56.110
Starting nmap 7.95 ( https://nmap.org ) at 2026-01-11 21:50:00 GSST
Nmap scan report for 192.168.56.110
Nmap nmap 223.18.267 (bnat. aoea)
Nmap: scanning dateexecuted is 8:08:33
Host is up. No sctt mcts found
Nmap sounds interesting port:58P
Nmap soiset 8K:2:188 (118515 eA8):
PORT      STATE      SERVICE      SERVICE
8/tso      6ivetary  sact        000K, amsoorient
28/tcp     2zon      open        006..sd1
29/tcp     3top      open        RRAP_R0P
30/tcp     3top      open        RRAP_RSP
258/tcp    3top      open        RRAP_HERIK
508/tcp    3top      open        RRAB_3FB
758/tcp    3top      open        RRAP_Registr
326/tcp    3top      open        RNDIS_18135
1288/tcp   3tzeeneant  foor      RRAP_IATSE
Press exit 'last' of 192.168.56.110 is 0A1EW in 192.168.56.110
(kali㉿kali)-[~]
```

Console de l'Attaquant (Kali Linux)

Mécanisme de Détection

Un scan agressif génère un comportement réseau anormal et des entrées spécifiques dans les logs système, identifiés par les règles natives d'OSSEC.

```
** Alert 1768164036.877621: mail - syslog,errors,
2026 Jan 11 15:40:36 mail->/var/log/syslog,errors,
2026 Jan 11 15:40:36 mail->/var/log/syslog.log
Rule: 100010 (level 10) -> "Nmap port scan"
860:20260111:15:40:36 Nmap scanning detected from: 192.168.56.1
860:20260111:15:40:36 Nmap scanning detected
** Alert 1768164036.877621: mail - syslog,errors,
2026 Jan 11 15:40:36 mail->/var/log/syslog.log
Rule: 100010 (level 10) -> "Nmap port scan"
860:20260111:15:40:36 Nmap scanning detected
860:20260111:15:40:36 Nmap scanning detected from: 192.168.56.1
** Alert 1768164036.877621: mail - syslog,errors,
2026 Jan 11 15:40:36 mail->/var/log/syslog.log
Rule: 100010 (level 10) -> "Nmap port scan"
860:20260111:15:40:36 Nmap scanning detected
860:20260111:15:40:36 Nmap scanning detected
```

Alerte sur le Serveur SOC (OSSEC)

Tests Réels : L'Attaque par Déni de Service (hping3)

Le Scénario d'Attaque

- Outil :** hping3 sur Kali Linux.
- Objectif :** Submerger le port 80 du serveur avec un 'SYN Flood'.
- Commande :** sudo hping3 -S --flood -V -p 80 192.168.56.110

```
amani@Mali:~  
Fichier Acctine Entrée Périphériques Aide  
  
[(bali㉿kali)-[~]]$ sudo hping3 -S --flood -V -p 80 192.168.56.110  
[sudo] password for kali:  
using eth0, addr: 20.0.2.15, MTU: 1500  
HPING 192.168.56.110 (eth0 192.168.56.110): S set, 40 headers + 0 data bytes  
hping in flood node, no replies will be shown  
-- 192.168.56.110 hping statistic --  
1158378 packets transmitted, 0 packets received, 100% packet loss  
round-trip min/avg/max = 0.0/0.0/0.0 ms  
[(bati㉿kali)-[~]]$
```

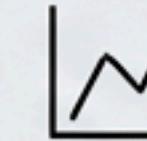
Détection sur Deux Fronts



Perspective Sécurité (OSSEC)

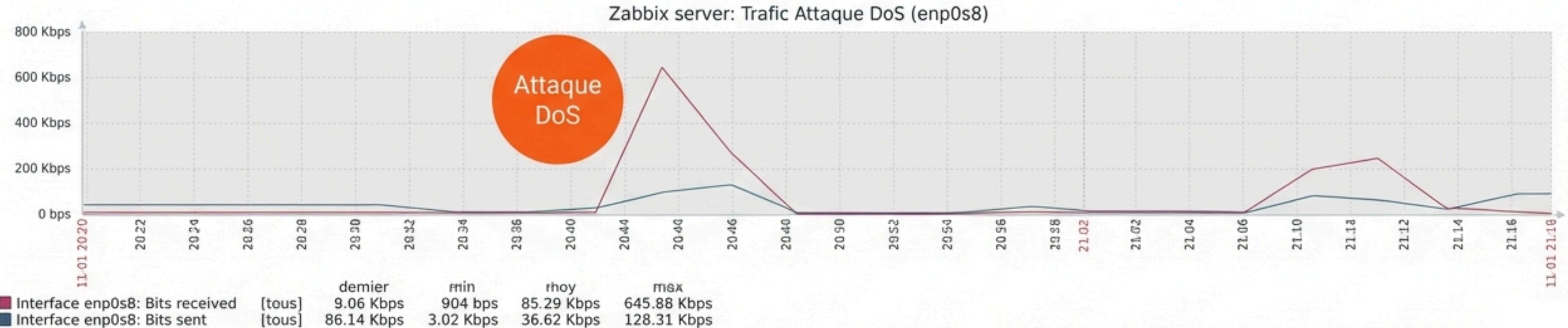
OSSEC détecte le comportement réseau anormal et l'afflux massif de paquets.

```
-- Alert 176618A329.LIB9A81: - web,accesslog,  
2018 Jan 11 20:41:50 mail-n/ves/log/soouchet/access.log  
Rm1s: 31101 (level 1) => "Unr- record 406 otros cedes."  
192.168.56.1 - - [11/Jan/2018:20:41:18 +0000] "GET /oetherized_keys HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"  
  
-- Alert 1766184129.LIB8721: - web,accesslog,  
2018 Jan 11 20:43:20 mail-n/raa/ligraacachei/access.log  
Rm1s: 31101 (level 5) => "066 server 400 error cede."  
192.168.56.1 - - [11/Jan/2018:20:43:18 +0000] "GET /authosers HTTP/1.1" 404 437 "--" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
```



Perspective Performance (Zabbix)

Zabbix affiche en temps réel l'impact : augmentation drastique du trafic réseau et de l'utilisation du CPU.



Discussion : L'Intérêt pour un SOC

Valeur Ajoutée Démontrée



Vue Unifiée

Corrélation directe entre les événements de sécurité (OSSEC) et l'impact sur la performance (Zabbix). Le test DoS en est la parfaite illustration.



Automatisation de la Réponse

Capacité de déclencher des actions (ex: `firewall-drop`) automatiquement suite à la détection, réduisant le temps de réaction.



Détection Multi-couches

Le système a prouvé sa capacité à détecter différents types de menaces : reconnaissance web (`Dirb`), scan de ports (`Nmap`), et attaque par déni de service (`hping3`).



Solution Économique et Puissante

Utilisation d'outils open-source robustes et éprouvés, offrant une alternative viable aux solutions propriétaires coûteuses.

Limites et Points de Vigilance



Complexité de la Configuration

- La création de règles personnalisées dans OSSEC demande une expertise et peut être complexe. Le système n'est pas "plug-and-play".
- Le risque de faux positifs est réel et nécessite un affinage constant des règles pour éviter de bloquer du trafic légitime.



Périmètre du PoC

- La réponse active testée ici est basique (blocage d'IP). Des scénarios plus complexes nécessiteraient des scripts plus élaborés.
- Cette solution est installée sur un seul serveur. Un déploiement à grande échelle (multi-agents) requiert une architecture plus complexe.



Gestion des Logs

- Sans une solution de centralisation de logs (comme un SIEM), l'analyse d'un grand volume d'alertes peut devenir difficile.

Perspectives et Intégrations Possibles

Améliorer la Réponse

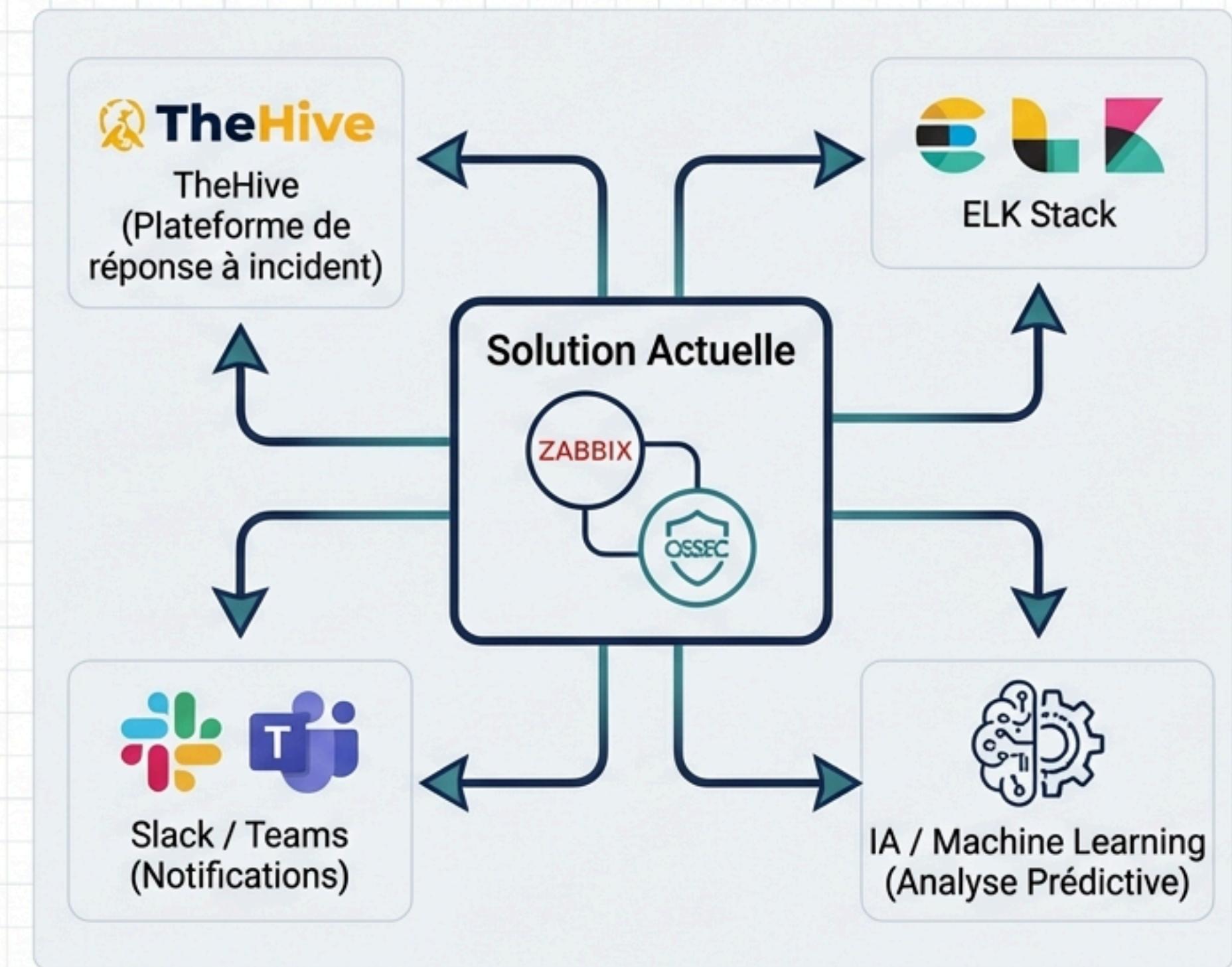
- Développer des scripts de réponse active plus intelligents (ex: isoler un conteneur, notifier sur Slack/Teams, créer un ticket dans Jira).

Enrichir la Détection

- Intégrer des flux de renseignement sur les menaces (Threat Intelligence).
- Utiliser des outils d'analyse de logs plus avancés.

Centraliser et Visualiser

- Envoyer les alertes vers un SIEM comme ELK Stack ou Splunk pour une corrélation et une visualisation avancées.





Merci de votre attention.

Questions ?
