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

Departement Mathematique Informatique

Filière d'ingenieur ***: GLSID***

*Déploiement d'une plateforme **SIEM** & **EDR** basée sur **Wazuh** dans un environnement **Cloud AWS***

Réalisé par :

RIANI Aymen

Encadré par :

Pr. Azeddine KHIAT

Module : **Virtualisation et Cloud computing**

Année universitaire 2025–2026

Introduction :

Avec l'essor du **Cloud Computing** et de la **virtualisation**, les systèmes d'information modernes sont devenus plus flexibles, évolutifs et accessibles. Cependant, cette évolution s'accompagne de nouveaux défis, notamment en matière de **sécurité**, car les infrastructures Cloud sont exposées à des menaces de plus en plus sophistiquées. La supervision, la détection et la réponse aux incidents de sécurité constituent aujourd'hui des enjeux majeurs pour les entreprises.

Dans ce contexte, les centres opérationnels de sécurité (**SOC – Security Operations Center**) s'appuient sur des solutions avancées telles que les **SIEM** (Security Information and Event Management) et les **EDR** (Endpoint Detection and Response) afin de collecter, corréler et analyser les événements de sécurité provenant de différents systèmes. Ces solutions permettent non seulement de détecter les attaques, mais aussi de mieux comprendre les comportements anormaux et de réagir efficacement face aux incidents.

Le présent projet s'inscrit dans le cadre du module Virtualisation et Cloud Computing et vise à mettre en pratique ces concepts à travers le déploiement d'une plateforme de supervision de la sécurité basée sur **Wazuh**. L'infrastructure est hébergée dans un environnement **Cloud AWS**, utilisant des machines virtuelles **EC2**, et couvre un scénario multi-systèmes incluant un serveur de supervision, un client Linux et un client Windows. Cette approche permet de démontrer la surveillance de différents types d'endpoints au sein d'une même architecture Cloud.

L'objectif principal de ce projet est de concevoir et déployer une solution complète intégrant les fonctions **SIEM** et **EDR**, capable de collecter des logs, de détecter des événements de sécurité et d'analyser des incidents sur des systèmes Linux et Windows. À travers plusieurs scénarios de démonstration, le projet met en évidence des cas concrets tels que les tentatives d'authentification échouées, l'élévation de privilèges, la création d'utilisateurs et la modification de fichiers sensibles.

Ce rapport présente l'ensemble des étapes réalisées, depuis la conception de l'architecture Cloud jusqu'à l'analyse des événements de sécurité via le tableau de bord Wazuh. Il met également en avant l'intérêt pédagogique et professionnel de la solution déployée, en soulignant le rôle essentiel de la virtualisation, du Cloud Computing et des outils de sécurité modernes dans la protection des systèmes d'information.

Chapitre 1 : Concepts théoriques

I. Virtualisation et Cloud Computing

La **virtualisation** est une technologie clé qui permet de faire fonctionner plusieurs systèmes d'exploitation et applications sur une même machine physique grâce à des machines virtuelles. Elle optimise l'utilisation des ressources matérielles, améliore la flexibilité et facilite la gestion des infrastructures informatiques.

Le **Cloud Computing** repose sur ce principe de virtualisation et permet de fournir des ressources informatiques (serveurs, stockage, réseau, applications) à la demande via Internet. Il offre plusieurs avantages majeurs tels que la scalabilité, la haute disponibilité, la réduction des coûts et la rapidité de déploiement. Dans ce projet, le Cloud AWS est utilisé pour héberger l'ensemble de l'infrastructure à travers des instances EC2, illustrant concrètement le modèle **Infrastructure as a Service (IaaS)**.

II. SIEM (Security Information and Event Management)

Un **SIEM** est une solution de sécurité qui permet de collecter, centraliser, corréler et analyser les journaux d'événements provenant de différents systèmes informatiques (serveurs, applications, équipements réseau, systèmes d'exploitation). L'objectif principal d'un SIEM est de détecter rapidement les incidents de sécurité et de fournir une vision globale de l'état de sécurité du système d'information.

Les fonctionnalités principales d'un SIEM incluent :

- La collecte centralisée des logs
- La corrélation des événements
- La génération d'alertes de sécurité
- La visualisation via des tableaux de bord
- Le support aux investigations de sécurité

Dans ce projet, **Wazuh** joue le rôle de SIEM en collectant les événements générés par les systèmes Linux et Windows, puis en les analysant afin d'identifier des comportements suspects ou malveillants.

III. 1.3 EDR (Endpoint Detection and Response)

L'**EDR** est une solution de sécurité focalisée sur la surveillance et la protection des endpoints (postes de travail, serveurs). Contrairement au SIEM qui offre une vision globale, l'EDR analyse en profondeur les activités locales d'un système afin de détecter des comportements anormaux tels que l'exécution de processus suspects, les accès non autorisés ou les modifications système

critiques.

Les solutions EDR permettent :

- La surveillance en temps réel des endpoints
- La détection avancée des menaces
- L'analyse comportementale
- La réponse aux incidents (isolation, investigation)

Dans le cadre de ce projet, Wazuh intègre également des fonctionnalités EDR grâce à ses agents installés sur les machines Linux et Windows, permettant ainsi une surveillance fine des activités sur chaque endpoint.

IV. Différences et complémentarité entre SIEM et EDR

Bien que le SIEM et l'EDR aient des objectifs similaires en matière de sécurité, ils se distinguent par leur approche :

- Le **SIEM** offre une vue centralisée et globale du système d'information.
- L'**EDR** se concentre sur la surveillance détaillée des endpoints.

Ces deux solutions sont complémentaires et sont généralement utilisées conjointement dans un **SOC moderne**. Dans ce projet, l'intégration des fonctions SIEM et EDR à travers Wazuh permet une détection plus efficace des menaces, aussi bien au niveau global qu'au niveau local des systèmes.

V. IAM et PAM dans la sécurité des systèmes

La gestion des identités et des accès (**IAM – Identity and Access Management**) est un élément essentiel de la sécurité informatique. Elle permet de contrôler qui peut accéder aux ressources et avec quels privilèges. Le **PAM (Privileged Access Management)**, quant à lui, se concentre sur la gestion des comptes à privilèges élevés.

Les événements liés à l'IAM et au PAM, tels que les échecs d'authentification, les créations d'utilisateurs ou les élévations de privilèges, sont des indicateurs importants en matière de sécurité. Dans ce projet, ces événements sont surveillés à l'aide de Wazuh afin de détecter toute activité suspecte ou non autorisée sur les systèmes Linux et Windows.

Conclusion du chapitre

Ce chapitre a permis de présenter les concepts nécessaires à la compréhension du projet, notamment la virtualisation, le Cloud Computing, le SIEM, l'EDR ainsi que la gestion des identités et des accès. Ces notions constituent la base théorique sur laquelle repose la mise en œuvre pratique de la plateforme de supervision de la sécurité présentée parsuite.

Chapitre 2 : Architecture du projet

I. Présentation générale de l'architecture

L'architecture du projet repose sur une infrastructure **Cloud virtualisée** déployée sur la plateforme **AWS Learner Lab**. Elle a été conçue dans le but de mettre en place une solution centralisée de supervision de la sécurité intégrant les fonctions **SIEM et EDR** à l'aide de la solution **Wazuh**.

L'environnement est volontairement simple, sécurisé et représentatif d'un contexte réel d'entreprise. Il repose sur une architecture **multi-systèmes**, incluant un serveur de supervision et deux endpoints de nature différente (Linux et Windows), permettant ainsi de démontrer la collecte et l'analyse d'événements de sécurité hétérogènes.

II. Architecture Cloud AWS

L'infrastructure Cloud est hébergée au sein d'un **VPC (Virtual Private Cloud)** unique, garantissant l'isolation réseau des ressources du projet. Toutes les instances sont déployées dans un **même subnet**, facilitant la communication interne tout en restant sécurisée grâce à l'utilisation de **Security Groups**.

L'architecture repose sur trois instances **EC2** :

- Une instance dédiée au **serveur Wazuh**
- Une instance **client Linux**
- Une instance **client Windows**

Ce choix permet de simuler un environnement Cloud minimaliste mais réaliste, intégrant à la fois la supervision centralisée et la surveillance des endpoints.

III. Description des composants

i. Serveur Wazuh (SIEM / EDR central)

Le serveur Wazuh constitue le cœur de l'architecture. Il est déployé sur une instance EC2 exécutant **Ubuntu 22.04 LTS** et intègre l'ensemble des composants Wazuh dans une installation *All-in-One* :

- **Wazuh Manager** : collecte et corrélation des événements
- **Wazuh Indexer** : stockage et indexation des données
- **Wazuh Dashboard** : visualisation et analyse des alertes

Ce serveur centralise les journaux de sécurité provenant des endpoints et permet leur analyse via une interface web sécurisée.

ii. Client Linux

Le client Linux est également déployé sur une instance EC2 sous **Ubuntu 22.04**. Il représente un serveur Linux classique au sein d'un système d'information.

Un agent Wazuh y est installé afin de collecter différents types d'événements, notamment :

- Tentatives d'authentification SSH
- Élévations de privilèges (sudo)
- Modifications de fichiers sensibles (FIM)

Ces événements sont ensuite transmis au serveur Wazuh pour analyse.

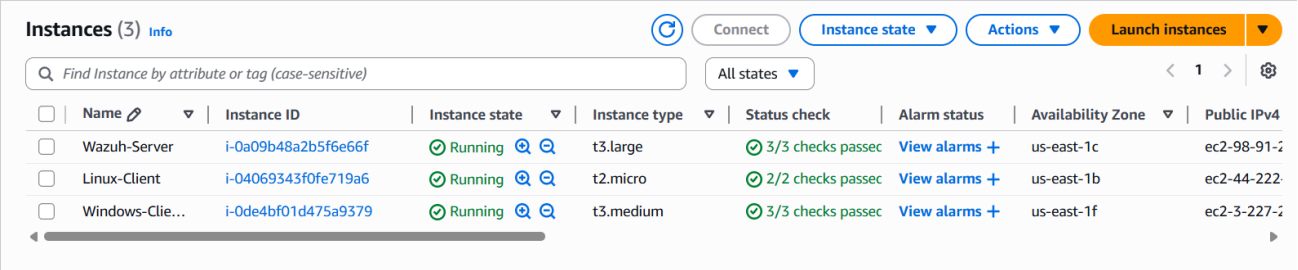
iii. Client Windows

Le client Windows est déployé sur une instance EC2 exécutant **Windows Server**. Il représente un environnement Windows typique en entreprise.

L'agent Wazuh installé sur ce système permet de surveiller :

- Les événements de sécurité Windows
- Les connexions échouées
- La gestion des utilisateurs et des groupes
- Les activités liées aux privilèges

Ce composant est essentiel pour démontrer les capacités **EDR** de la solution dans un environnement Windows.



IV. Sécurité réseau et flux de communication

La sécurité réseau est assurée par l'utilisation de **Security Groups**, appliquant le principe du **moindre privilège**. Seuls les ports strictement nécessaires au fonctionnement de la solution sont ouverts.

Les principaux flux de communication sont les suivants :

- Les agents Linux et Windows communiquent avec le serveur Wazuh via :
  - Le port **1514/TCP** pour l'envoi des événements
  - Le port **1515/TCP** pour l'enrôlement des agents
- L'accès au tableau de bord Wazuh se fait via le protocole **HTTPS**
- Les accès administratifs sont limités :
  - SSH pour les machines Linux

- RDP pour la machine Windows

Aucun endpoint n'est exposé directement à Internet pour des services inutiles, ce qui renforce la sécurité globale de l'architecture.

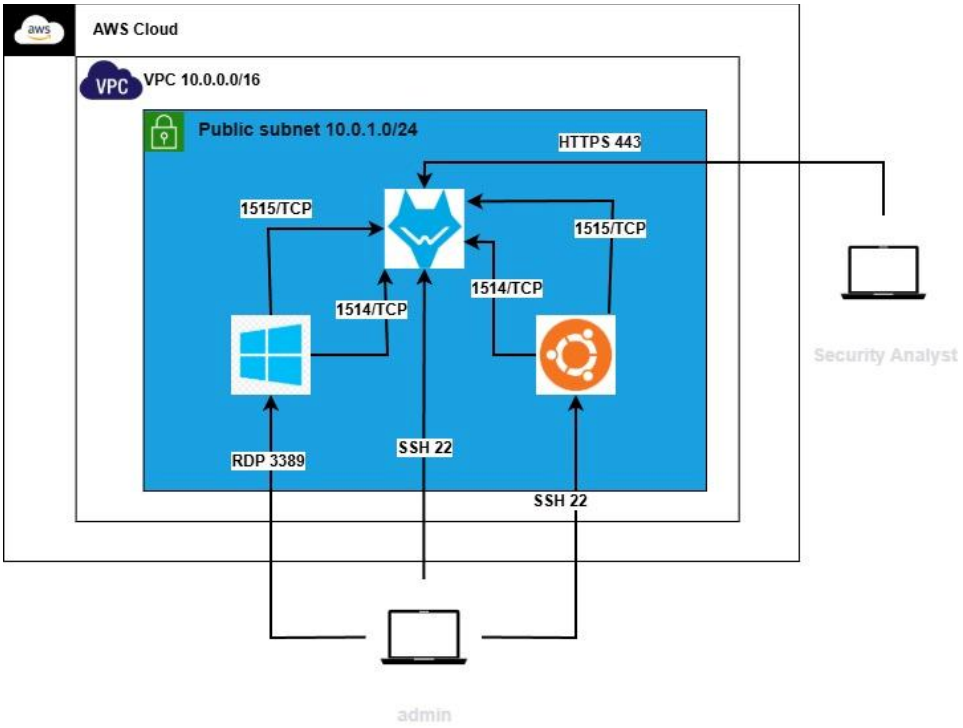
V. Schéma d'architecture

Un schéma d'architecture a été réalisé afin d'illustrer la structure du projet, les composants Cloud, ainsi que les flux de communication entre les différentes instances.

Ce schéma met en évidence :

- Le VPC et le subnet
- Les instances EC2
- Les ports utilisés
- Les flux agents → serveur Wazuh

Le schéma est présenté dans la figure ci-dessous



Conclusion du chapitre

Ce chapitre a présenté l'architecture globale du projet, en détaillant l'infrastructure Cloud AWS, les différents composants et les mécanismes de sécurité réseau mis en place. Cette architecture constitue une base solide pour la mise en œuvre de la plateforme Wazuh et permet de démontrer de manière concrète l'intégration des concepts de virtualisation, de Cloud Computing et de sécurité des systèmes.

Chapitre 3 : Mise en œuvre technique

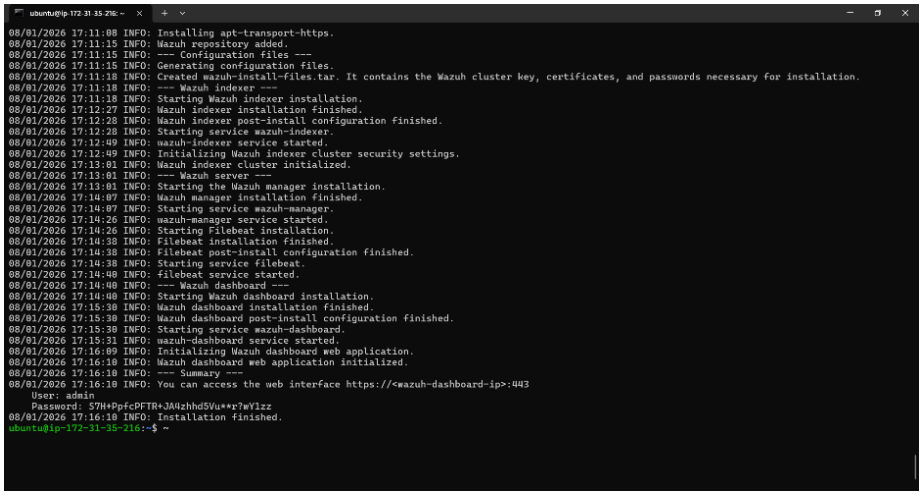
I. Déploiement de l'infrastructure Cloud

L'infrastructure du projet a été déployée sur la plateforme **AWS Learner Lab** en utilisant des machines virtuelles **EC2**. Un **VPC unique** a été configuré afin d'assurer l'isolation réseau des ressources. Les instances ont été placées dans un même subnet et protégées par des **Security Groups** permettant de contrôler précisément les flux entrants et sortants.

Trois instances EC2 ont été créées :

- Une instance dédiée au **serveur Wazuh**
- Une instance **client Linux**
- Une instance **client Windows**

Chaque instance a été dimensionnée en fonction de son rôle, en tenant compte des besoins en ressources et des bonnes pratiques Cloud.



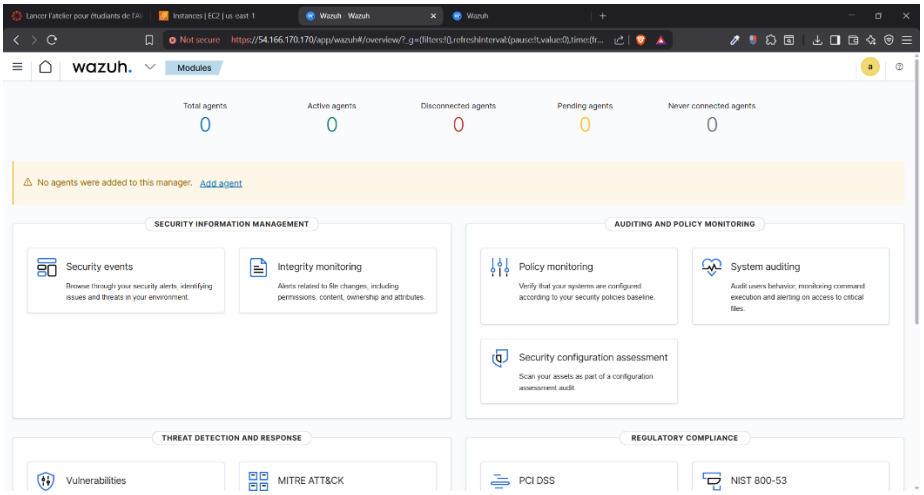
II. Installation du serveur Wazuh

Le serveur Wazuh a été installé sur une instance EC2 exécutant **Ubuntu 22.04 LTS**. Après la mise à jour du système, l'installation a été réalisée à l'aide du script officiel **Wazuh All-in-One**, permettant de déployer l'ensemble des composants nécessaires au fonctionnement de la plateforme.

Cette installation inclut :

- Le **Wazuh Manager** pour la gestion et l'analyse des événements
- Le **Wazuh Indexer** pour le stockage des données
- Le **Wazuh Dashboard** pour la visualisation des alertes

Une fois l'installation terminée, les services ont été vérifiés afin de s'assurer du bon fonctionnement de la plateforme.



III. Enrôlement du client Linux

Le client Linux, exécutant **Ubuntu 22.04**, a été configuré pour être supervisé par la plateforme Wazuh. L'enrôlement de l'agent a été effectué via le **tableau de bord Wazuh**, en utilisant la fonctionnalité de déploiement automatique des agents.

Une fois l'agent installé et configuré, la communication avec le serveur Wazuh a été vérifiée. Le client Linux apparaît alors comme **actif** dans le tableau de bord, confirmant la bonne remontée des événements de sécurité.

IV. Enrôlement du client Windows

Le client Windows a été intégré à la plateforme Wazuh à l'aide de l'agent Wazuh pour Windows. L'installation a été réalisée via les commandes fournies par le tableau de bord Wazuh, puis exécutées sur le système Windows à l'aide de PowerShell.

Après l'installation, le service **Wazuh Agent** a été vérifié afin de s'assurer qu'il était bien en cours d'exécution. Le client Windows est ensuite apparu comme **actif** dans le tableau de bord Wazuh, permettant la collecte et l'analyse des événements de sécurité Windows.

V. Configuration des communications et de la sécurité

La communication entre les agents et le serveur Wazuh repose sur des ports dédiés et sécurisés. Les **Security Groups AWS** ont été configurés de manière à autoriser uniquement les flux nécessaires au fonctionnement de la solution :

- Port **1514/TCP** pour la transmission des événements de sécurité
- Port **1515/TCP** pour l'enrôlement automatique des agents
- Port **HTTPS** pour l'accès au tableau de bord
- Accès SSH et RDP limités à l'adresse IP de l'administrateur

Cette configuration garantit une communication fiable tout en respectant les bonnes pratiques de sécurité Cloud.

Conclusion du chapitre

Ce chapitre a présenté l'ensemble des étapes techniques réalisées pour déployer l'infrastructure Cloud et mettre en place la plateforme Wazuh. L'installation du serveur, l'enrôlement des agents Linux et Windows ainsi que la configuration des communications ont permis d'obtenir un environnement opérationnel, prêt à être utilisé pour les démonstrations SIEM et EDR présentées dans le chapitre suivant.

Chapitre 4 : Démonstrations SIEM & EDR

I. Objectif des démonstrations

Ce chapitre présente plusieurs scénarios pratiques visant à **illustrer le fonctionnement de la plateforme Wazuh** et à démontrer :

- La collecte d'événements de sécurité depuis différents systèmes d'exploitation
- La détection d'incidents via le SIEM
- La surveillance des endpoints via les fonctions EDR

Ces démonstrations permettent de valider la configuration de l'infrastructure et l'efficacité des agents Wazuh sur les systèmes Linux et Windows.

II. Démonstrations sur le client Linux

Scénario 1 : Tentatives SSH échouées (Brute force simulé)

- Plusieurs tentatives de connexion SSH avec des identifiants invalides ont été effectuées depuis une autre machine.
- Résultat attendu : alertes **authentication failed** générées par Wazuh.
- Source des événements : /var/log/auth.log

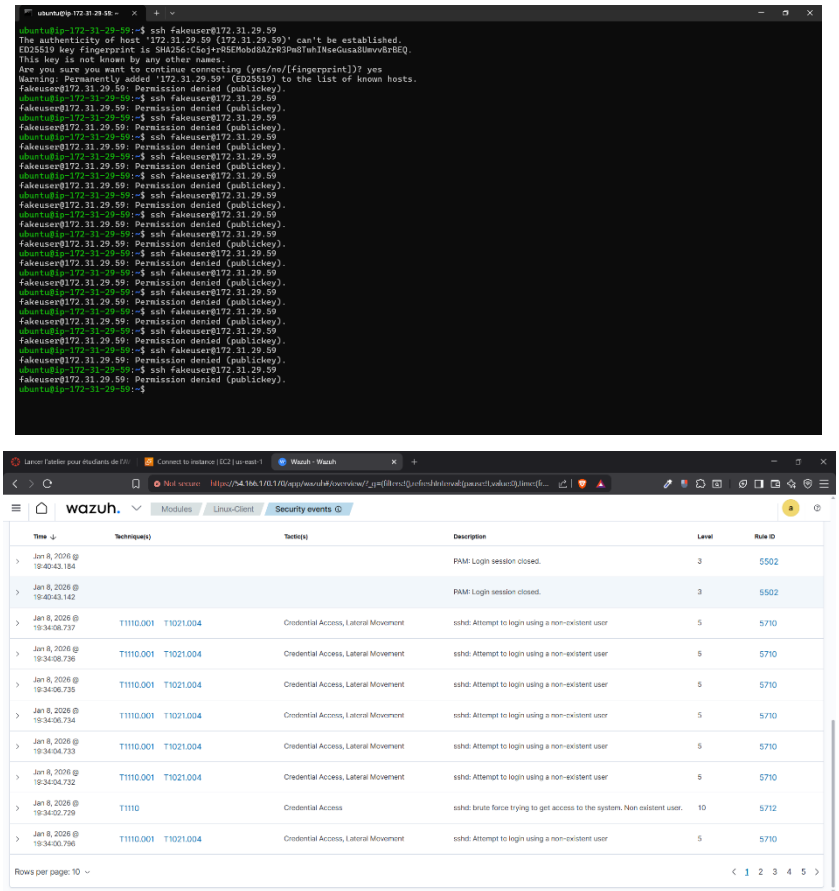


Figure 1 - alerte brute force SSH sur le Dashboard Wazuh

Scénario 2 : Élévation de privilèges

- Commande exécutée :

sudo su

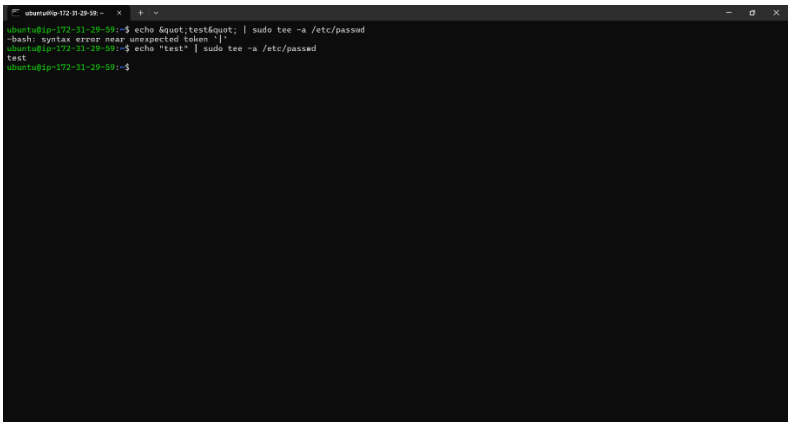
- Résultat attendu : génération d'événements sudo dans Wazuh, indiquant une élévation de privilèges.

Scénario 3 : Modification de fichiers sensibles (FIM)

- Commande exécutée :

echo "test" | sudo tee -a /etc/passwd

- Résultat attendu : alerte **File Integrity Monitoring (FIM)** détectée par Wazuh.



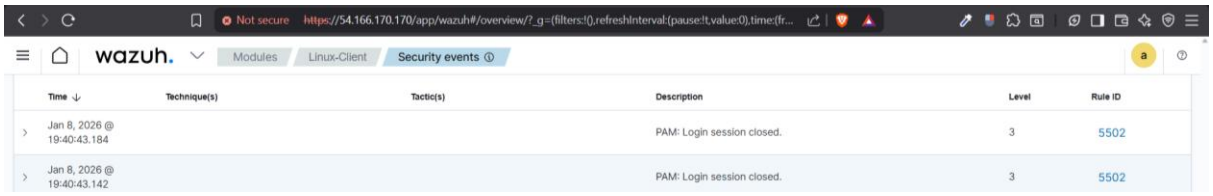
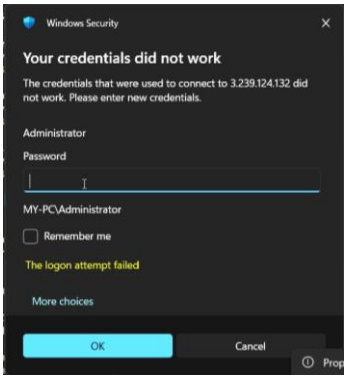


Figure 2 - alerte FIM sur le Dashboard Wazuh

III. Démonstrations sur le client Windows

Scénario 1 : Échecs de connexion RDP

- Plusieurs tentatives de connexion RDP avec de mauvais identifiants ont été effectuées.
- Résultat attendu : événements **Failed Logon (4625)** remontés par Wazuh.



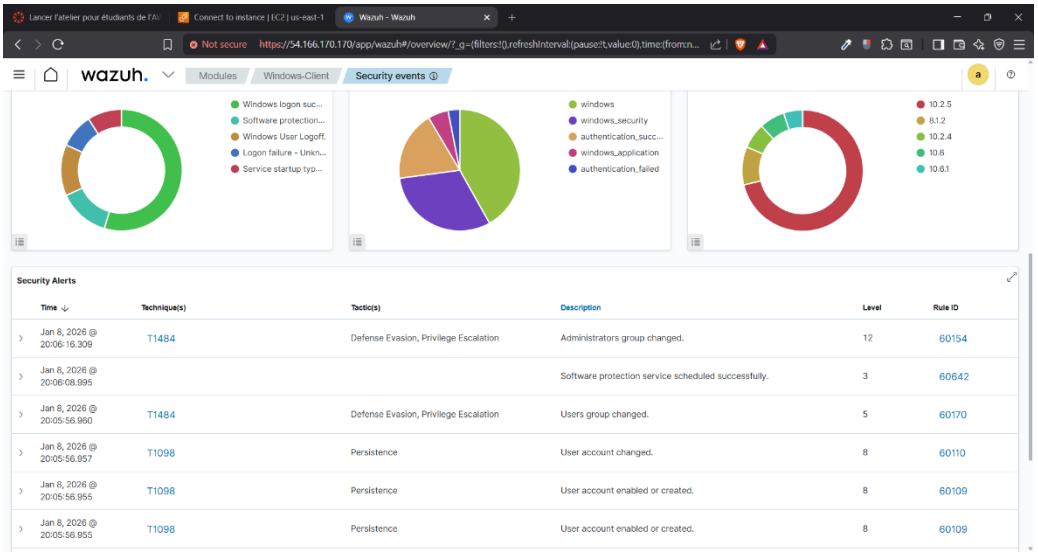
Scénario 2 : Création d'un utilisateur local et modification de groupe

- Commandes PowerShell exécutées :

net user labuser P@ssw0rd! /add

net localgroup administrators labuser /add

- Résultat attendu : alertes sur la création d'utilisateur et la modification de groupe dans Wazuh.



IV. Analyse des résultats

Tous les événements simulés ont été correctement remontés et analysés par le serveur Wazuh.

Les agents Linux et Windows apparaissent comme **actifs** sur le tableau de bord.

Les alertes sont cohérentes avec les scénarios planifiés, ce qui valide la **mise en œuvre technique** et le **fonctionnement de la plateforme SIEM & EDR**.

Conclusion du chapitre

Ce chapitre a démontré concrètement la capacité de la plateforme Wazuh à **collecter et analyser des événements de sécurité sur des systèmes hétérogènes**. Les démonstrations couvrent à la fois des incidents Linux et Windows, mettant en évidence la complémentarité entre **SIEM** (vue centralisée) et **EDR** (surveillance des endpoints). Cette phase valide la configuration de l'infrastructure et constitue une base solide pour l'analyse approfondie présentée dans le chapitre suivant.

Chapitre 5 : Analyse & Threat Hunting

I. Objectifs de l'analyse

Après avoir validé la remontée des alertes lors des démonstrations, ce chapitre se concentre sur l'exploitation intelligente des données collectées. L'objectif est de passer d'une posture réactive (recevoir des alertes) à une posture proactive (chercher des menaces), une pratique essentielle au sein d'un **SOC (Security Operations Center)** moderne.

II. Supervision et gestion des alertes

La plateforme Wazuh classe les événements par niveaux de priorité, permettant aux analystes de se concentrer sur les incidents critiques.

- **Filtrage par niveau :** Nous avons utilisé le Dashboard pour isoler les alertes de niveau élevé, notamment celles liées aux tentatives de brute force et aux modifications système.
- **Analyse multi-agents :** Grâce à la vue centralisée, nous avons pu comparer en temps réel les journaux d'événements provenant simultanément du client Linux et du client Windows.

III. Pratiques de Threat Hunting

Le **Threat Hunting** consiste à interroger la base de données de logs pour identifier des comportements suspects qui n'auraient pas déclenché d'alerte automatique immédiate.

- **Requêtes sur les identités (IAM/PAM) :** Nous avons effectué des recherches ciblées sur les créations de comptes et les changements de groupes d'utilisateurs afin de détecter d'éventuels vecteurs de persistance.
- **Analyse des flux réseau :** En filtrant les communications via les ports spécifiques (1514/1515), nous avons vérifié l'intégrité de la chaîne de remontée des logs entre les agents et le manager.
- **Investigation sur les fichiers (FIM) :** L'analyse approfondie des alertes de monitoring d'intégrité a permis de retracer l'heure exacte et l'utilisateur à l'origine de modifications sur des fichiers sensibles comme /etc/passwd.

Conclusion du chapitre

Ce chapitre a démontré que la plateforme déployée n'est pas seulement un outil de stockage de logs, mais un véritable instrument d'investigation. La combinaison des filtres de recherche de Wazuh et de la précision de Sysmon offre une visibilité complète sur l'état de sécurité de l'infrastructure Cloud AWS.

Chapitre 6 : Valorisation du projet

I. Dépôt GitHub

Le dépôt GitHub constitue un élément essentiel de la valorisation de ce projet. Il permet de rendre l'ensemble du travail accessible à d'autres personnes, que ce soit pour une reproduction fidèle du projet, pour de futures améliorations ou pour des retours techniques et pédagogiques. Ce dépôt regroupe l'intégralité des fichiers nécessaires à la mise en place du projet, y compris les scripts, les fichiers de configuration, ainsi que la documentation complète.

Le dépôt GitHub est organisé de manière à offrir une structure claire et cohérente, permettant à tout utilisateur de se repérer facilement et d'utiliser les ressources mises à disposition.

En rendant ce projet accessible sur GitHub, je m'assure qu'il peut être utilisé à des fins pédagogiques et professionnelles par d'autres développeurs, chercheurs, ou professionnels de la cybersécurité. Cela permet de partager des pratiques, d'échanger des idées, d'obtenir des retours constructifs et d'améliorer continuellement le projet. Ce partage est également un moyen de garantir que les résultats obtenus dans ce projet sont non seulement accessibles à la communauté mais aussi vérifiables et reproductibles.

II. Lien GitHub

Le projet est désormais disponible en ligne, et tout le travail accompli peut être consulté, modifié ou utilisé par d'autres utilisateurs. Voici le lien vers le dépôt GitHub où l'intégralité du projet est accessible :

- **Lien vers le dépôt GitHub** : https://github.com/AYMEN-RN/SIEM_lab

En partageant ce projet sur GitHub, je m'assure que les résultats sont non seulement accessibles à la communauté mais aussi vérifiables et reproductibles. Cela permet à d'autres utilisateurs de tester, modifier ou étendre ce projet en fonction de leurs besoins, ou d'expérimenter de nouvelles fonctionnalités. La possibilité d'intégrer des retours et d'améliorer le projet à l'avenir est également un des grands avantages de la plateforme.

III. Impact pédagogique et professionnel

Ce projet a une valeur pédagogique évidente, car il permet de comprendre les concepts de la cybersécurité en pratique. Il démontre comment mettre en place une solution complète de surveillance et de réponse aux incidents dans un environnement Cloud, tout en offrant une compréhension concrète de l'implémentation d'un SIEM (Security Information and Event Management) et d'un EDR (Endpoint Detection and Response). Ce projet m'a permis d'acquérir

une compréhension approfondie des outils utilisés dans les environnements de sécurité moderne.

Du point de vue professionnel, ce projet est une démonstration de la capacité à déployer une solution de cybersécurité robuste dans le Cloud, un secteur en forte demande dans l'industrie. Les compétences acquises en matière de gestion des systèmes de sécurité, de configuration des agents de surveillance, et de gestion des flux de données dans un environnement Cloud sont directement applicables dans des projets réels, ce qui augmente considérablement mon employabilité dans le domaine de la cybersécurité.

IV. Conclusion de ce chapitre

Le **Chapitre 6** vise à mettre en lumière l'impact et la valeur ajoutée de ce projet, à la fois d'un point de vue académique et professionnel. Le dépôt GitHub, bien structuré et documenté, joue un rôle clé dans cette valorisation en offrant une ressource ouverte, accessible et facilement réutilisable. Ce projet montre non seulement la mise en œuvre concrète d'une solution de sécurité dans un environnement Cloud, mais aussi l'engagement vers l'ouverture et le partage des connaissances avec la communauté. Enfin, le projet reflète les compétences acquises et l'impact qu'il peut avoir dans le domaine de la cybersécurité.