ECOLE NORMALE SUPÉRIEURE DE L'ENSEIGNEMENT
TECHNIQUE DE MOHAMMEDIA
UNIVERSITÉ HASSAN II DE CASABLANCA

Département Mathématique et Informatique

Filière : « BDCC-II »

Compte rendu

Atelier : Sécurité des endpoints et supervision SIEM : étude de cas multi-OS (Linux & Windows)

Réalisé par :

Marouane Mounir

Encadré par :

Pr. Azeddine KHIAT

Année universitaire : 2025/2026

Table de Matière :

Liste des Figures :

1. Introduction

Dans le cadre de cet atelier pratique, nous avons mis en œuvre une plateforme complète de supervision et de protection de la sécurité des systèmes d'information, basée sur la solution **Wazuh**, combinant les approches **SIEM (Security Information and Event Management)** et **EDR (Endpoint Detection and Response)**.

L'environnement a été déployé sur **AWS Learner Lab** et repose sur une architecture Cloud intégrant des systèmes **Linux** et **Windows**, représentatifs d'un contexte d'entreprise réel. L'objectif principal de cet atelier est de comprendre, configurer et exploiter une solution SOC moderne permettant la **collecte**, la **corrélation**, l'**analyse** et la **détection des menaces** en temps réel.

2. Objectifs de l'atelier

Les objectifs principaux de ce travail sont :

- Mettre en place une architecture SIEM/EDR fonctionnelle dans le Cloud
- Superviser des endpoints Linux et Windows
- Centraliser et analyser les événements de sécurité
- Illustrer les concepts de **Endpoint Security**, **IAM/PAM** et **Threat Hunting**
- Produire des alertes exploitables à des fins SOC

3. Architecture du lab

L'architecture mise en place repose sur une VPC AWS unique, dans lequel trois instances EC2 communiquent de manière sécurisée.

3.1. Composants

EC2-1 : Wazuh Server (Ubuntu 24.04)

- Wazuh Manager
- Wazuh Indexer

- Wazuh Dashboard

EC2-2 : Client Linux (Ubuntu 24.04)

- Wazuh Agent

EC2-3 : Client Windows (Windows Server)

- Wazuh Agent

3.2. Flux réseau

- Agents → Serveur Wazuh : TCP 1514
- Enrôlement des agents : TCP 1515
- Accès au Dashboard : HTTPS 443

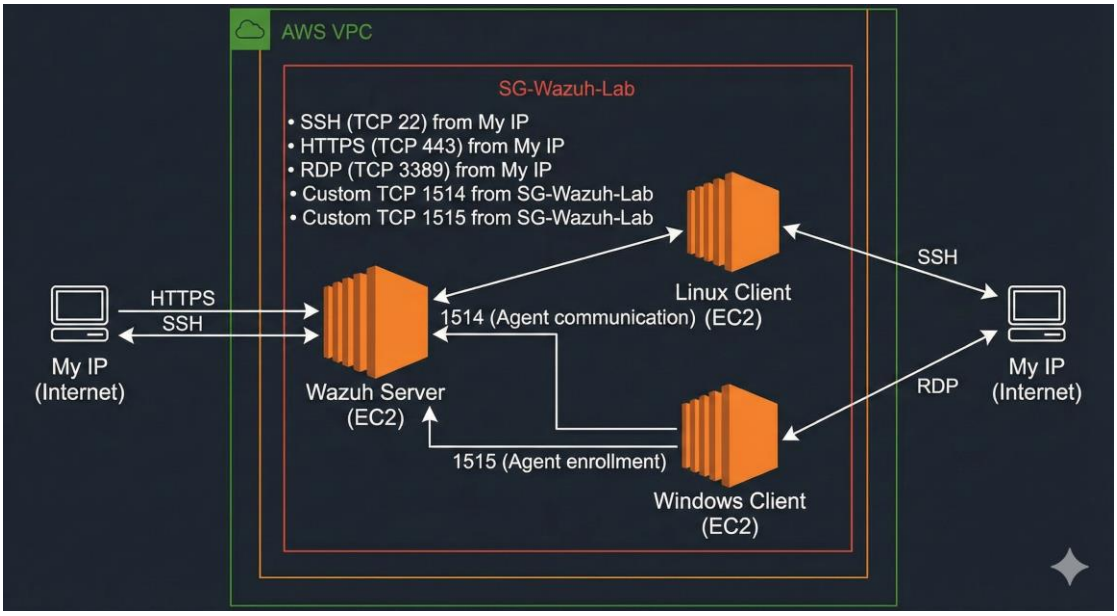

Figure 1 : Schéma de l'architecture VPC (instances + Security Groups)

4. Mise en place de l'infrastructure AWS

4.1. Création des instances EC2

Trois instances EC2 ont été créées dans le même VPC et le même subnet afin de faciliter la communication interne :

- Une instance Ubuntu 24.04 pour le serveur Wazuh
- Une instance Ubuntu 24.04 pour le client Linux

- Une instance Windows Server pour le client Windows

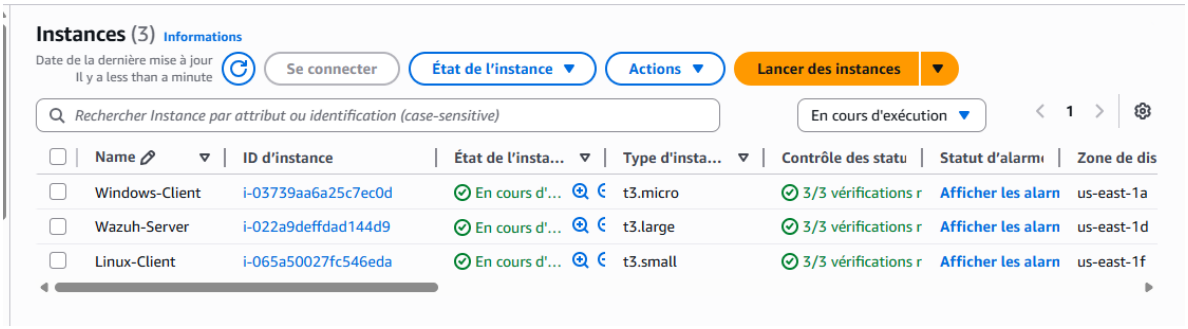

Figure 2 : Vue d'ensemble des trois instances EC2 actives

4.2. Configuration des Security Groups

Les règles de sécurité ont été configurées selon le principe du **moindre privilège** :

- Accès SSH (22/TCP) et HTTPS (443/TCP) limités à l'adresse IP de l'administrateur
- Ports 1514/TCP et 1515/TCP autorisés uniquement depuis le Security Group des clients
- Accès RDP (3389/TCP) restreint à l'IP administrateur

Cette configuration garantit une exposition minimale des services critiques.

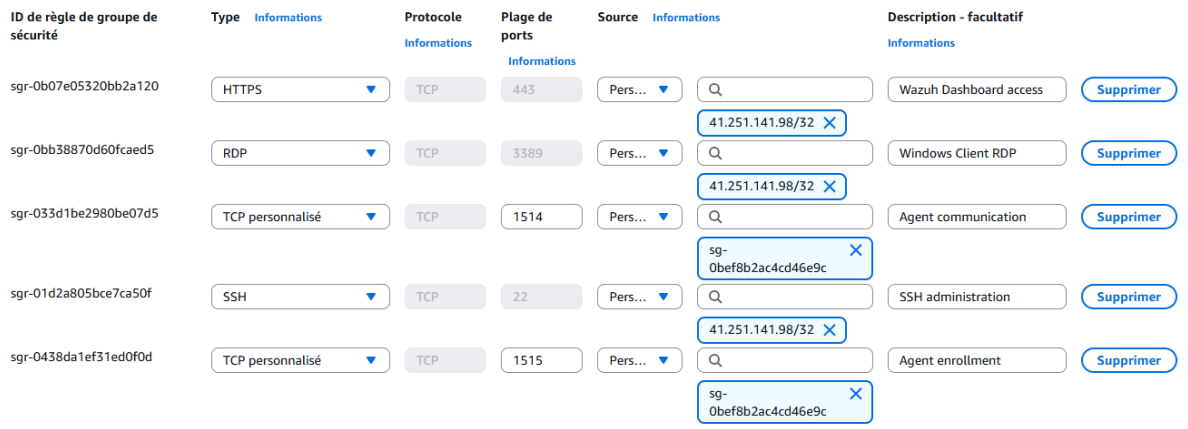

Figure 3 : Règles du Security Group SG-Wazuh-Lab

5. Déploiement de la plateforme Wazuh

5.1. Installation du serveur Wazuh

Le serveur Wazuh a été installé en mode **All-in-One** sur Ubuntu 24.04, intégrant l'ensemble des composants nécessaires (manager, indexer et dashboard). Une fois l'installation terminée, l'accès au dashboard a été vérifié via HTTPS.

```
ubuntu@ip-172-31-40-18:~$ sudo bash wazuh-install.sh -a -i
02/01/2026 13:07:22 INFO: Starting Wazuh installation assistant. Wazuh version: 4.7.5
02/01/2026 13:07:22 INFO: Verbose logging redirected to /var/log/wazuh-install.log
02/01/2026 13:07:30 WARNING: Hardware and system checks ignored.
02/01/2026 13:07:30 INFO: Wazuh web interface port will be 443.
02/01/2026 13:07:35 INFO: --- Dependencies ----
02/01/2026 13:07:35 INFO: Installing apt-transport-https.
02/01/2026 13:07:40 INFO: Wazuh repository added.
02/01/2026 13:07:40 INFO: --- Configuration files ---
02/01/2026 13:07:40 INFO: Generating configuration files.
02/01/2026 13:07:43 INFO: Created wazuh-install-files.tar. It contains the Wazuh cluster key, certificates, and passwords necessary for installation
.
02/01/2026 13:07:43 INFO: --- Wazuh indexer ---
02/01/2026 13:07:43 INFO: Starting Wazuh indexer installation.
02/01/2026 13:08:46 INFO: Wazuh indexer installation finished.
02/01/2026 13:08:46 INFO: Wazuh indexer post-install configuration finished.
02/01/2026 13:08:46 INFO: Starting service wazuh-indexer.
02/01/2026 13:09:07 INFO: wazuh-indexer service started.
02/01/2026 13:09:07 INFO: Initializing Wazuh indexer cluster security settings.
02/01/2026 13:09:18 INFO: Wazuh indexer cluster initialized.
02/01/2026 13:09:18 INFO: --- Wazuh server ---
02/01/2026 13:09:18 INFO: Starting the Wazuh manager installation.
02/01/2026 13:10:12 INFO: Wazuh manager installation finished.
02/01/2026 13:10:12 INFO: Starting service wazuh-manager.
02/01/2026 13:10:29 INFO: wazuh-manager service started.
02/01/2026 13:10:29 INFO: Starting Filebeat installation.
```

Figure 4 : Exécution du script d'installation automatique

```
02/01/2026 13:10:40 INFO: --- Wazuh dashboard ---
02/01/2026 13:10:40 INFO: Starting Wazuh dashboard installation.
02/01/2026 13:11:24 INFO: Wazuh dashboard installation finished.
02/01/2026 13:11:24 INFO: Wazuh dashboard post-install configuration finished.
02/01/2026 13:11:24 INFO: Starting service wazuh-dashboard.
02/01/2026 13:11:25 INFO: wazuh-dashboard service started.
02/01/2026 13:12:00 INFO: Initializing Wazuh dashboard web application.
02/01/2026 13:12:01 INFO: Wazuh dashboard web application initialized.
02/01/2026 13:12:01 INFO: --- Summary ---
02/01/2026 13:12:01 INFO: You can access the web interface https://<wazuh-dashboard-ip>:443
    User: admin
    Password: xN7MMp.CYa6dCopqCOqr?8nrN14Ir*6U
02/01/2026 13:12:01 INFO: Installation finished.
ubuntu@ip-172-31-40-18:~$
```

Figure 5 : Informations de connexion générées par le script

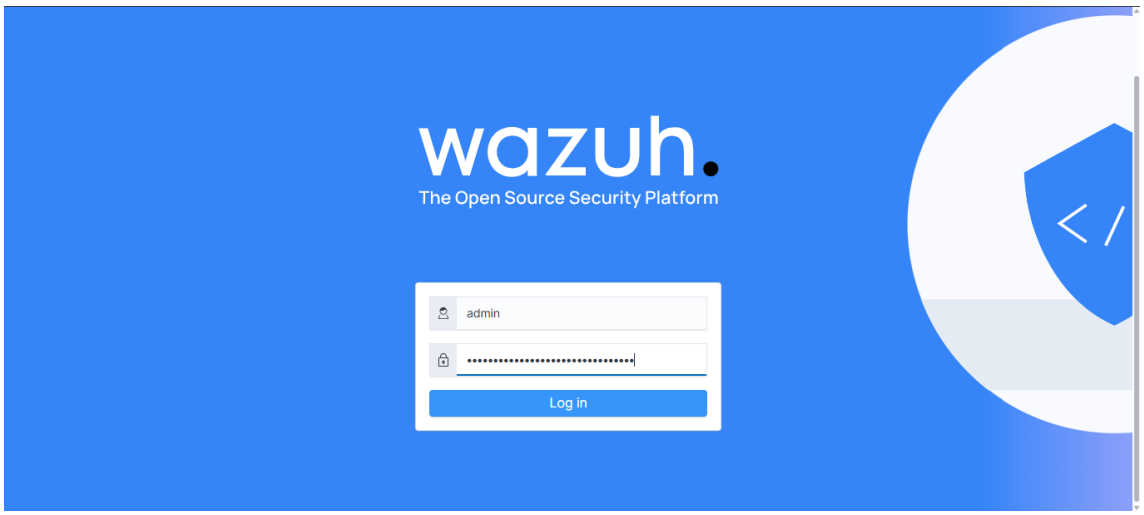

Figure 6 : Interface de connexion au Dashboard Wazuh

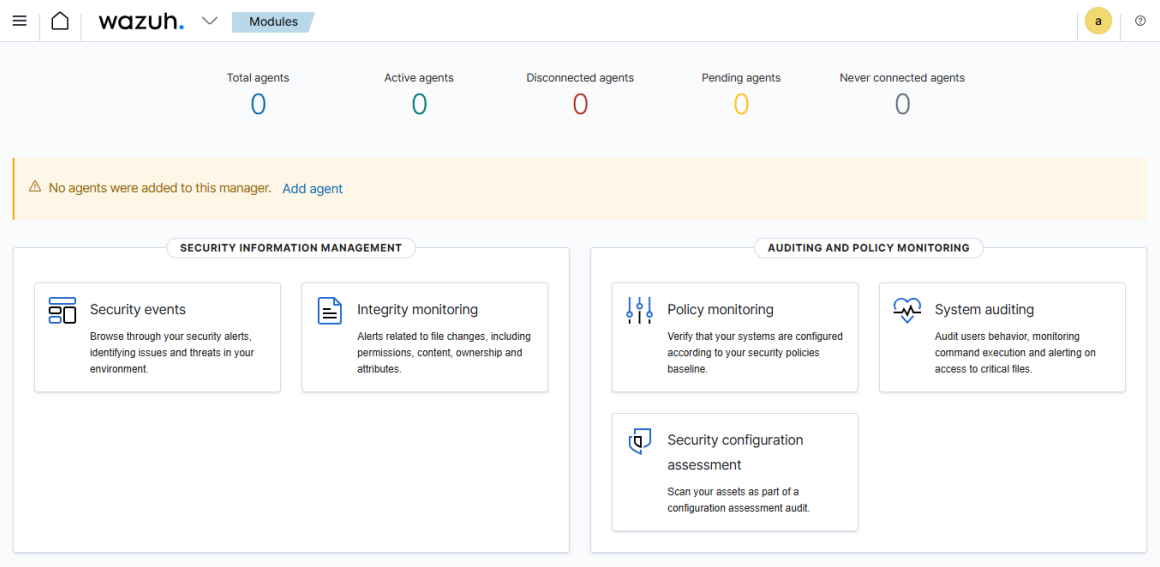

Figure 7 : Vue principale du Dashboard Wazuh après connexion

5.2. Enrôlement des agents

Nous avons déployé les agents Wazuh sur les deux machines clientes afin qu'elles remontent leurs logs vers le serveur manager.

- **Client Linux** : Installation via la commande wget et dpkg fournie par le dashboard.
- **Client Windows** : Installation via PowerShell avec Invoke-WebRequest et démarrage du service WazuhSvc.

Les deux agents sont correctement connectés et apparaissent comme **"Active"** dans le tableau de bord.

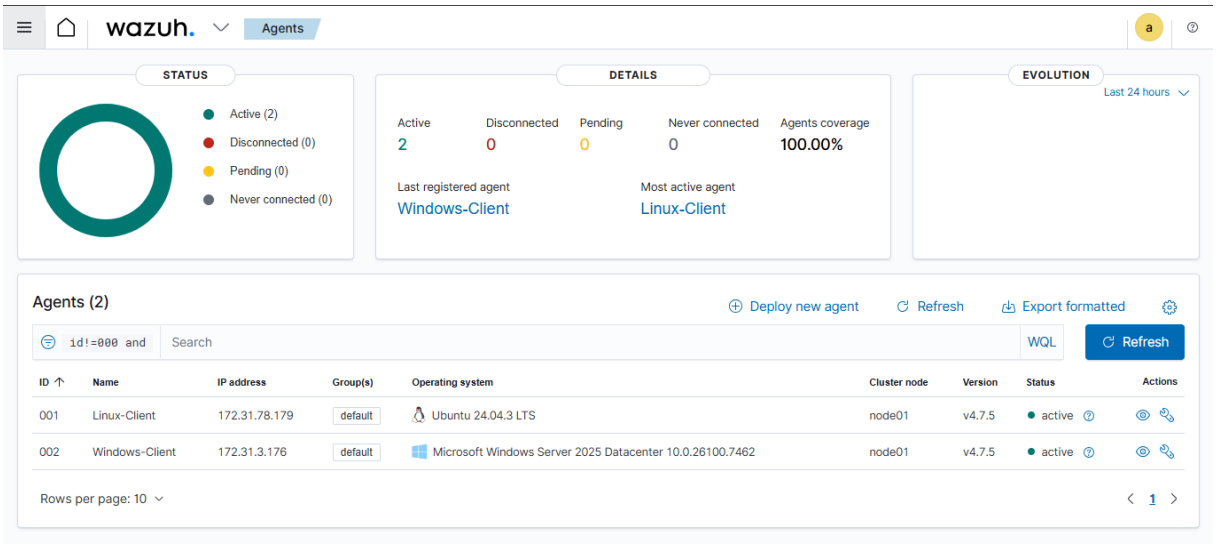

Figure 8 : Liste des agents actifs (Linux et Windows)

6. Démonstration SIEM et EDR

6.1. Scénarios de sécurité côté Linux (SIEM)

Scénario 1 : Attaque par Force Brute SSH

Nous avons simulé une attaque par force brute sur le service SSH du client Linux en utilisant un utilisateur inexistant (fakeuser).

Observations : Wazuh a corrélé les échecs d'authentification et généré une alerte de sécurité de niveau élevé :

Règle ID 5710 : "sshd : Attempt to login using a non-existent user".

```
PS C:\Users\pc\OneDrive\Desktop> ssh fakeuser@3.238.182.238
The authenticity of host '3.238.182.238 (3.238.182.238)' can't be established.
ED25519 key fingerprint is SHA256:E4TY7jr+HwyBCr7OeNK90V3iuisiRlz881OsFYm/WQE.
This host key is known by the following other names/addresses:
    C:\Users\pc/.ssh/known_hosts:37: ec2-3-238-182-238.compute-1.amazonaws.com
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '3.238.182.238' (ED25519) to the list of known hosts.
fakeuser@3.238.182.238: Permission denied (publickey).
PS C:\Users\pc\OneDrive\Desktop> ssh fakeuser@3.238.182.238
fakeuser@3.238.182.238: Permission denied (publickey).
PS C:\Users\pc\OneDrive\Desktop> ssh fakeuser@3.238.182.238
fakeuser@3.238.182.238: Permission denied (publickey).
PS C:\Users\pc\OneDrive\Desktop> ssh fakeuser@3.238.182.238
fakeuser@3.238.182.238: Permission denied (publickey).
PS C:\Users\pc\OneDrive\Desktop> ssh fakeuser@3.238.182.238
fakeuser@3.238.182.238: Permission denied (publickey).
PS C:\Users\pc\OneDrive\Desktop> ssh fakeuser@3.238.182.238
fakeuser@3.238.182.238: Permission denied (publickey).
PS C:\Users\pc\OneDrive\Desktop> ssh fakeuser@3.238.182.238
fakeuser@3.238.182.238: Permission denied (publickey).
PS C:\Users\pc\OneDrive\Desktop> ssh fakeuser@3.238.182.238
fakeuser@3.238.182.238: Permission denied (publickey).
PS C:\Users\pc\OneDrive\Desktop> ssh fakeuser@3.238.182.238
fakeuser@3.238.182.238: Permission denied (publickey).
PS C:\Users\pc\OneDrive\Desktop> ssh fakeuser@3.238.182.238
fakeuser@3.238.182.238: Permission denied (publickey).
PS C:\Users\pc\OneDrive\Desktop> |
```

Figure 9 : Tentatives de connexion SSH échouées sur le client Linux

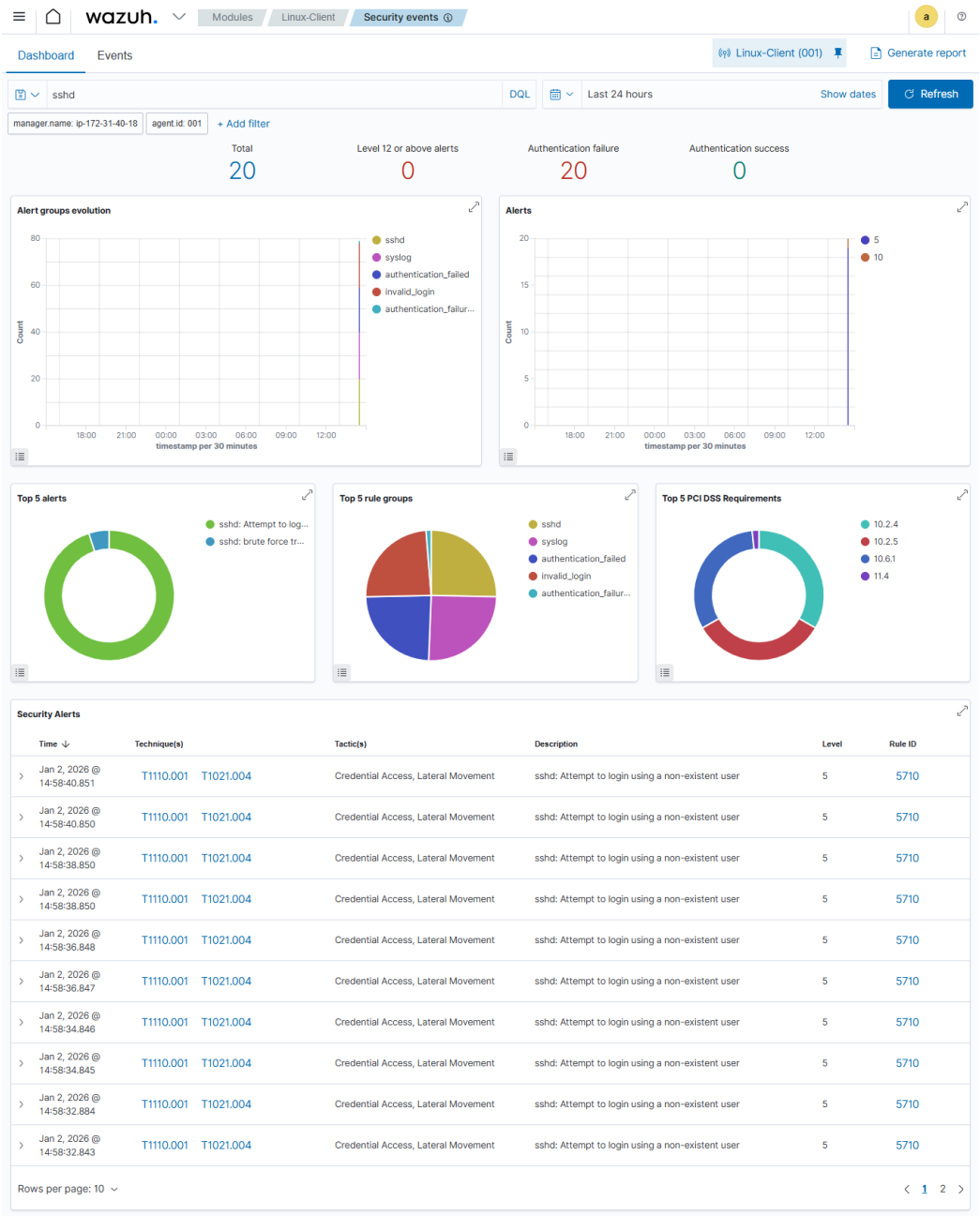

Figure 10 : Alertes de sécurité Wazuh pour l'agent Linux (Bruteforce T1110)

Scénario 2 : Élévation de privilèges (Sudo)

Ensuite, nous avons effectué une élévation de privilèges légitime mais critique via la commande sudo su pour passer en root.

Observations : Wazuh a détecté l'ouverture de session privilégiée via PAM (Pluggable Authentication Modules) :

Règle ID 5501 : "PAM : Login session opened".

Technique MITRE : T1078 (Valid Accounts).

Ce type d'événement est essentiel pour le SOC afin de surveiller l'activité des administra- teurs.

```
ubuntu@ip-172-31-78-179:~$ sudo su
root@ip-172-31-78-179:/home/ubuntu# whoami
root
root@ip-172-31-78-179:/home/ubuntu# exit
exit
ubuntu@ip-172-31-78-179:~$
```

Figure 11 : Élévation de privilèges via sudo

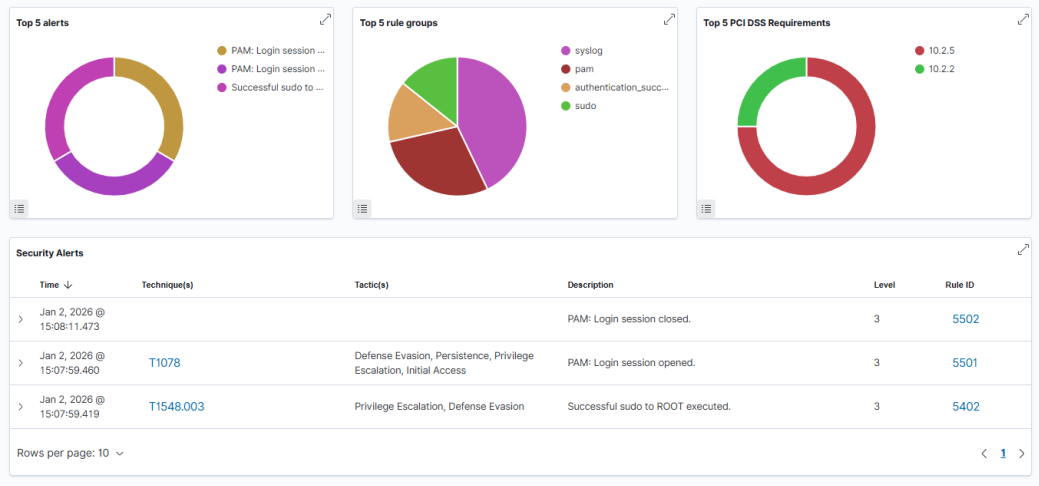

Figure 12 : Détection de l'élévation de privilèges via sudo

6.2. Scénarios de sécurité côté Windows (EDR)

Scénario 1 : Échecs de connexion RDP

Le protocole RDP (Remote Desktop Protocol) est une cible privilégiée des attaquants pour accéder à distance aux systèmes Windows. La détection de tentatives de connexion échouées est cruciale.

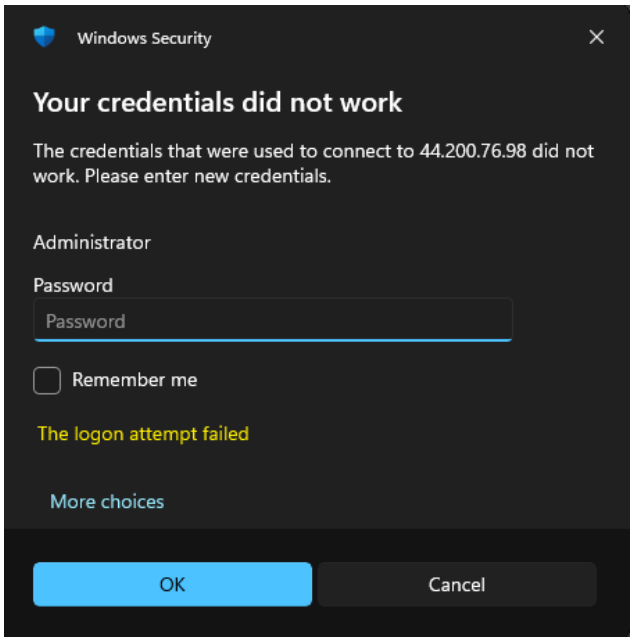

Figure 13 : Échecs répétés d'authentification RDP

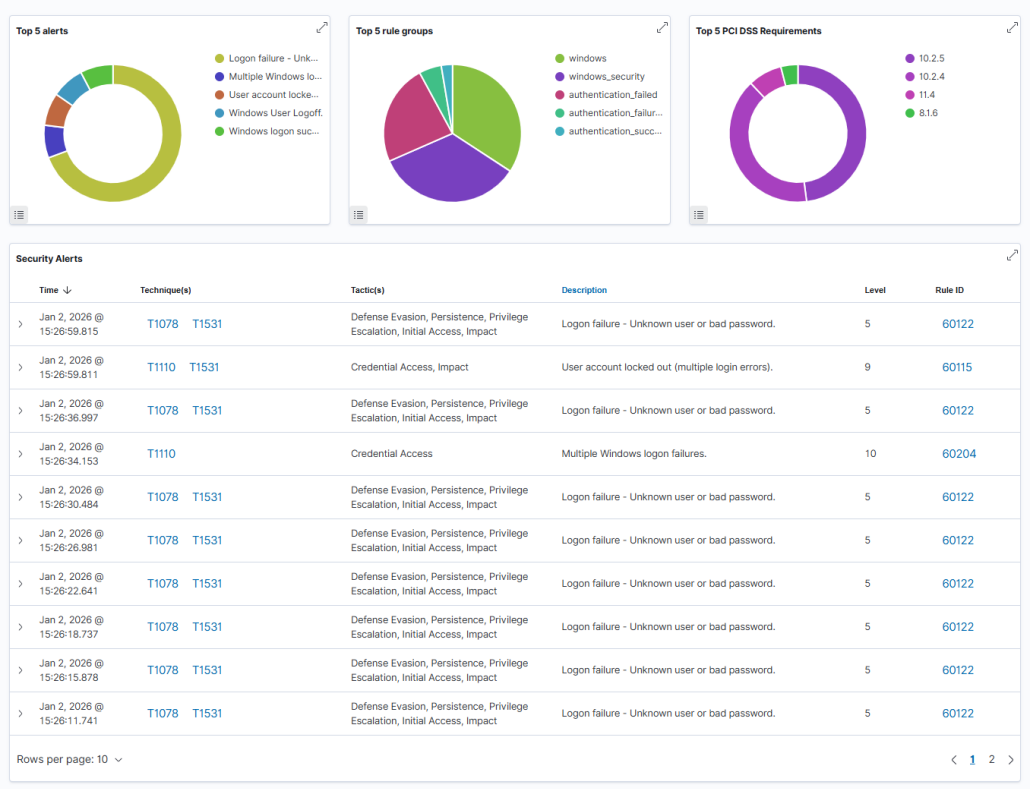

Figure 14 : Détection des tentatives de connexion RDP échouées

Scénario 2 : Création d'utilisateur et modification de groupes

La création d'utilisateurs locaux et leur ajout à des groupes privilégiés (Administrators) sont des indicateurs critiques pouvant signaler :

- Une compromission du système

- Une tentative de persistance par un attaquant
- Une création de backdoor

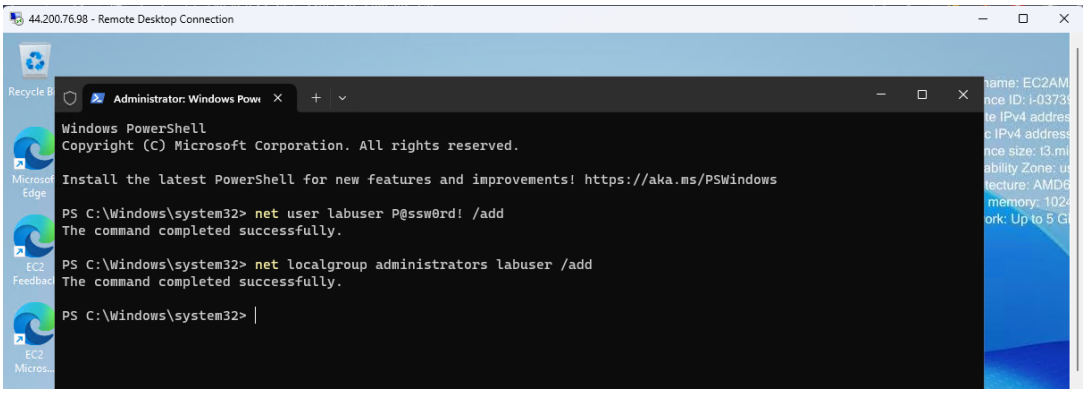

Figure 15 : Commandes PowerShell pour créer un utilisateur privilégié

›	Jan 2, 2026 @ 15:40:27.918	T1484	Defense Evasion, Privilege Escalation	Administrators group changed.	12	60154
›	Jan 2, 2026 @ 15:40:21.617	T1484	Defense Evasion, Privilege Escalation	Users group changed.	5	60170
›	Jan 2, 2026 @ 15:40:21.607	T1098	Persistence	User account changed.	8	60110
›	Jan 2, 2026 @ 15:40:21.593	T1098	Persistence	User account enabled or created.	8	60109
›	Jan 2, 2026 @ 15:40:21.583	T1098	Persistence	User account enabled or created.	8	60109
›	Jan 2, 2026 @ 15:40:21.535	T1484	Defense Evasion, Privilege Escalation	Domain users group changed.	5	60160

Figure 16 : Détection de la création d'utilisateur et modification de groupe

Scénario 3 (Optionnel) : Monitoring avancé avec Sysmon

Sysmon (System Monitor) est un outil Microsoft Sysinternals qui fournit une télémétrie détaillée sur l'activité système :

- Création de processus
- Connexions réseau
- Modifications de fichiers
- Chargement de DLL
- Activité registre

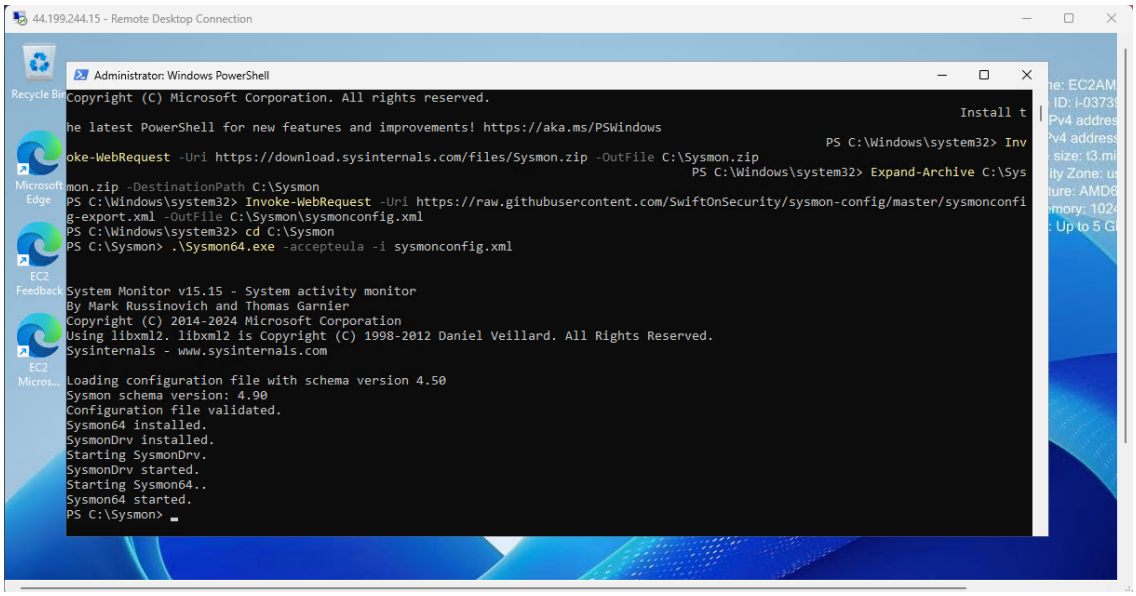

Figure 17 : Sysmon installé et configuré avec succès

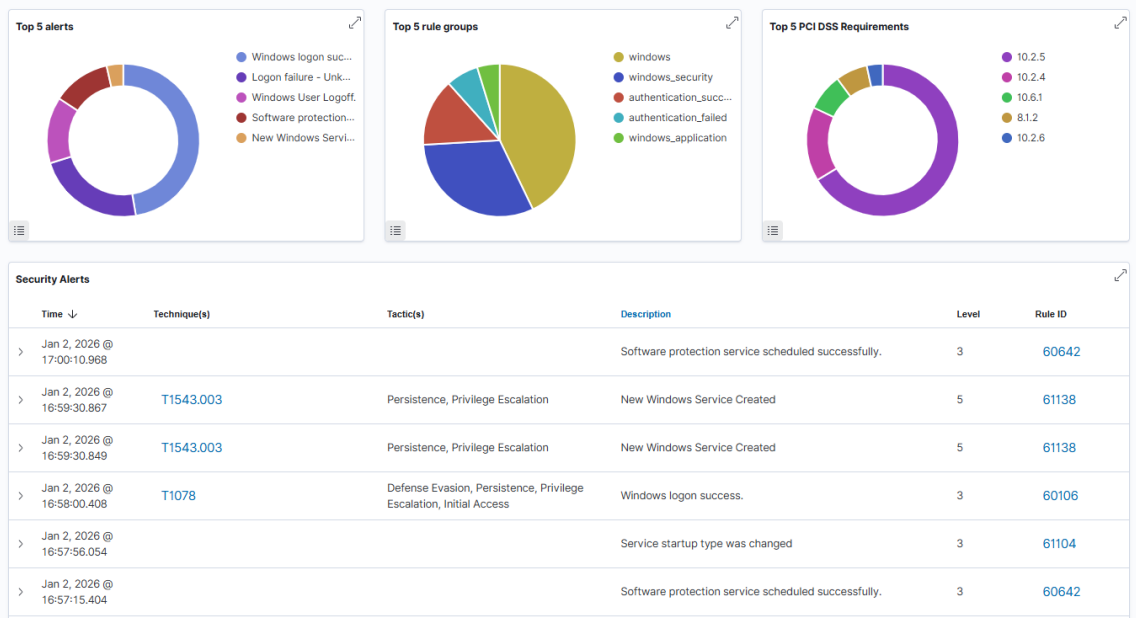

Figure 18 : Détection de création de processus via Sysmon

7. Analyse : SIEM, EDR et IAM

7.1. SIEM vs EDR

- **SIEM** : centralisation des logs, corrélation d'événements, visibilité globale
- **EDR** : surveillance fine des endpoints, détection comportementale et réponse

Wazuh permet de combiner efficacement ces deux approches.

7.2. IAM / PAM

Les événements liés à l'authentification, aux échecs de connexion et aux changements de privilèges illustrent l'importance de la gestion des identités et des accès (IAM/PAM) dans un SOC.

8. Initiation au Threat Hunting

Trois exemples de requêtes de chasse aux menaces ont été étudiés :

- Détection de brute force SSH
- Analyse des échecs d'authentification Windows
- Surveillance des créations d'utilisateurs et des élévations de privilèges

Ces requêtes permettent d'identifier des comportements suspects avant qu'un incident majeur ne survienne.

9. Conclusion

Cet atelier a permis de mettre en pratique les concepts fondamentaux d'un **SOC moderne**, en s'appuyant sur une architecture Cloud réaliste et une solution open-source robuste. La combinaison SIEM et EDR offerte par Wazuh constitue une base solide pour la détection et l'analyse des menaces de sécurité dans un environnement hybride Linux/Windows.

L'ensemble des objectifs pédagogiques a été atteint, et la plateforme déployée peut servir de fondation pour des travaux plus avancés en cybersécurité.

Lien GitHub :

Le dépôt GitHub du projet, contenant l'ensemble des ressources et de la documentation, est disponible à l'adresse suivante :

https://github.com/Marouanemounir/wazuh-siem-edr-lab.