

Département Mathématique et Informatique

Filière : Génie de logiciel et système informatique distribuée

Compte rendu

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

Réalisé par :

Chaimae Mouhssine

Encadré par :

Pr. Azeddine KHIAT

Table of content

1. Introduction.....	3
2. Objectifs pédagogiques de l'atelier.....	3
3. Architecture du laboratoire	3
4. Mise en place de l'infrastructure Cloud AWS	4
4.1 Création d'un VPC personnalisé.....	4
4.2 Création du subnet public	5
4.3 Internet Gateway et table de routage	6
5. Déploiement des instances EC2 dans le VPC	7
5.1 Client Linux	8
5.2 Windows Client.....	9
6. Configuration des Security Groups.....	11
7. Installation et configuration de Wazuh All-in-One	12
8. Enrôler le client Linux et Windows	16
9. Scénarios de démonstration SIEM & EDR.....	23
9.1 Côté Linux	23
9.1.1 Scénario 1 — Tentatives SSH échouées (bruteforce simulé)	23
9.1.2 Scénario 2 — Élévation de privileges.....	24
9.1.3 Scénario 3 — Modification fichier sensible	25
9.2 Scénarios côté Windows	26
9.2.1 Scénario 1 — Échecs de login	26
9.2.2 Scénario 2 — Crédit d'un utilisateur local.....	28
9.2.3 Scénario 3— Option “EDR plus riche”: installer Sysmon	28
10. Visualisation dans le Dashboard Wazuh	30
11. Conclusion	31
Webographie	32

1. Introduction

Dans un contexte où les cyberattaques ciblant les endpoints sont de plus en plus fréquentes et sophistiquées, la mise en place de solutions de supervision et de détection avancée devient indispensable.

Cet atelier a pour objectif la mise en œuvre complète d'une plateforme de sécurité basée sur Wazuh, combinant les approches SIEM (Security Information and Event Management) et EDR (Endpoint Detection and Response), déployée dans un environnement Cloud AWS.

L'atelier s'appuie sur un laboratoire multi-systèmes (**Linux et Windows**) afin de reproduire un contexte réaliste proche de celui rencontré en entreprise ou dans un SOC (Security Operations Center).

2. Objectifs pédagogiques de l'atelier

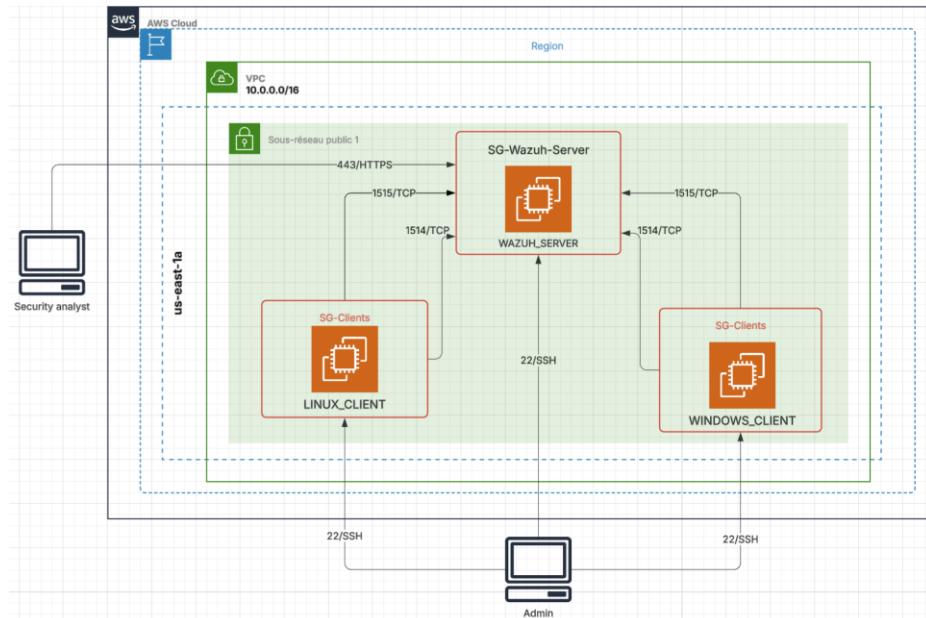
Les objectifs principaux de cet atelier sont :

- Concevoir une architecture Cloud sécurisée sur AWS
- Créer et configurer un réseau virtuel isolé (VPC)
- Déployer une solution SIEM & EDR centralisée avec Wazuh
- Superviser et analyser des événements de sécurité réels
- Comprendre la corrélation des événements multi-OS
- Mettre en évidence les capacités de détection et de réponse aux incidents

3. Architecture du laboratoire

L'architecture déployée est composée des éléments suivants :

- **EC2-1 (Ubuntu 22.04)** : Serveur Wazuh (Manager, Indexer, Dashboard)
- **EC2-2 (Ubuntu 22.04)** : Client Linux avec agent Wazuh
- **EC2-3 (Windows Server)** : Client Windows avec agent Wazuh et option Sysmon



Flux réseau principaux :

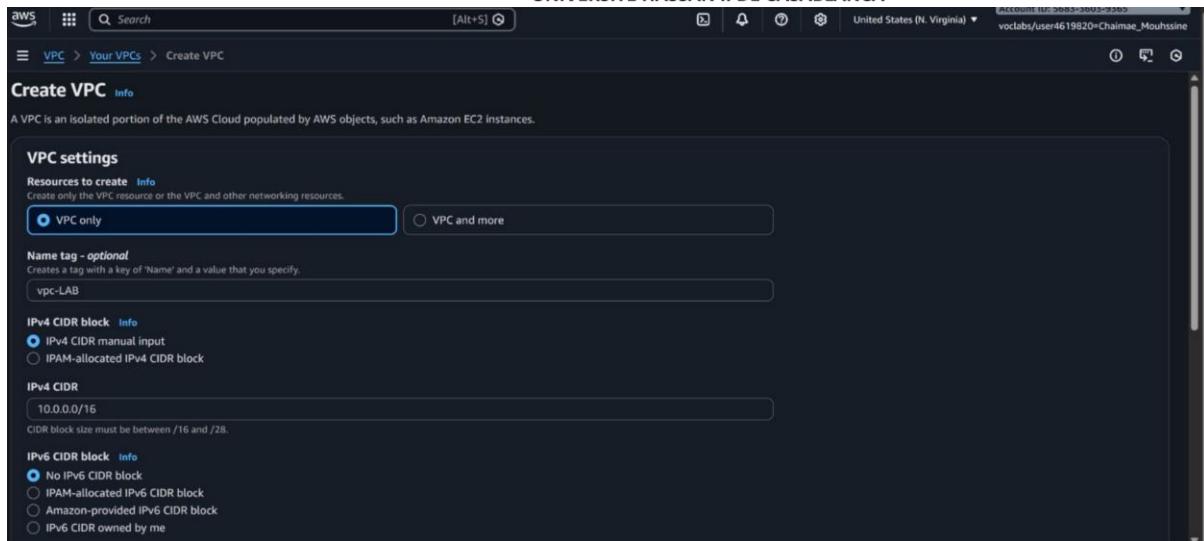
- Agents → Serveur Wazuh : 1514/TCP
- Enrôlement des agents : 1515/TCP
- Accès au Dashboard Wazuh : 443/TCP (HTTPS)

4. Mise en place de l'infrastructure Cloud AWS

4.1 Crédation d'un VPC personnalisé

Afin d'isoler totalement l'environnement du laboratoire et de maîtriser le réseau, **un VPC dédié a été créé manuellement**.

- **Nom du VPC** : VPC-SIEM-LAB
- **Plage CIDR** : 10.0.0.0/16



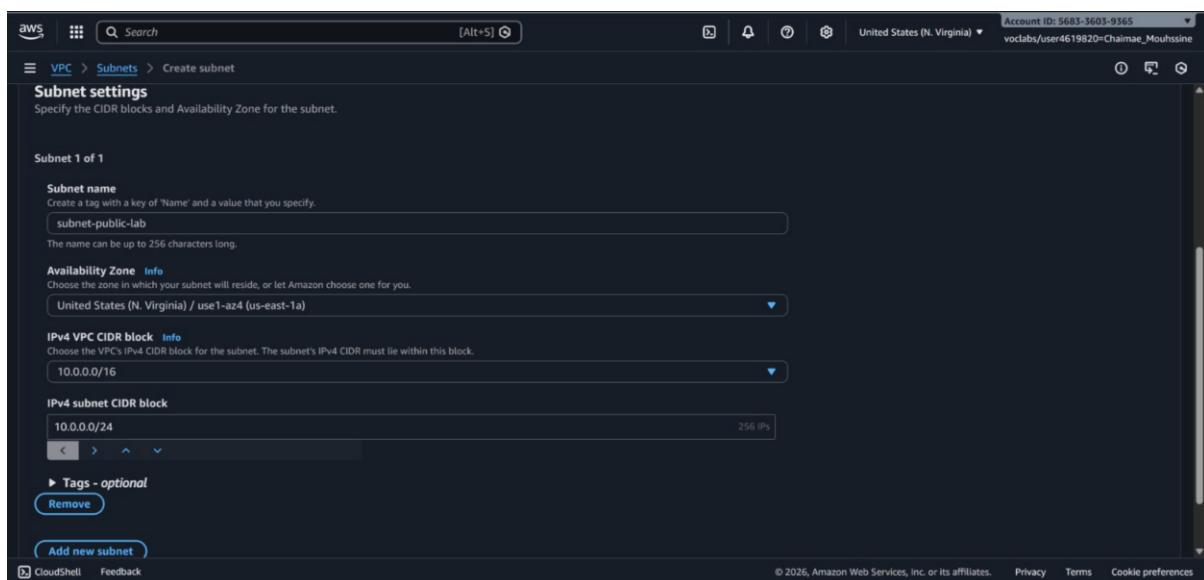
A screenshot of the AWS VPC creation interface. The 'Resources to create' section has 'VPC only' selected. Other options like 'VPC and more' and 'IPAM-allocated IPv4 CIDR block' are shown. A 'Name tag - optional' field contains 'vpc-LAB'. Under 'IPv4 CIDR block', 'IPv4 CIDR manual input' is selected, with '10.0.0.0/16' entered. The 'IPv6 CIDR block' section shows 'No IPv6 CIDR block' selected.

La création d'un VPC personnalisé permet d'éviter l'utilisation du VPC par défaut d'AWS et offre un meilleur contrôle sur le routage, la sécurité et l'isolation réseau.

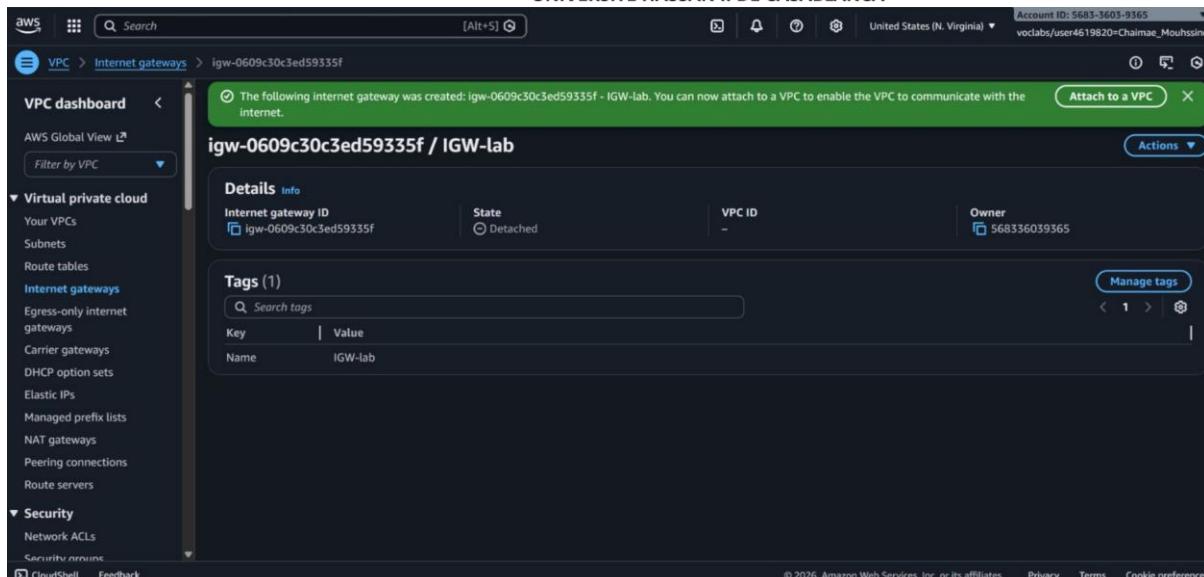
4.2 Création du subnet public

Un subnet public a été configuré pour héberger toutes les instances EC2.

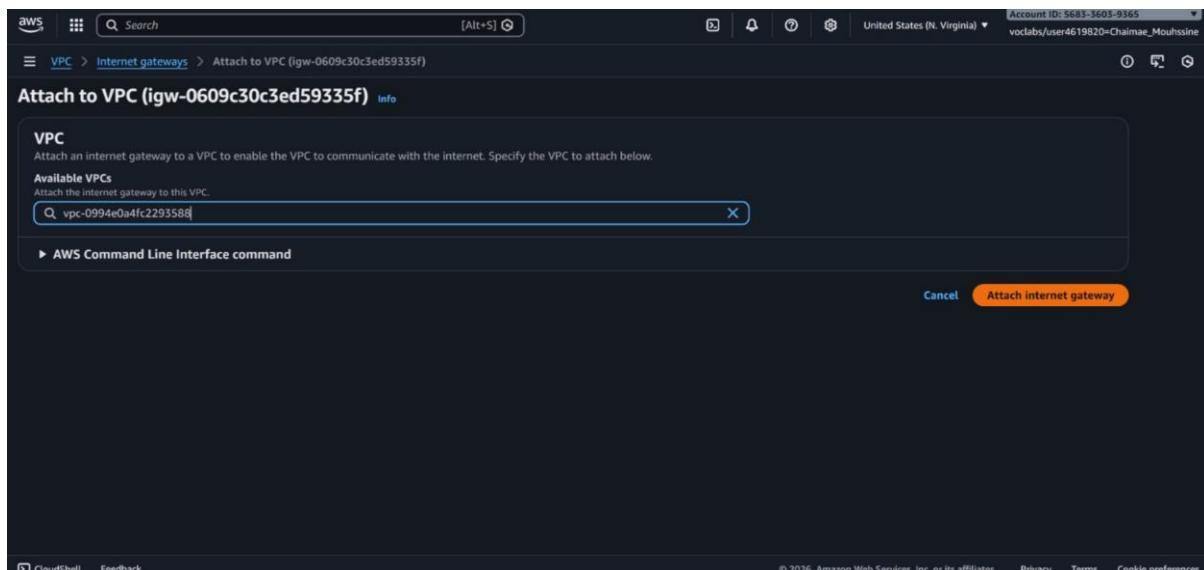
- **Nom du subnet :** Subnet-Public-SIEM
- **CIDR :** 10.0.1.0/24
- **Attribution automatique d'IP publique :** activée



A screenshot of the AWS Subnet creation interface. It shows the 'Subnet settings' section where 'Subnet name' is set to 'subnet-public-lab', 'Availability Zone' is 'United States (N. Virginia) / us-east-1a (us-east-1a)', and 'IPv4 VPC CIDR block' is '10.0.0.0/16'. The 'IPv4 subnet CIDR block' is also '10.0.0.0/24'. There are sections for 'Tags - optional' and 'Add new subnet'.



The screenshot shows the AWS VPC Internet Gateways dashboard. A message at the top indicates that an internet gateway has been created and can now be attached to a VPC. The main card displays details for 'igw-0609c30c3ed59335f / IGW-lab', including its Internet gateway ID, state (Detached), VPC ID (empty), and owner (Account ID: 5683-3603-9365). It also shows a single tag named 'Name' with the value 'IGW-lab'. The left sidebar lists various VPC-related options like Your VPCs, Subnets, Route tables, and Internet gateways.



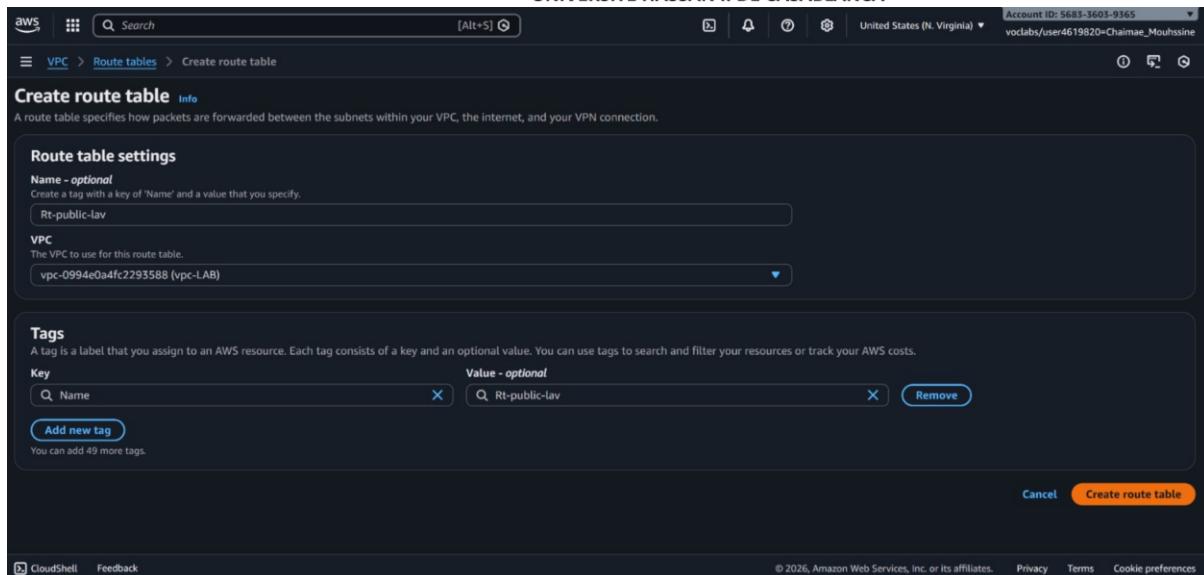
This screenshot shows a confirmation dialog box titled 'Attach to VPC (igw-0609c30c3ed59335f)'. It asks to attach an internet gateway to a VPC to enable communication with the internet. It lists 'Available VPCs' and shows one selected: 'vpc-0994e0a4fc2293588'. There is an 'AWS Command Line Interface command' section below. At the bottom right are 'Cancel' and 'Attach internet gateway' buttons.

Ce choix simplifie la communication entre les instances et permet l'accès externe au dashboard Wazuh et aux services distants (SSH / RDP).

4.3 Internet Gateway et table de routage

- Création d'une Internet Gateway (IGW)
- Association de l'IGW au VPC
- Création d'une table de routage publique

Ajout de la route:



A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

Route table settings

Name - optional
Create a tag with a key of 'Name' and a value that you specify.
Rt-public-lav

VPC
The VPC to use for this route table.
vpc-0994e0a4fc2295588 (vpc-LAB)

Tags
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - optional
Q Name	Q Rt-public-lav

Add new tag

You can add 49 more tags.

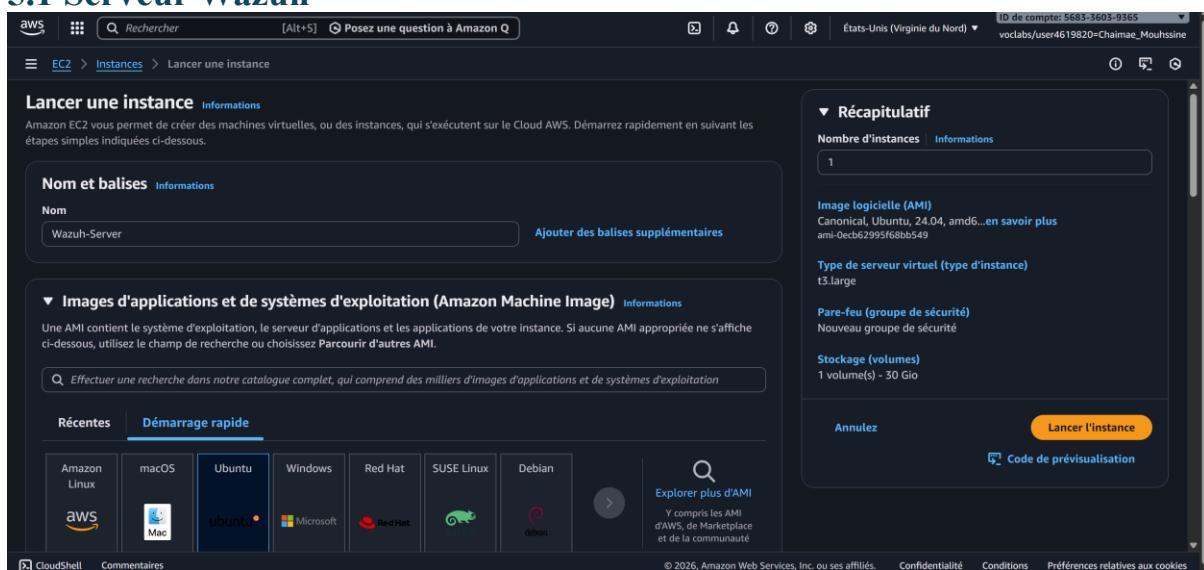
Cancel Create route table

Cette configuration garantit l'accès Internet pour les instances tout en conservant une architecture réseau simple et efficace.

5. Déploiement des instances EC2 dans le VPC

Toutes les instances ont été déployées dans le même VPC et le même subnet, assurant une communication privée directe.

5.1 Serveur Wazuh



Lancer une instance Informations

Amazon EC2 vous permet de créer des machines virtuelles, ou des instances, qui s'exécutent sur le Cloud AWS. Démarrer rapidement en suivant les étapes simples indiquées ci-dessous.

Nom et balises Informations

Nom
Wazuh-Server

Ajouter des balises supplémentaires

Images d'applications et de systèmes d'exploitation (Amazon Machine Image) Informations

Une AMI contient le système d'exploitation, le serveur d'applications et les applications de votre instance. Si aucune AMI appropriée ne s'affiche ci-dessous, utilisez le champ de recherche ou choisissez Parcourir d'autres AMI.

Effectuer une recherche dans notre catalogue complet, qui comprend des milliers d'images d'applications et de systèmes d'exploitation

Récentes Démarrage rapide

Amazon Linux	macOS	Ubuntu	Windows	Red Hat	SUSE Linux	Debian

Explorer plus d'AMI
Y compris les AMI d'AWS, de Marketplace et de la communauté

Récapitulatif

Nombre d'instances Informations
1

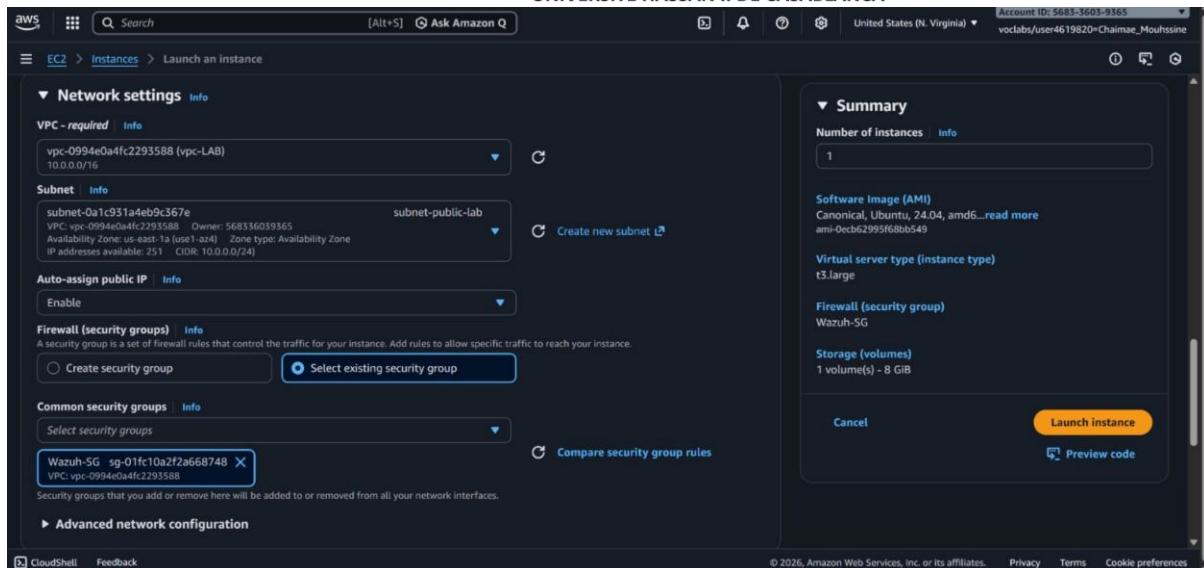
Image logicielle (AMI)
Canonical, Ubuntu, 24.04, amd64...en savoir plus
ami-0eb62995f686b549

Type de serveur virtuel (type d'instance)
t3.large

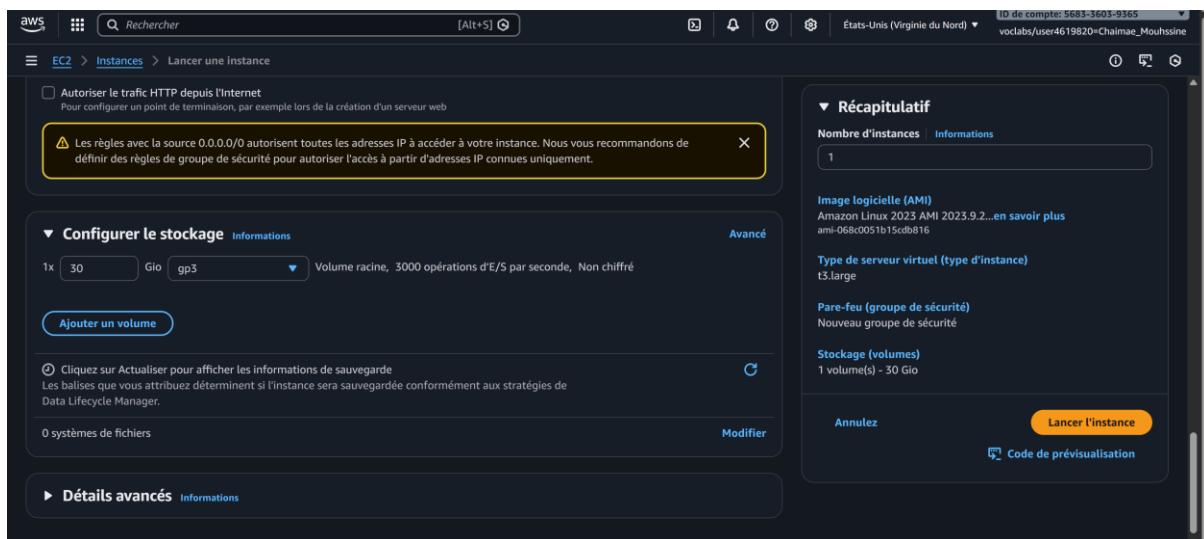
Par-feu (groupe de sécurité)
Nouveau groupe de sécurité

Stockage (volumes)
1 volume(s) - 30 Gio

Annuler Lancer l'instance Code de prévisualisation



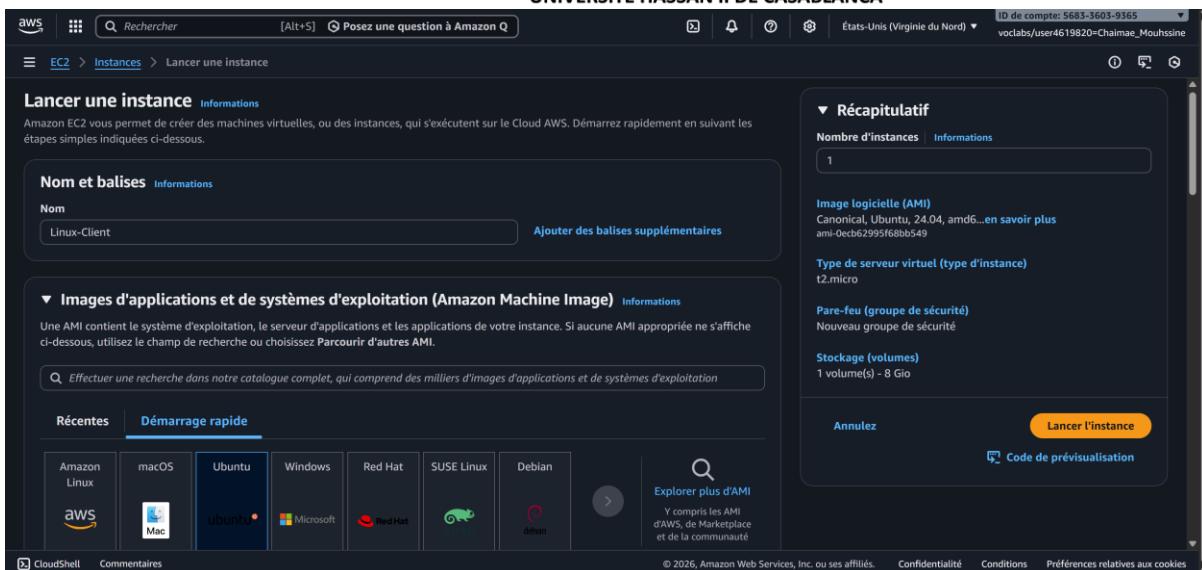
This screenshot shows the 'Network settings' step of the AWS EC2 instance launch wizard. It includes fields for VPC, Subnet, Auto-assign public IP, Firewall (security groups), and Common security groups. On the right, a summary panel shows 1 instance, the AMI (Canonical, Ubuntu, 24.04, amd64), instance type (t3.large), and storage (1 volume(s) - 8 GiB). Buttons for 'Launch instance' and 'Preview code' are at the bottom.



This screenshot shows the 'Configure the storage' step of the AWS EC2 instance launch wizard. It shows a 30 GiB gp3 volume being added. A note about enabling HTTP traffic is visible. On the right, a summary panel shows the AMI (Amazon Linux 2023 AMI 2023.9.2...), instance type (t3.large), and storage (1 volume(s) - 30 GiB). Buttons for 'Annuler' (Cancel) and 'Lancer l'instance' (Launch instance) are at the bottom.

5.1 Client Linux

- Ubuntu 22.04 (t2.micro)



Lancer une instance Informations

Amazon EC2 vous permet de créer des machines virtuelles, ou des instances, qui s'exécutent sur le Cloud AWS. Démarrer rapidement en suivant les étapes simples indiquées ci-dessous.

Nom et balises Informations

Nom: Linux-Client Ajouter des balises supplémentaires

Images d'applications et de systèmes d'exploitation (Amazon Machine Image) Informations

Une AMI contient le système d'exploitation, le serveur d'applications et les applications de votre instance. Si aucune AMI appropriée ne s'affiche ci-dessous, utilisez le champ de recherche ou choisissez Parcourir d'autres AMI.

Effectuer une recherche dans notre catalogue complet, qui comprend des milliers d'images d'applications et de systèmes d'exploitation

Récapitulatif

Nombre d'instances: Informations 1

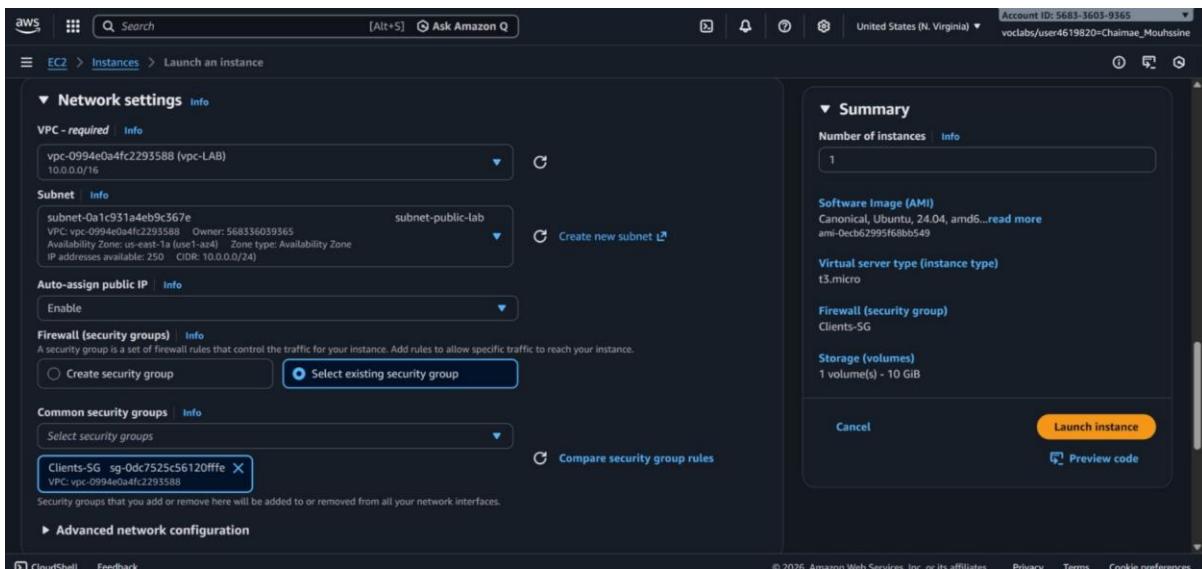
Image logicielle (AMI): Canonical, Ubuntu, 24.04, amd64...en savoir plus
ami-0ecb62995f68bb549

Type de serveur virtuel (type d'instance): t2.micro

Pare-feu (groupe de sécurité): Nouveau groupe de sécurité

Stockage (volumes): 1 volume(s) - 8 Go

Annuler Lancer l'instance Code de prévisualisation



Network settings Info

VPC - required Info

vpc-0994e0a4fc2293588 (vpc-LAB) 10.0.0.0/16

Subnet: subnet-0a1c931a4eb9c367e Create new subnet

Auto-assign public IP: Enable

Firewall (security groups): Info

Create security group Select existing security group

Common security groups: Info

Select security groups Clients-SG

Advanced network configuration

Summary

Number of instances: Info 1

Software Image (AMI): Canonical, Ubuntu, 24.04, amd64...read more
ami-0ecb62995f68bb549

Virtual server type (instance type): t2.micro

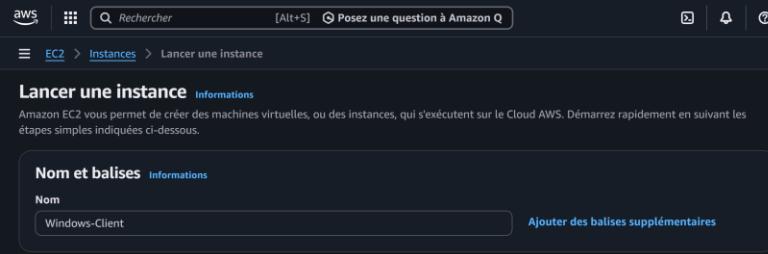
Firewall (security group): Clients-SG

Storage (volumes): 1 volume(s) - 10 GiB

Cancel Launch instance Preview code

5.2 Windows Client

- Windows Server (t3.medium)



Lancer une instance Informations

Amazon EC2 vous permet de créer des machines virtuelles, ou des instances, qui s'exécutent sur le Cloud AWS. Démarrer rapidement en suivant les étapes simples indiquées ci-dessous.

Nom et balises Informations

Nom

Windows-Client Ajouter des balises supplémentaires

Images d'applications et de systèmes d'exploitation (Amazon Machine Image) Informations

Une AMI contient le système d'exploitation, le serveur d'applications et les applications de votre instance. Si aucune AMI appropriée ne s'affiche ci-dessous, utilisez le champ de recherche ou choisissez Parcourir d'autres AMI.

Effectuer une recherche dans notre catalogue complet, qui comprend des milliers d'images d'applications et de systèmes d'exploitation

Récentes **Démarrage rapide**

Amazon Linux | macOS | Ubuntu | Windows | Red Hat | SUSE Linux | Debian | > | Q Explorer plus d'AMI
Y compris les AMI d'AWS, de Marketplace et de la communauté

Récapitulatif

Nombre d'instances Informations

1

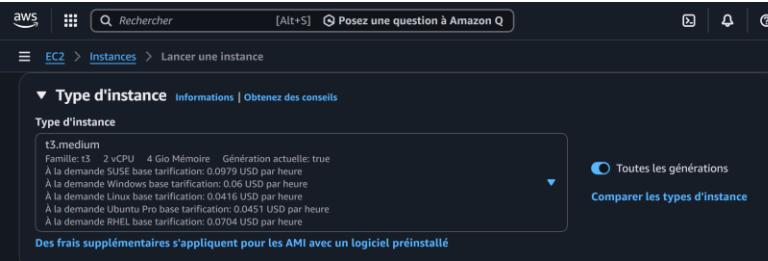
Image logicielle (AMI)
Microsoft Windows Server 2025 ...en savoir plus
ami-06777e7ef7441def

Type de serveur virtuel (type d'instance)
t3.medium

Pare-feu (groupe de sécurité)
Nouveau groupe de sécurité

Stockage (volumes)
1 volume(s) - 30 Gio

Annuler Lancer l'instance Code de prévisualisation



Type d'instance Informations | Obtenez des conseils

Type d'instance

t3.medium

Famille: t3 2 vCPU 4 Go Mémoire Génération actuelle: true
À la demande SUSE base tarification: 0.0979 USD par heure
À la demande Windows base tarification: 0.06 USD par heure
À la demande Linux base tarification: 0.0416 USD par heure
À la demande Ubuntu Pro base tarification: 0.0451 USD par heure
À la demande RHEL base tarification: 0.0704 USD par heure

Toutes les générations Comparer les types d'instance

Récapitulatif

Nombre d'instances Informations

1

Image logicielle (AMI)
Microsoft Windows Server 2025 ...en savoir plus
ami-06777e7ef7441def

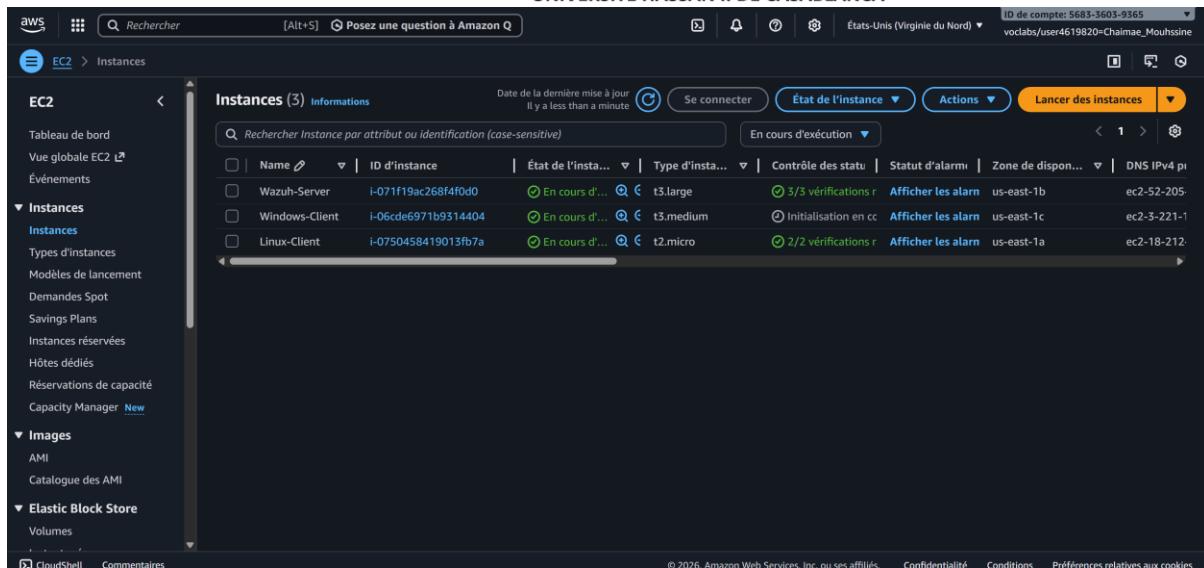
Type de serveur virtuel (type d'instance)
t3.medium

Pare-feu (groupe de sécurité)
Nouveau groupe de sécurité

Stockage (volumes)
1 volume(s) - 30 Gio

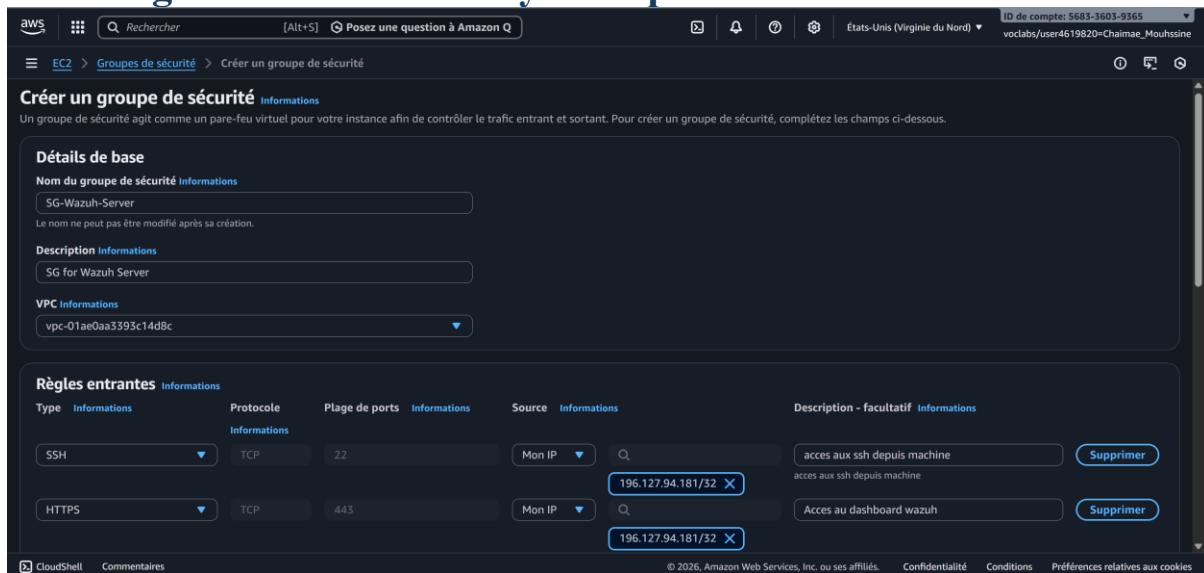
Annuler Lancer l'instance Code de prévisualisation

Les instances sont créées avec succès

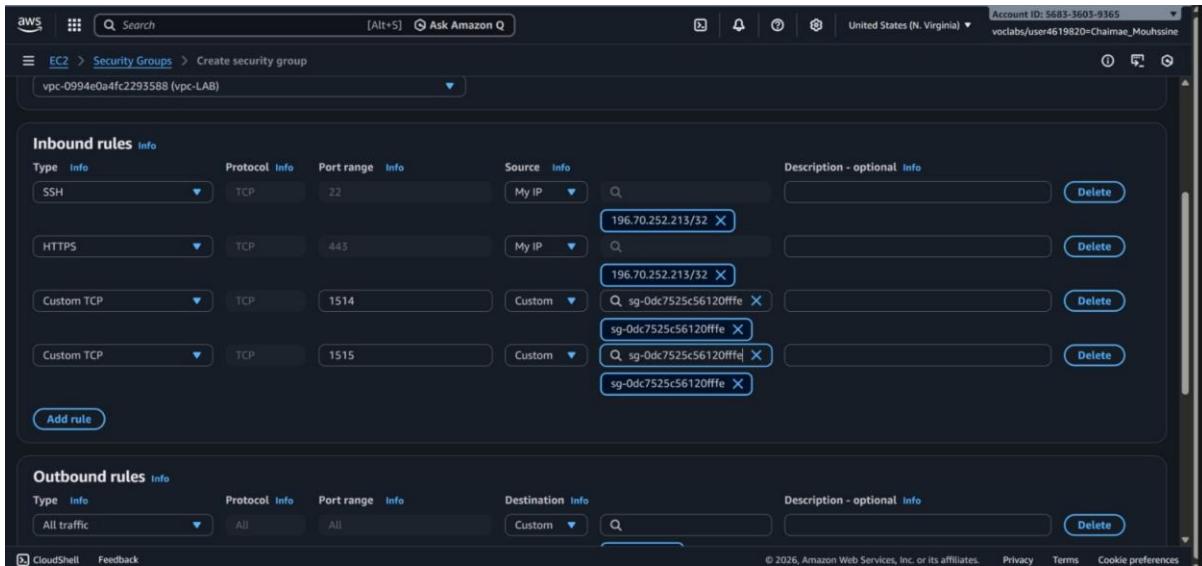


The screenshot shows the AWS EC2 Instances dashboard. On the left, there's a sidebar with navigation links like 'Tableau de bord', 'Instances', 'Types d'instances', 'Modèles de lancement', 'Demandes Spot', 'Savings Plans', 'Instances réservées', 'Hôtes dédiés', 'Réservations de capacité', and 'Capacity Manager'. The main area displays a table of three instances: 'Wazuh-Server' (t3.large), 'Windows-Client' (t3.medium), and 'Linux-Client' (t2.micro). Each instance has a status column showing 'En cours d...', a dropdown menu for actions, and a link to 'Afficher les alarmes'.

6. Configuration des Security Groups



The screenshot shows the 'Créer un groupe de sécurité' (Create Security Group) form. In the 'Détails de base' section, the group name is 'SG-Wazuh-Server' and the description is 'SG for Wazuh Server'. Under 'VPC Informations', the VPC is set to 'vpc-01ae0aa3395c14d8c'. In the 'Règles entrantes' (Inbound Rules) section, there are two rules: one for SSH (Protocol TCP, Port 22, Source 'Mon IP' with value '196.127.94.181/32') and one for HTTPS (Protocol TCP, Port 443, Source 'Mon IP' with value '196.127.94.181/32'). Both rules have a description of 'acces aux ssh depuis machine' and 'Accès au dashboard wazuh' respectively, with a 'Supprimer' (Delete) button next to each.



The screenshot shows the AWS Management Console interface for managing security groups. The top navigation bar includes 'Search', 'Ask Amazon Q', 'United States (N. Virginia)', and 'Account ID: 5683-3603-9365'. The main page title is 'EC2 > Security Groups > Create security group'. Below this, there are two sections: 'Inbound rules' and 'Outbound rules'. The 'Inbound rules' section lists four entries: 'SSH' (TCP port 22 from 'My IP'), 'HTTPS' (TCP port 443 from 'My IP'), 'Custom TCP' (TCP port 1514 from 'Custom' source), and 'Custom TCP' (TCP port 1515 from 'Custom' source). The 'Outbound rules' section shows 'All traffic' (All ports to Custom destination). At the bottom, there are links for 'CloudShell', 'Feedback', and copyright information: '© 2026, Amazon Web Services, Inc. or its affiliates.' and 'Cookie preferences'.

7. Installation et configuration de Wazuh All-in-One

- Enrôlement via le Dashboard Wazuh

sudo apt update && sudo apt -y upgrade

```
ubuntu@ip-10-0-0-20:~$ 
ubuntu@ip-10-0-0-20:~$ sudo apt update && sudo apt -y upgrade
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [127 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [14.1 MB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe Translation-en [5652 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 c-n-f Metadata [286 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [217 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse Translation-en [112 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 c-n-f Metadata [8372 B]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [3161 kB]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [484 kB]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 c-n-f Metadata [19.0 kB]
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [5043 kB]
```

curl -sO https://packages.wazuh.com/4.7/wazuh-install.sh

```
ubuntu@ip-10-0-0-20:~$ 
ubuntu@ip-10-0-0-20:~$ curl -sO https://packages.wazuh.com/4.7/wazuh-install.sh
ubuntu@ip-10-0-0-20:~$
```

sudo bash wazuh-install.sh -a

```
ubuntu@ip-10-0-0-20:~$ sudo bash wazuh-install.sh -a
04/01/2026 10:26:31 INFO: Starting Wazuh installation assistant. Wazuh version: 4.7.5
04/01/2026 10:26:31 INFO: Verbose logging redirected to /var/log/wazuh-install.log
04/01/2026 10:26:40 INFO: Wazuh web interface port will be 443.
04/01/2026 10:26:45 INFO: --- Dependencies ---
04/01/2026 10:26:45 INFO: Installing apt-transport-https.
04/01/2026 10:26:50 INFO: Wazuh repository added.
04/01/2026 10:26:50 INFO: --- Configuration files ---
04/01/2026 10:26:50 INFO: Generating configuration files.
04/01/2026 10:26:52 INFO: Created wazuh-install-files.tar. It contains the Wazuh cluster key, certificates, and password
s necessary for installation.
04/01/2026 10:26:53 INFO: --- Wazuh indexer ---
04/01/2026 10:26:53 INFO: Starting Wazuh indexer installation.
04/01/2026 10:27:55 INFO: Wazuh indexer installation finished.
04/01/2026 10:27:55 INFO: Wazuh indexer post-install configuration finished.
04/01/2026 10:27:55 INFO: Starting service wazuh-indexer.
04/01/2026 10:28:15 INFO: wazuh-indexer service started.
04/01/2026 10:28:15 INFO: Initializing Wazuh indexer cluster security settings.
04/01/2026 10:28:27 INFO: Wazuh indexer cluster initialized.
04/01/2026 10:28:27 INFO: --- Wazuh server ---
04/01/2026 10:28:27 INFO: Starting the Wazuh manager installation.
04/01/2026 10:29:20 INFO: Wazuh manager installation finished.
04/01/2026 10:29:20 INFO: Starting service wazuh-manager.
04/01/2026 10:29:37 INFO: wazuh-manager service started.
04/01/2026 10:29:37 INFO: Starting Filebeat installation.
04/01/2026 10:29:46 INFO: Filebeat installation finished.
04/01/2026 10:29:47 INFO: Filebeat post-install configuration finished.
04/01/2026 10:29:47 INFO: Starting service filebeat.
04/01/2026 10:29:48 INFO: filebeat service started.
04/01/2026 10:29:48 INFO: --- Wazuh dashboard ---
04/01/2026 10:29:48 INFO: Starting Wazuh dashboard installation.
04/01/2026 10:30:41 INFO: Wazuh dashboard installation finished.
04/01/2026 10:30:42 INFO: Wazuh dashboard post-install configuration finished.
04/01/2026 10:30:42 INFO: Starting service wazuh-dashboard.
04/01/2026 10:30:42 INFO: wazuh-dashboard service started.
04/01/2026 10:31:17 INFO: Initializing Wazuh dashboard web application.
04/01/2026 10:31:17 INFO: Wazuh dashboard web application not yet initialized. Waiting...
04/01/2026 10:31:33 INFO: Wazuh dashboard web application not yet initialized. Waiting...
04/01/2026 10:31:48 INFO: Wazuh dashboard web application initialized.
04/01/2026 10:31:48 INFO: --- Summary ---
04/01/2026 10:31:48 INFO: You can access the web interface https://<wazuh-dashboard-ip>:443
  User: admin
  Password: QFsi?vvUedlsgXbupZ*RVaixAf5oYhN4
04/01/2026 10:31:48 INFO: Installation finished.
ubuntu@ip-10-0-0-20:~$
```

À la fin, le script fournit :

- **URL du dashboard:** <https://54.196.16.171:443>
- **Utilisateur :** admin
- **Mot de passe :** CsJpEbgRNAZR495aMkkvukVUceUvdv+9

Vérification des services :

```
ubuntu@ip-10-0-0-20:~$ sudo systemctl status wazuh-manager
● wazuh-manager.service - Wazuh manager
   Loaded: loaded (/lib/systemd/system/wazuh-manager.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2026-01-04 10:29:37 UTC; 28min ago
     Tasks: 121 (limit: 4580)
    Memory: 266.3M
      CPU: 48.378s
     CGroup: /system.slice/wazuh-manager.service
             └─58728 /var/ossec/framework/python/bin/python3 /var/ossec/api/scripts/wazuh-apid.py
               ├─58767 /var/ossec/bin/wazuh-authd
               ├─58781 /var/ossec/bin/wazuh-db
               ├─58804 /var/ossec/bin/wazuh-execd
               ├─58815 /var/ossec/bin/wazuh-analysisd
               ├─58859 /var/ossec/bin/wazuh-syscheckd
               ├─58875 /var/ossec/bin/wazuh-remoted
               ├─58898 /var/ossec/framework/python/bin/python3 /var/ossec/api/scripts/wazuh-apid.py
               ├─58901 /var/ossec/framework/python/bin/python3 /var/ossec/api/scripts/wazuh-apid.py
               ├─58904 /var/ossec/framework/python/bin/python3 /var/ossec/api/scripts/wazuh-apid.py
               ├─58917 /var/ossec/bin/wazuh-logcollector
               ├─58936 /var/ossec/bin/wazuh-monitord
               └─58958 /var/ossec/bin/wazuh-modulesd

Jan 04 10:29:29 ip-10-0-0-20 env[58672]: Started wazuh-db...
Jan 04 10:29:29 ip-10-0-0-20 env[58672]: Started wazuh-execd...
Jan 04 10:29:30 ip-10-0-0-20 env[58672]: Started wazuh-analysisd...
Jan 04 10:29:31 ip-10-0-0-20 env[58672]: Started wazuh-syscheckd...
Jan 04 10:29:32 ip-10-0-0-20 env[58672]: Started wazuh-remoted...
Jan 04 10:29:33 ip-10-0-0-20 env[58672]: Started wazuh-logcollector...
Jan 04 10:29:34 ip-10-0-0-20 env[58672]: Started wazuh-monitord...
Jan 04 10:29:35 ip-10-0-0-20 env[58672]: Started wazuh-modulesd...
Jan 04 10:29:37 ip-10-0-0-20 env[58672]: Completed.
Jan 04 10:29:37 ip-10-0-0-20 systemd[1]: Started Wazuh manager.

ubuntu@ip-10-0-0-20:~$ |
```

```
ubuntu@ip-10-0-0-20:~$ sudo systemctl status wazuh-indexer
● wazuh-indexer.service - Wazuh-indexer
   Loaded: loaded (/lib/systemd/system/wazuh-indexer.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2026-01-04 10:28:15 UTC; 34min ago
     Docs: https://documentation.wazuh.com
     Main PID: 15204 (java)
        Tasks: 74 (limit: 4580)
       Memory: 2.2G
          CPU: 1min 32.390s
     CGroup: /system.slice/wazuh-indexer.service
             └─15204 /usr/share/wazuh-indexer/jdk/bin/java -Xshare:auto -Dopensearch.networkaddress.cache.ttl=60 -Dopen>

Jan 04 10:27:55 ip-10-0-0-20 systemd[1]: Starting Wazuh-indexer...
Jan 04 10:27:58 ip-10-0-0-20 systemd-entrypoint[15204]: WARNING: A terminally deprecated method in java.lang.System has>
Jan 04 10:27:58 ip-10-0-0-20 systemd-entrypoint[15204]: WARNING: System::setSecurityManager has been called by org.open>
Jan 04 10:27:58 ip-10-0-0-20 systemd-entrypoint[15204]: WARNING: Please consider reporting this to the maintainers of o>
Jan 04 10:27:58 ip-10-0-0-20 systemd-entrypoint[15204]: WARNING: System::setSecurityManager will be removed in a future>
Jan 04 10:28:00 ip-10-0-0-20 systemd-entrypoint[15204]: WARNING: A terminally deprecated method in java.lang.System has>
Jan 04 10:28:00 ip-10-0-0-20 systemd-entrypoint[15204]: WARNING: System::setSecurityManager has been called by org.open>
Jan 04 10:28:00 ip-10-0-0-20 systemd-entrypoint[15204]: WARNING: Please consider reporting this to the maintainers of o>
Jan 04 10:28:00 ip-10-0-0-20 systemd-entrypoint[15204]: WARNING: System::setSecurityManager will be removed in a future>
Jan 04 10:28:15 ip-10-0-0-20 systemd[1]: Started Wazuh-indexer.

ubuntu@ip-10-0-0-20:~$ |
```

```
ubuntu@ip-10-0-0-20:~$ 
ubuntu@ip-10-0-0-20:~$ sudo systemctl status wazuh-dashboard
● wazuh-dashboard.service - wazuh-dashboard
   Loaded: loaded (/etc/systemd/system/wazuh-dashboard.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2026-01-04 10:31:04 UTC; 34min ago
     Main PID: 61046 (node)
        Tasks: 11 (limit: 4580)
       Memory: 210.0M
          CPU: 12.710s
        CGroup: /system.slice/wazuh-dashboard.service
              └─61046 /usr/share/wazuh-dashboard/node/bin/node --no-warnings --max-http-header-size=65536 --unhandled-rejecti>

Jan 04 10:31:15 ip-10-0-0-20 opensearch-dashboards[61046]: {"type": "log", "@timestamp": "2026-01-04T10:31:15Z", "tags": ["info"], >
Jan 04 10:31:15 ip-10-0-0-20 opensearch-dashboards[61046]: {"type": "log", "@timestamp": "2026-01-04T10:31:15Z", "tags": ["info"], >
Jan 04 10:31:15 ip-10-0-0-20 opensearch-dashboards[61046]: {"type": "log", "@timestamp": "2026-01-04T10:31:15Z", "tags": ["info"], >
Jan 04 10:31:15 ip-10-0-0-20 opensearch-dashboards[61046]: {"type": "log", "@timestamp": "2026-01-04T10:31:15Z", "tags": ["info"], >
Jan 04 10:31:16 ip-10-0-0-20 opensearch-dashboards[61046]: {"type": "log", "@timestamp": "2026-01-04T10:31:16Z", "tags": ["info"], >
Jan 04 10:31:16 ip-10-0-0-20 opensearch-dashboards[61046]: {"type": "log", "@timestamp": "2026-01-04T10:31:16Z", "tags": ["info"], >
Jan 04 10:31:16 ip-10-0-0-20 opensearch-dashboards[61046]: {"type": "log", "@timestamp": "2026-01-04T10:31:16Z", "tags": ["info"], >
Jan 04 10:31:16 ip-10-0-0-20 opensearch-dashboards[61046]: {"type": "log", "@timestamp": "2026-01-04T10:31:16Z", "tags": ["info"], >
Jan 04 10:31:16 ip-10-0-0-20 opensearch-dashboards[61046]: {"type": "log", "@timestamp": "2026-01-04T10:31:16Z", "tags": ["info"], >
Jan 04 10:31:17 ip-10-0-0-20 opensearch-dashboards[61046]: {"type": "log", "@timestamp": "2026-01-04T10:31:17Z", "tags": ["listen"], >
Jan 04 10:31:17 ip-10-0-0-20 opensearch-dashboards[61046]: {"type": "log", "@timestamp": "2026-01-04T10:31:17Z", "tags": ["info"], >
Jan 04 10:31:33 ip-10-0-0-20 opensearch-dashboards[61046]: {"type": "response", "@timestamp": "2026-01-04T10:31:32Z", "tags": []}, >

ubuntu@ip-10-0-0-20:~$ |
```

➤ Connexion à l'interface Wazuh



The dashboard displays the following agent counts:

- Total agents: 0
- Active agents: 0
- Disconnected agents: 0
- Pending agents: 0
- Never connected agents: 0

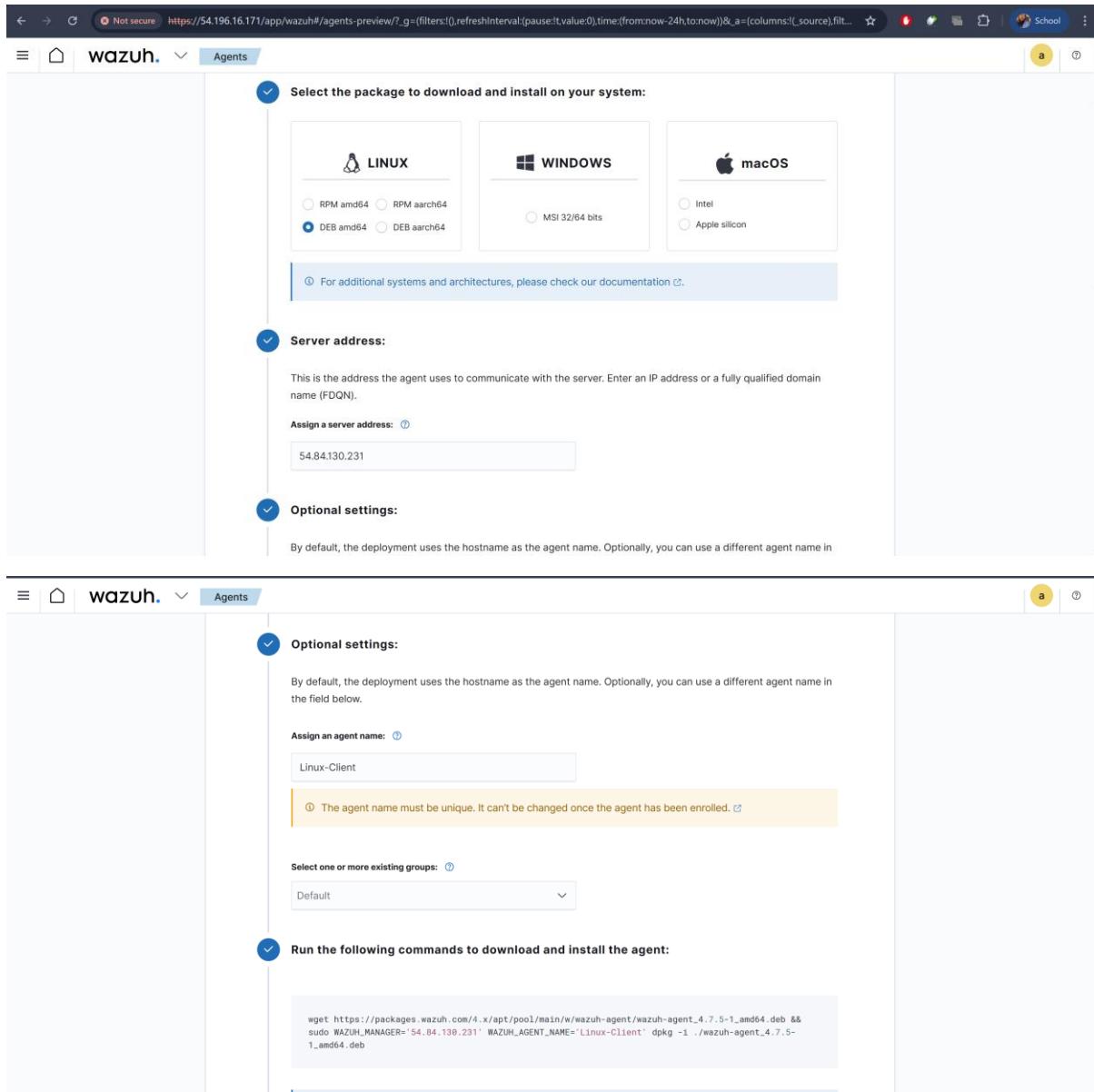
A message indicates: "⚠️ No agents were added to this manager. [Add agent](#)"

The dashboard is organized into several sections:

- SECURITY INFORMATION MANAGEMENT** includes:
 - Security events: Browse through your security alerts, identifying issues and threats in your environment.
 - Integrity monitoring: Alerts related to file changes, including permissions, content, ownership and attributes.
- AUDITING AND POLICY MONITORING** includes:
 - Policy monitoring: Verify that your systems are configured according to your security policies baseline.
 - System auditing: Audit users behavior, monitoring command execution and alerting on access to critical files.
 - Security configuration assessment: Scan your assets as part of a configuration assessment audit.
- THREAT DETECTION AND RESPONSE** includes:
 - Vulnerabilities
 - MITRE ATT&CK
- REGULATORY COMPLIANCE** includes:
 - PCI DSS
 - NIST 800-53

8. Enrôler le client Linux et Windows

- Via le dashboard Wazuh → Agents management → Deploy new agent



The screenshot shows the 'Agents' management section of the Wazuh dashboard. A modal window titled 'Deploy new agent' is open, guiding the user through the process of installing an agent on different systems.

Step 1: Select the package to download and install on your system:

- LINUX:**
 - RPM amd64
 - RPM aarch64
 - DEB amd64** (selected)
 - DEB aarch64
- WINDOWS:**
 - MSI 32/64 bits
- macOS:**
 - Intel
 - Apple silicon

For additional systems and architectures, please check our documentation.

Step 2: Server address:

This is the address the agent uses to communicate with the server. Enter an IP address or a fully qualified domain name (FQDN).

Assign a server address:

Step 3: Optional settings:

By default, the deployment uses the hostname as the agent name. Optionally, you can use a different agent name in the field below.

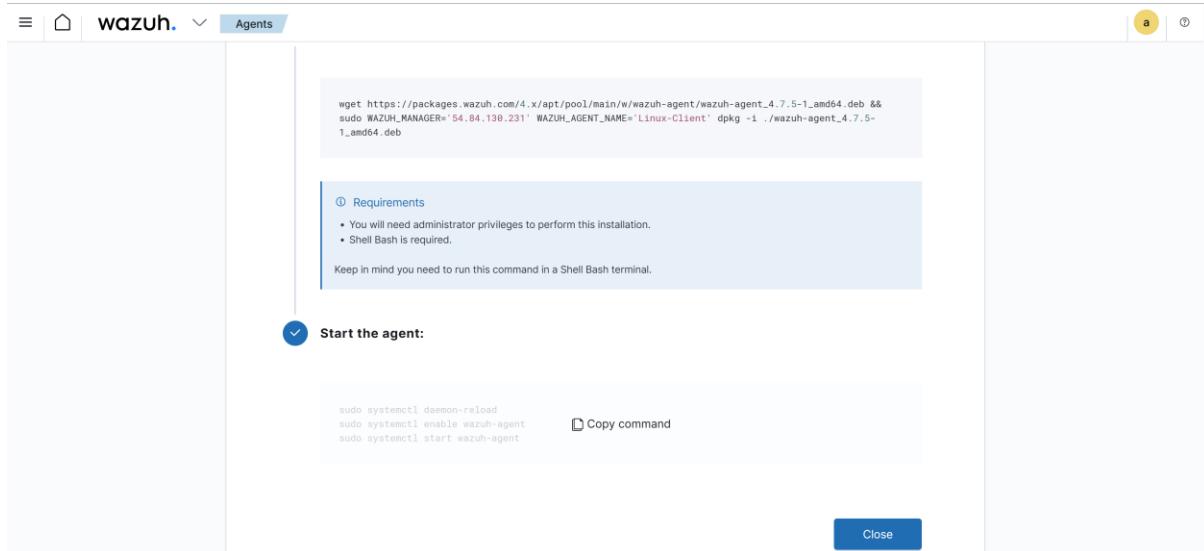
Assign an agent name:

The agent name must be unique. It can't be changed once the agent has been enrolled.

Select one or more existing groups:

Step 4: Run the following commands to download and install the agent:

```
wget https://packages.wazuh.com/4.x/apt/pool/main/w/wazuh-agent/wazuh-agent_4.7.5-1_amd64.deb &&
sudo WAZUH_MANAGER='54.84.130.231' WAZUH_AGENT_NAME='Linux-Client' dpkg -i ./wazuh-agent_4.7.5-
1_amd64.deb
```



The screenshot shows a web-based guide for installing the Wazuh agent. It includes a command to download the agent package, requirements (administrator privileges and Shell Bash), and instructions to run the command in a Shell Bash terminal. Below this, there's a section titled "Start the agent:" with a command to run `sudo systemctl daemon-reload` and `sudo systemctl enable wazuh-agent` followed by `sudo systemctl start wazuh-agent`. A "Copy command" button is also present.

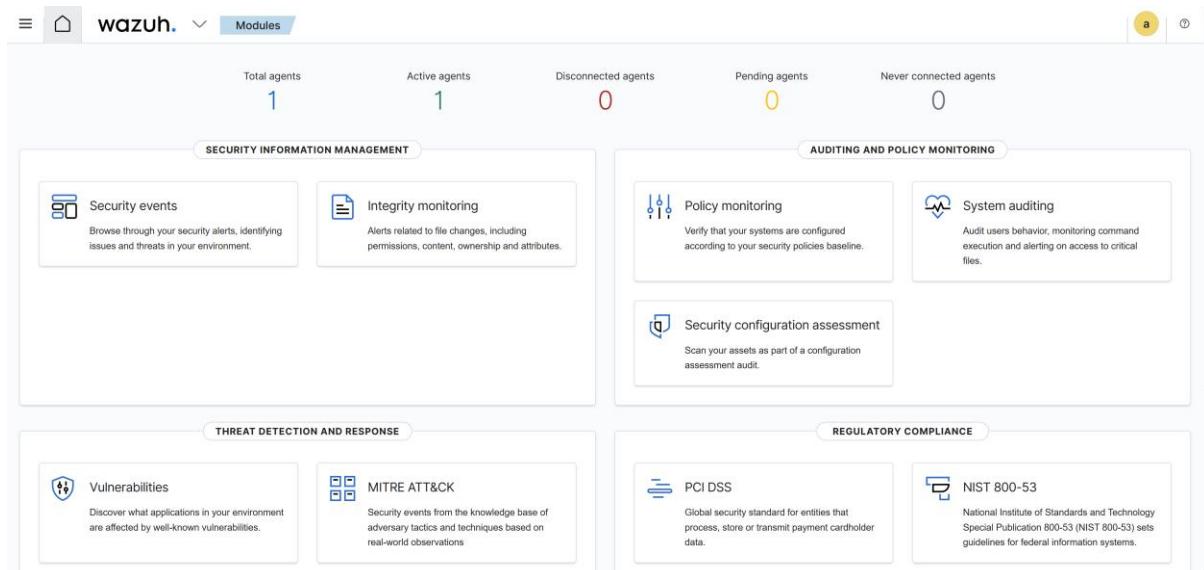
➤ Sélection Linux → Copier commandes et exécuter sur le client

```
ubuntu@ip-10-0-0-178:~$ wget https://packages.wazuh.com/4.x/apt/pool/main/w/wazuh-agent/wazuh-agent_4.7.5-1_amd64.deb &&
sudo WAZUH_MANAGER='34.229.165.166' WAZUH_AGENT_NAME='Linux-Client' dpkg -i ./wazuh-agent_4.7.5-1_amd64.deb
--2026-01-06 14:40:06-- https://packages.wazuh.com/4.x/apt/pool/main/w/wazuh-agent/wazuh-agent_4.7.5-1_amd64.deb
Resolving packages.wazuh.com (packages.wazuh.com)... 13.226.209.39, 13.226.209.78, 13.226.209.111, ...
Connecting to packages.wazuh.com (packages.wazuh.com)|13.226.209.39|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 9378818 (8.9M) [application/vnd.debian.binary-package]
Saving to: 'wazuh-agent_4.7.5-1_amd64.deb'

wazuh-agent_4.7.5-1_amd64.deb 100%[=====] 8.94M 51.7MB/s in 0.2s

2026-01-06 14:40:07 (51.7 MB/s) - 'wazuh-agent_4.7.5-1_amd64.deb' saved [9378818/9378818]

Selecting previously unselected package wazuh-agent.
(Reading database ... 71735 files and directories currently installed.)
Preparing to unpack .../wazuh-agent_4.7.5-1_amd64.deb ...
Unpacking wazuh-agent (4.7.5-1) ...
Setting up wazuh-agent (4.7.5-1) ...
ubuntu@ip-10-0-0-178:~$ sudo systemctl daemon-reload
sudo systemctl enable wazuh-agent
sudo systemctl start wazuh-agent
Created symlink /etc/systemd/system/multi-user.target.wants/wazuh-agent.service → /usr/lib/systemd/system/wazuh-agent.service.
ubuntu@ip-10-0-0-178:~$
```



The screenshot shows the Wazuh dashboard interface. At the top, it displays statistics: Total agents (1), Active agents (1), Disconnected agents (0), Pending agents (0), and Never connected agents (0). The dashboard is divided into several sections: SECURITY INFORMATION MANAGEMENT (Security events, Integrity monitoring), AUDITING AND POLICY MONITORING (Policy monitoring, System auditing, Security configuration assessment), THREAT DETECTION AND RESPONSE (Vulnerabilities, MITRE ATT&CK), and REGULATORY COMPLIANCE (PCI DSS, NIST 800-53).

WAZUh. Agents

STATUS

- Active (1)
- Disconnected (0)
- Pending (0)
- Never connected (0)

DETAILS

Active 1	Disconnected 0	Pending 0	Never connected 0	Agents coverage 100.00%
Last registered agent Linux-Client		Most active agent Linux-Client		

EVOLUTION

Last 24 hours

No results found

Agents (1)

ID	Name	IP address	Group(s)	Operating system	Cluster node	Version	Status	Actions
001	Linux-Client	10.0.0.178	default	Ubuntu 24.04.3 LTS	node01	v4.7.5	● active	🔗 🔗

Rows per page: 10 < 1 >

WAZUh. Agents

Deploy new agent

Select the package to download and install on your system:

 **LINUX**

RPM amd64 RPM aarch64
 DEB amd64 DEB aarch64

 **WINDOWS**

MSI 32/64 bits

 **macOS**

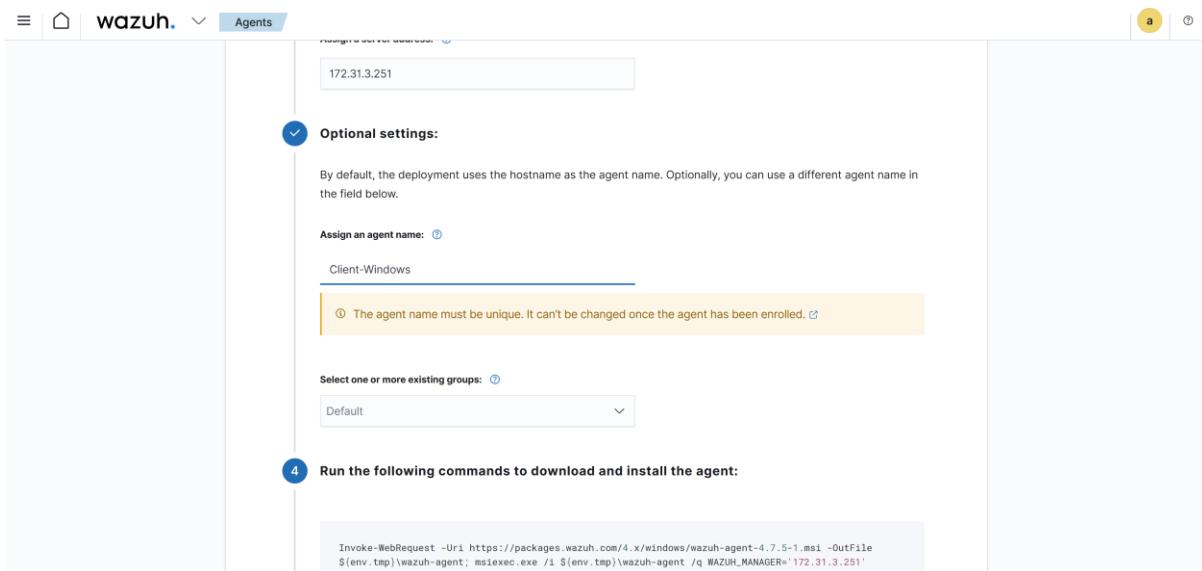
Intel Apple silicon

For additional systems and architectures, please check our documentation 

Server address:

This is the address the agent uses to communicate with the server. Enter an IP address or a fully qualified domain name (FQDN).

Assign a server address: 
172.31.3.251

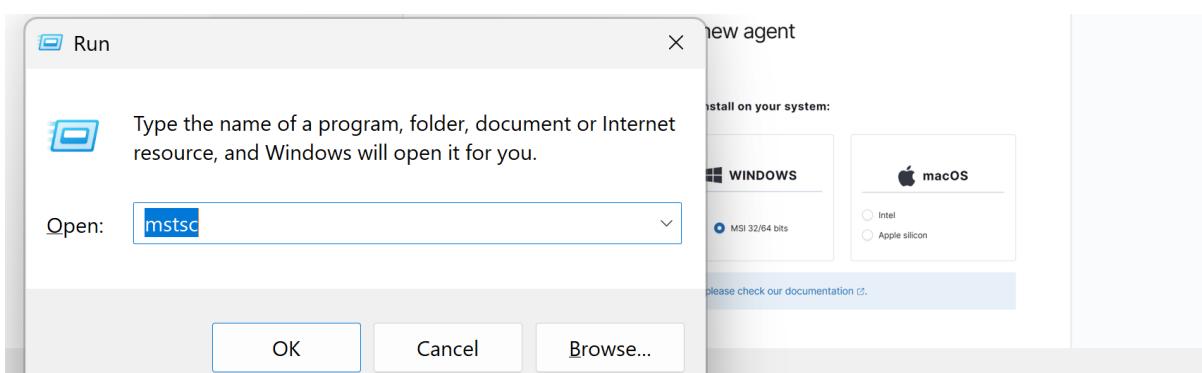
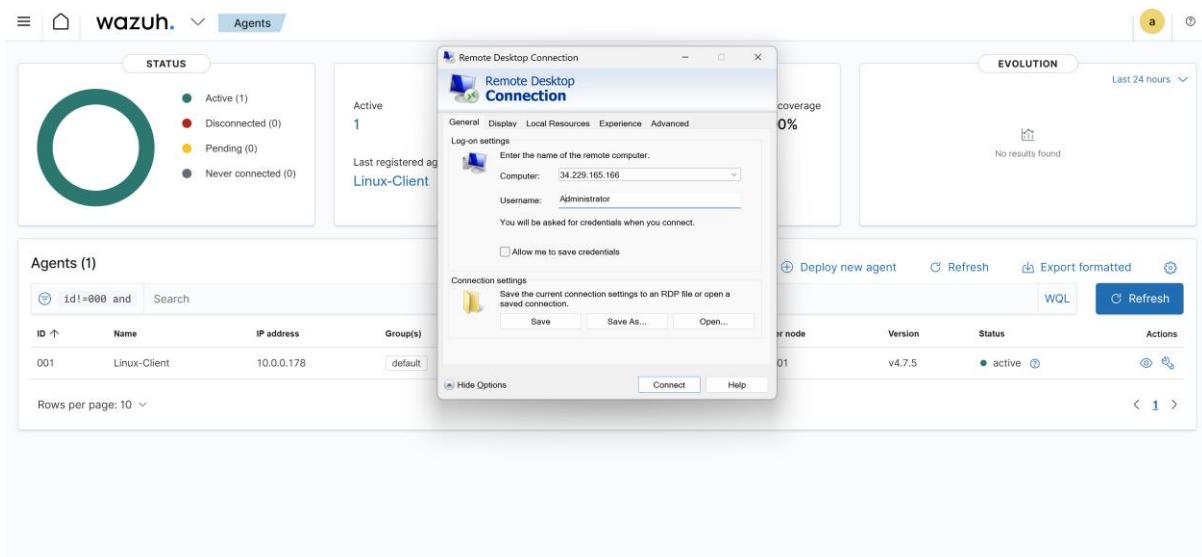


The screenshot shows the Wazuh Agent Deployment Wizard. Step 1: Search for hostnames. Step 2: Optional settings. Step 3: Assign an agent name (Client-Windows) and select groups (Default). Step 4: Run commands to download and install the agent. The command shown is:

```
Invoke-WebRequest -Uri https://packages.wazuh.com/4.x/windows/wazuh-agent-4.7.5-1.msi -OutFile $env:tmp\wazuh-agent; msixexec.exe /I $env:tmp\wazuh-agent /q WAZUH_MANAGER='172.31.3.251'
```

Connexion au EC2 WINDOWS_CLIENT via RDP

- On ouvre cmd+r et on tape la commande : mstsc

The dashboard shows 1 active agent (Linux-Client). A 'Remote Desktop Connection' window is open, showing connection details for a Linux-Client with IP 34.229.165.166 and user Administrator. The RDP window also has 'Save' and 'Open...' buttons for connection settings.

aws Search [Alt+S] Ask Amazon Q Account ID: 5683-3603-9365
 United States (N. Virginia) v vocabs/user4619820=Chaimae_Mouhsine

EC2 Instances I-06488a88591728d16 Connect to instance

Session Manager RDP client EC2 serial console

Record RDP connections You can now record RDP connections using AWS Systems Manager just-in-time node access. Learn more Try for free

Instance ID I-06488a88591728d16 (Windows-Client)

Connection Type
 Connect using RDP client
 Download a file to use with your RDP client and retrieve your password.
 Connect using Fleet Manager
 To connect to the instance using Fleet Manager Remote Desktop, the SSM Agent must be installed and running on the instance. For more information, see Working with SSM Agent

You can connect to your Windows instance using a remote desktop client of your choice, and by downloading and running the RDP shortcut file below:

[Download remote desktop file](#)

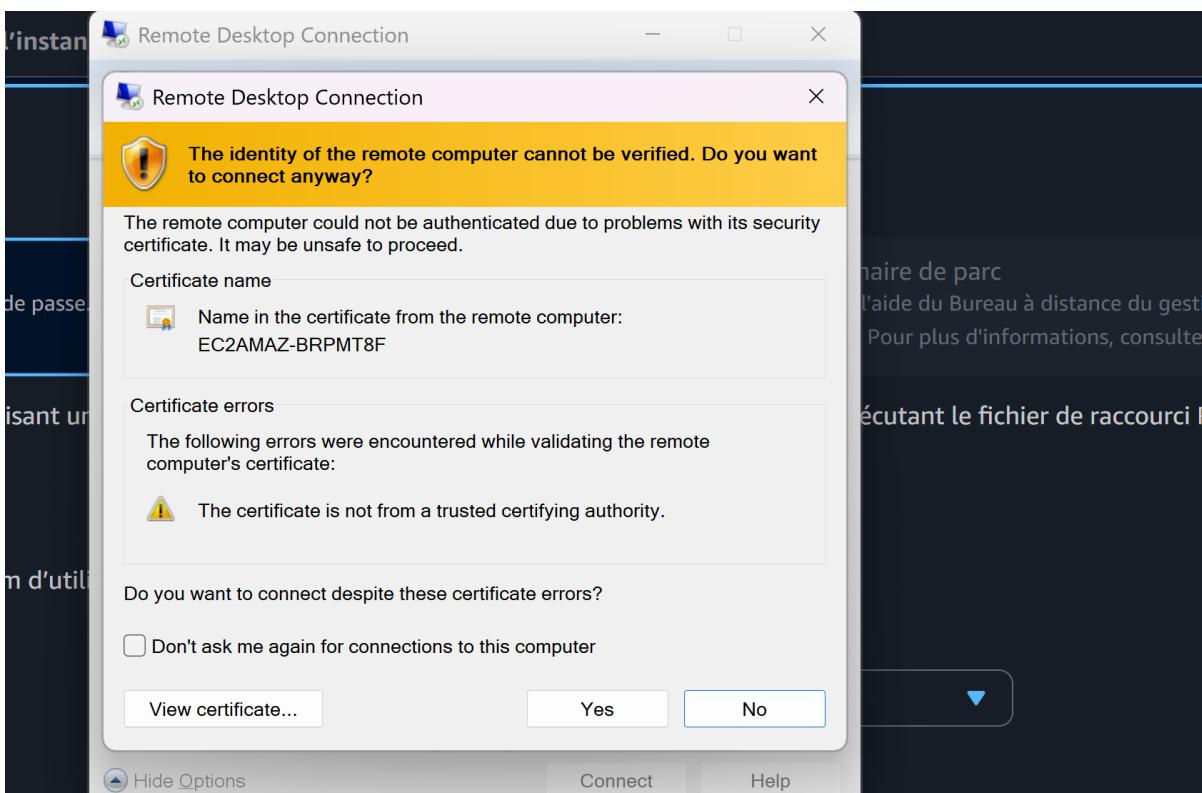
When prompted, connect to your instance using the following username and password:

Public IP 50.17.106.163 Username Info Administrator

Password 6gS7*;kD+taJ-%?B2rq%*msLjd9IN=

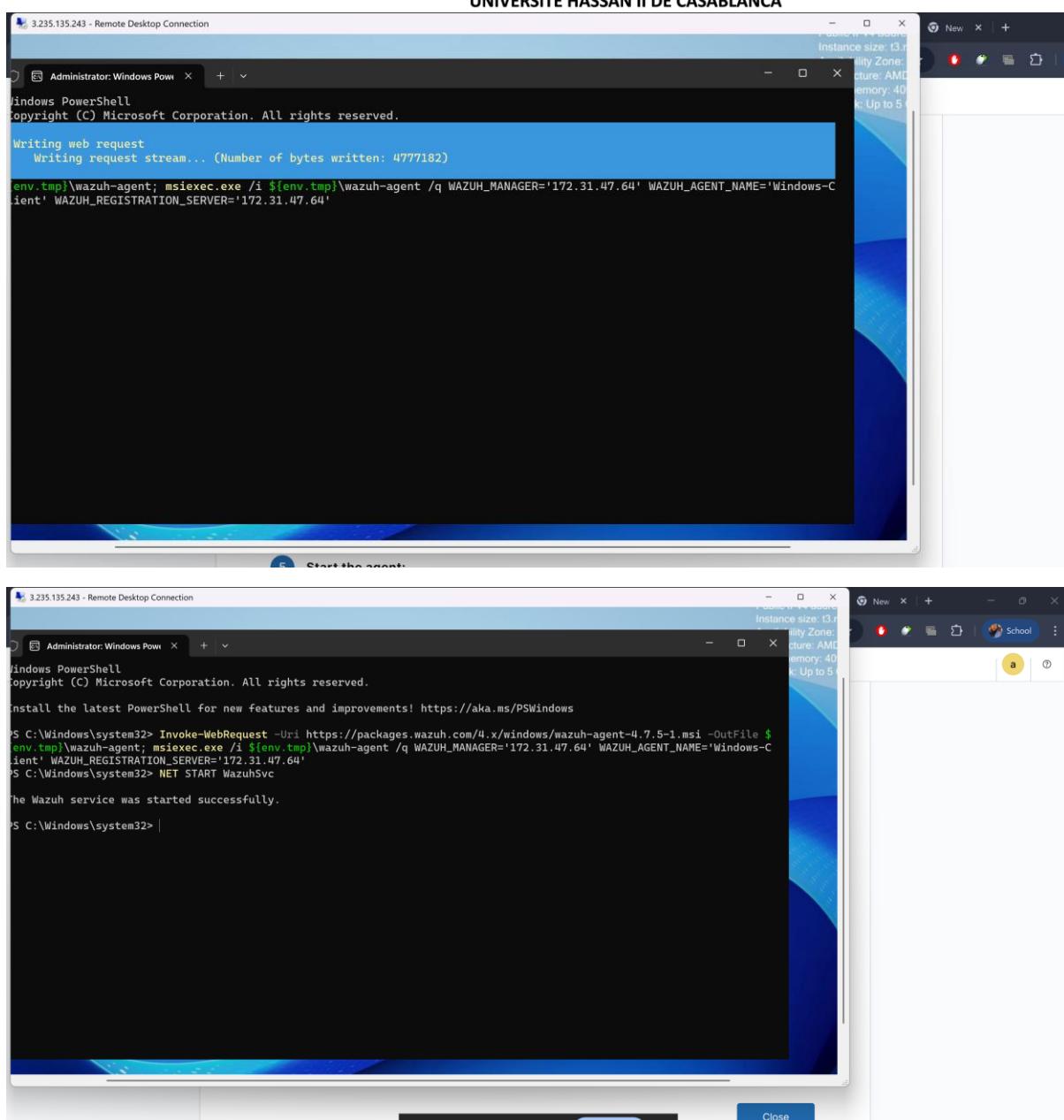
If you've joined your instance to a directory, you can use your directory credentials to connect to your instance

CloudShell Feedback © 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences



Passer dans powershell :

- Installation agent depuis dashboard (PowerShell)





Administrator: Windows Pow

```
PS C:\Windows\system32> Invoke-WebRequest -Uri https://packages.wazuh.com/4.x/windows/wazuh-agent-4.7.5-1.msi -Outfile $env.tmp\wazuh-agent; msixec.exe /i $env.tmp\wazuh-agent /q WAZUH_MANAGER='10.0.0.160' WAZUH_AGENT_NAME='Windows-Client' WAZUH_REGISTRATION_SERVER='10.0.0.160'
PS C:\Windows\system32> NET START WazuhSvc
The Wazuh service is starting.
The Wazuh service was started successfully.

PS C:\Windows\system32>
```

Hostname: EC2AM
Instance ID: i-0648
Private IPv4 address:
Public IPv4 address:
Instance size: t3.micro
Availability Zone: us-east-1a
Architecture: AMD64
Total memory: 4096 MB
Network: Up to 5 Gbps

EVOLUTION Last 24 hours

No results found

refresh Export formatted Version Status Actions WQL Refresh

Version	Status	Actions
v4.7.5	active	Details
v4.7.5	active	Details

wazuh. Agents

STATUS

- Active (2)
- Disconnected (0)
- Pending (0)
- Never connected (0)

DETAILS

Active	Disconnected	Pending	Never connected	Agents coverage
2	0	0	0	100.00%

Last registered agent: Windows-Client

Most active agent: Linux-Client

EVOLUTION

Last 24 hours

No results found

Agents (2)

id!=00 and

WQL Refresh Export formatted

ID	Name	IP address	Group(s)	Operating system	Cluster node	Version	Status	Actions	
001	Linux-Client	10.0.0.178	default	Ubuntu 24.04.3 LTS	node01	v4.7.5	● active		
002	Windows-Client	10.0.0.169	default	Microsoft Windows Server 2025 Datacenter 10.0.26100.7462	node01	v4.7.5	● active		

Rows per page: 10 < 1 >

L'enrôlement centralisé facilite la gestion des endpoints à grande échelle.

9. Scénarios de démonstration SIEM & EDR

9.1 Côté Linux

9.1.1 Scénario 1 — Tentatives SSH échouées (bruteforce simulé)

Sur Linux-Client (ou depuis une autre machine), fais plusieurs tentatives de login SSH invalides :

ssh fakeuse@10.0.0.178

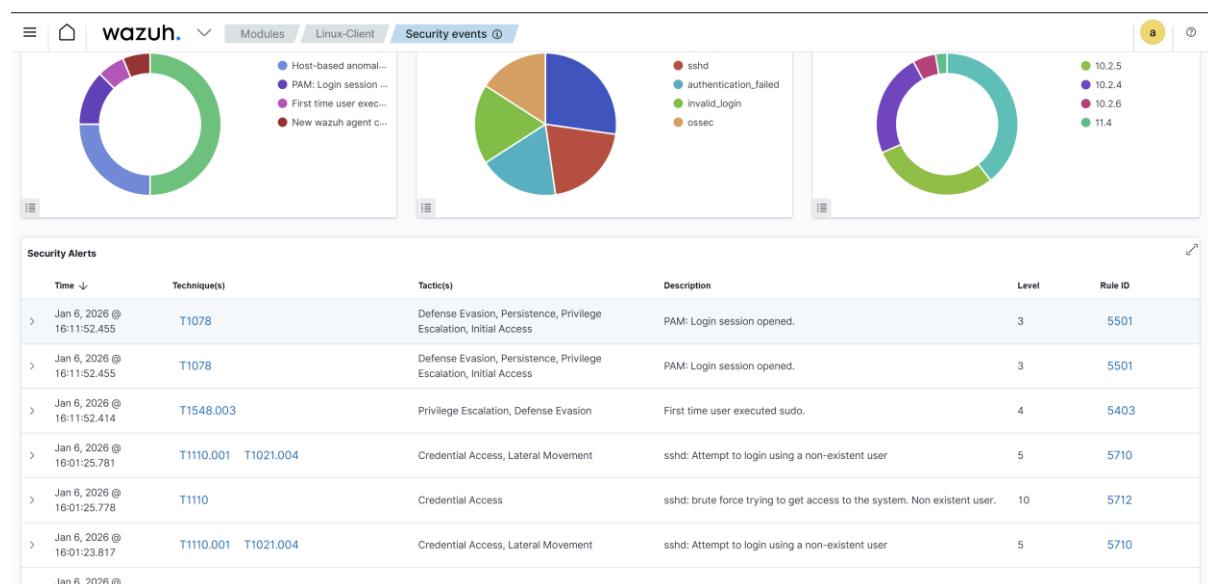
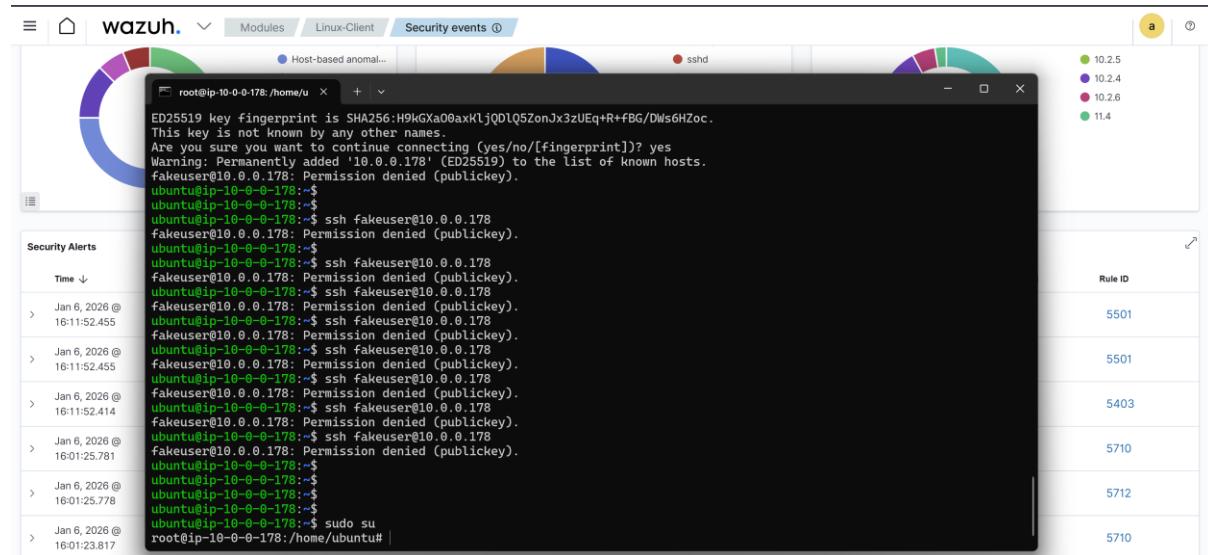
> Jan 6, 2026 @ 16:01:25.781	06/0: ubuntu@ip-10-0-0-178:~\$	5712
>	06/0: ubuntu@ip-10-0-0-178:~\$ ssh fakeuser@10.0.0.178	
> Jan 6, 2026 @ 16:01:23.817	06/0: The authenticity of host '10.0.0.178 (10.0.0.178)' can't be established.	5710
>	06/0: ED25519 key fingerprint is SHA256:HQ9Gxa00axKlJQDLQ5ZonJx3zUEq+FBG/DW5s6HZoc.	
> Jan 6, 2026 @ 16:01:23.767	06/0: This key is not known by any other names.	5710
>	06/0: Are you sure you want to continue connecting (yes/no/[fingerprint])? yes	
> Jan 6, 2026 @ 16:01:23.767	06/0: Warning: Permanently added '10.0.0.178' (ED25519) to the list of known hosts.	5710
>	06/0: fakeuser@10.0.0.178: Permission denied (publickey).	
> Jan 6, 2026 @ 16:01:21.767	06/0: ubuntu@ip-10-0-0-178:~\$	5710
>	06/0: ubuntu@ip-10-0-0-178:~\$ ssh fakeuser@10.0.0.178	
> Jan 6, 2026 @ 16:01:21.778	06/0: fakeuser@10.0.0.178: Permission denied (publickey).	5710
>	06/0: ubuntu@ip-10-0-0-178:~\$ ssh fakeuser@10.0.0.178	
> Jan 6, 2026 @ 16:01:21.778	06/0: fakeuser@10.0.0.178: Permission denied (publickey).	5710
>	06/0: ubuntu@ip-10-0-0-178:~\$ ssh fakeuser@10.0.0.178	
> Jan 6, 2026 @ 16:01:17.778	06/0: fakeuser@10.0.0.178: Permission denied (publickey).	5710
>	06/0: ubuntu@ip-10-0-0-178:~\$ ssh fakeuser@10.0.0.178	
> Jan 6, 2026 @ 16:01:13.766	06/0: fakeuser@10.0.0.178: Permission denied (publickey).	5710
>	06/0: ubuntu@ip-10-0-0-178:~\$ ssh fakeuser@10.0.0.178	
> Jan 6, 2026 @ 15:46:54.584	06/0: fakeuser@10.0.0.178: Permission denied (publickey).	510
>	06/0: fakeuser@10.0.0.178: Permission denied (publickey).	
> Jan 6, 2026 @ 15:46:54.584	ubuntu@ip-10-0-0-178:~\$	

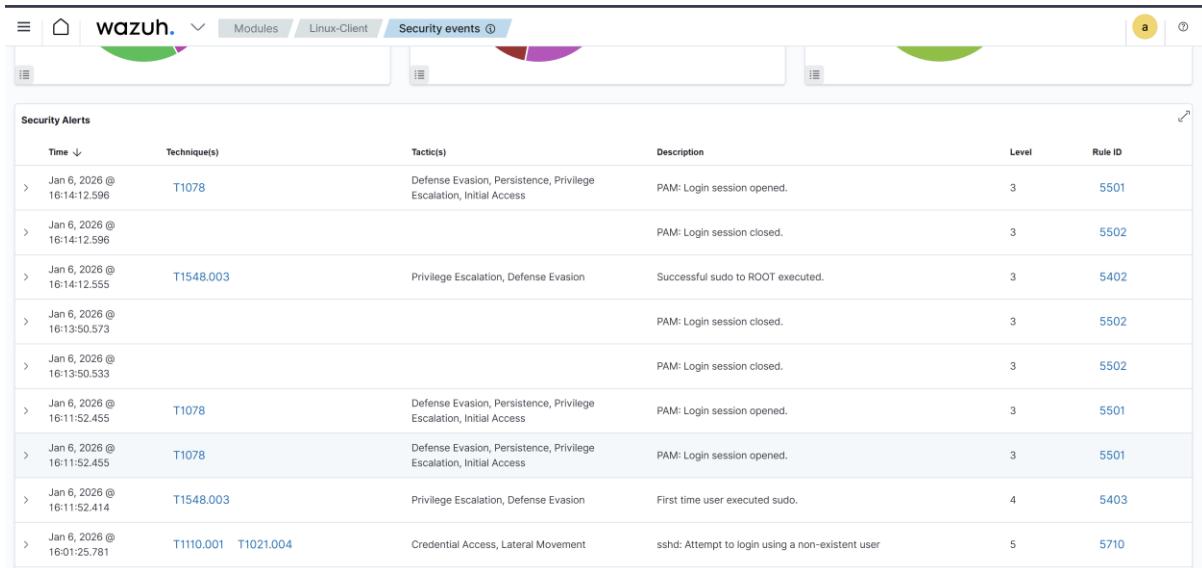
wazuh. Security events							①	
Security Alerts		Technique(s)			Tactic(s)	Description	Level	Rule ID
> Jan 6, 2026 @ 16:01:25.781	T1110.001 T1021.004	Credential Access, Lateral Movement	ssh: Attempt to login using a non-existent user	5	5710			
> Jan 6, 2026 @ 16:01:25.778	T1110	Credential Access	ssh: brute force trying to get access to the system. Non-existent user.	10	5712			
> Jan 6, 2026 @ 16:01:23.817	T1110.001 T1021.004	Credential Access, Lateral Movement	ssh: Attempt to login using a non-existent user	5	5710			
> Jan 6, 2026 @ 16:01:23.776	T1110.001 T1021.004	Credential Access, Lateral Movement	ssh: Attempt to login using a non-existent user	5	5710			
> Jan 6, 2026 @ 16:01:21.780	T1110.001 T1021.004	Credential Access, Lateral Movement	ssh: Attempt to login using a non-existent user	5	5710			
> Jan 6, 2026 @ 16:01:21.776	T1110.001 T1021.004	Credential Access, Lateral Movement	ssh: Attempt to login using a non-existent user	5	5710			
> Jan 6, 2026 @ 16:01:21.774	T1110.001 T1021.004	Credential Access, Lateral Movement	ssh: Attempt to login using a non-existent user	5	5710			
> Jan 6, 2026 @ 16:01:17.770	T1110.001 T1021.004	Credential Access, Lateral Movement	ssh: Attempt to login using a non-existent user	5	5710			
> Jan 6, 2026 @ 16:01:13.766	T1110.001 T1021.004	Credential Access, Lateral Movement	ssh: Attempt to login using a non-existent user	5	5710			
> Jan 6, 2026 @ 15:46:54.584			Host-based anomaly detection event (rootcheck).	7	510			

- Tentatives SSH échouées
- Élévation de privilège
- Modification de fichiers sensibles (FIM)

9.1.2 Scénario 2 — Élévation de priviléges

sudo su





The screenshot shows the Wazuh security events dashboard. At the top, there are tabs for 'Modules', 'Linux-Client', and 'Security events'. The 'Security events' tab is active. Below the tabs, there is a search bar and a date range selector. The main area displays a table of security alerts:

Time	Technique(s)	Tactic(s)	Description	Level	Rule ID
> Jan 6, 2026 @ 16:14:12.596	T1078	Defense Evasion, Persistence, Privilege Escalation, Initial Access	PAM: Login session opened.	3	5501
> Jan 6, 2026 @ 16:14:12.596			PAM: Login session closed.	3	5502
> Jan 6, 2026 @ 16:14:12.555	T1548.003	Privilege Escalation, Defense Evasion	Successful sudo to ROOT executed.	3	5402
> Jan 6, 2026 @ 16:13:50.573			PAM: Login session closed.	3	5502
> Jan 6, 2026 @ 16:13:50.533			PAM: Login session closed.	3	5502
> Jan 6, 2026 @ 16:11:52.455	T1078	Defense Evasion, Persistence, Privilege Escalation, Initial Access	PAM: Login session opened.	3	5501
> Jan 6, 2026 @ 16:11:52.455	T1078	Defense Evasion, Persistence, Privilege Escalation, Initial Access	PAM: Login session opened.	3	5501
> Jan 6, 2026 @ 16:11:52.414	T1548.003	Privilege Escalation, Defense Evasion	First time user executed sudo.	4	5403
> Jan 6, 2026 @ 16:01:25.781	T1110.001 T1021.004	Credential Access, Lateral Movement	ssh: Attempt to login using a non-existent user	5	5710

9.1.3 Scénario 3 — Modification fichier sensible

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

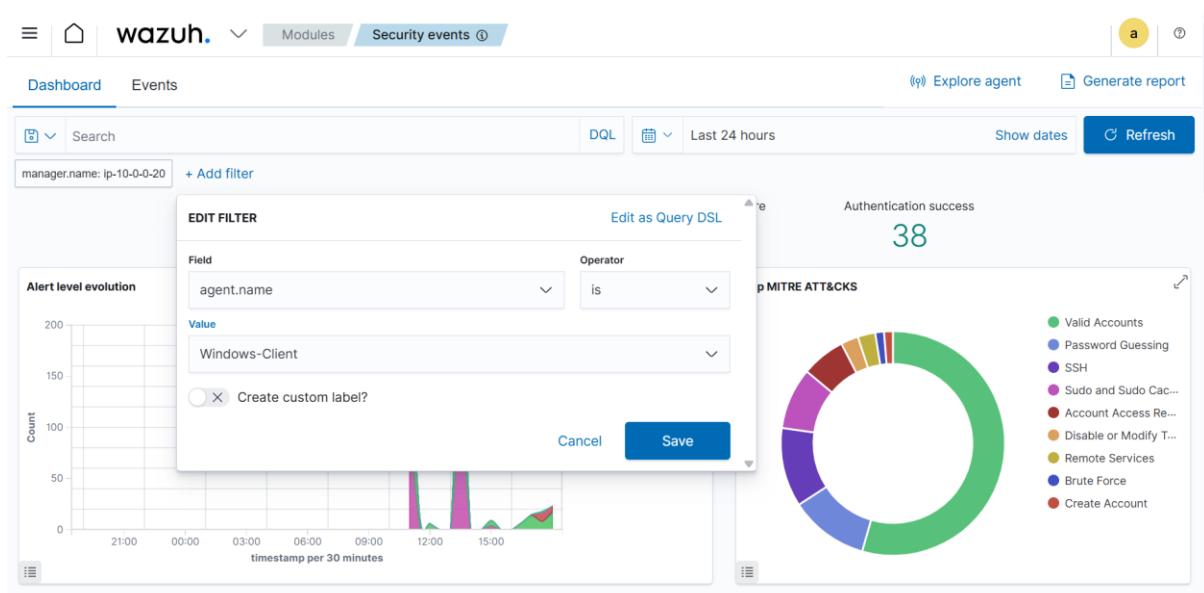
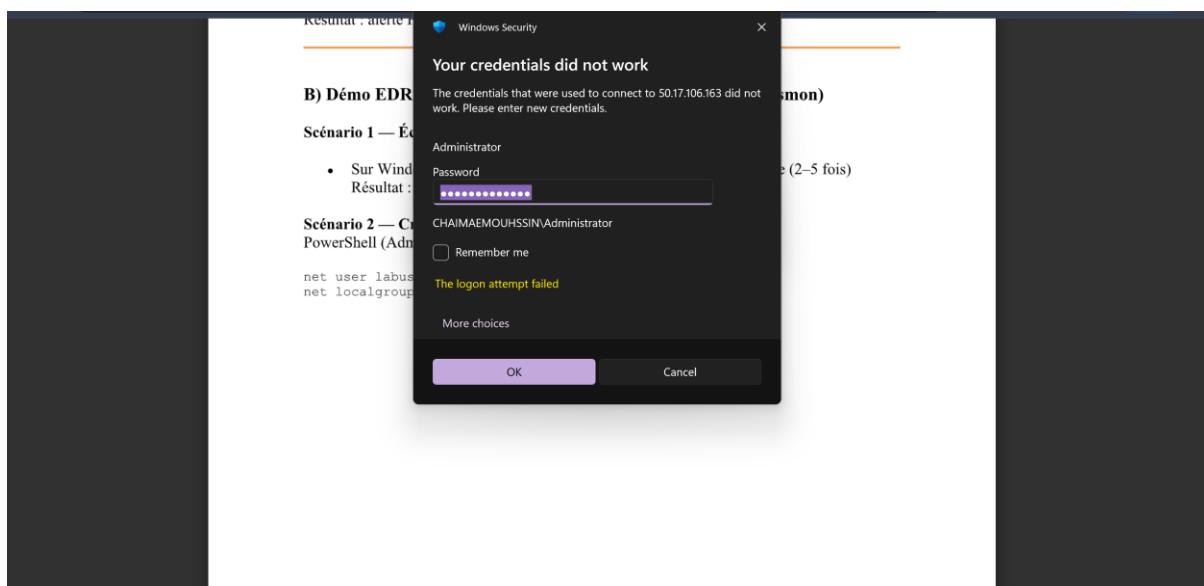
```
root@ip-10-0-0-178:/home/ubuntu# exit
ubuntu@ip-10-0-0-178:~$ echo "test" | sudo tee -a /etc/passwd
test
ubuntu@ip-10-0-0-178:~$ |
```

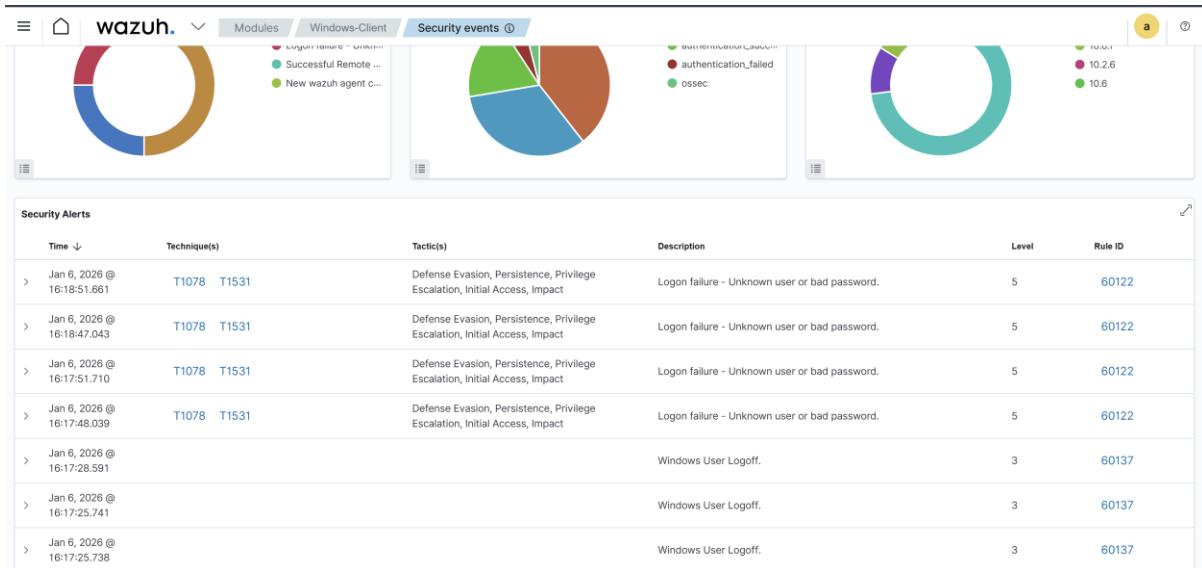
> Jan 6, 2026 @ 16:11:52.455	T1078	Defense Evasion, Persistence, Privilege Escalation, Initial Access	PAM: Login session opened.	3	5501
> Jan 6, 2026 @ 16:11:52.455	T1078	Defense Evasion, Persistence, Privilege Escalation, Initial Access	PAM: Login session opened.	3	5501
> Jan 6, 2026 @ 16:11:52.414	T1548.003	Privilege Escalation, Defense Evasion	First time user executed sudo.	4	5403

9.2 Scénarios côté Windows

9.2.1 Scénario 1 — Échecs de login

Sur Windows : on réalise des connexions RDP avec mauvais mot de passe

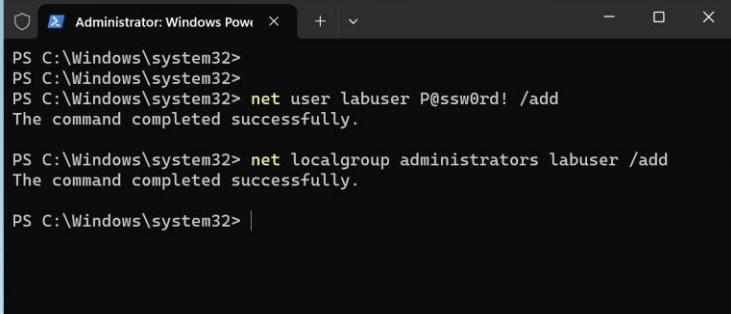




9.2.2 Scénario 2 — Création d'un utilisateur local

PowerShell (Admin) :

- net user labuser P@ssw0rd! /add
- net localgroup administrators labuser /add



```

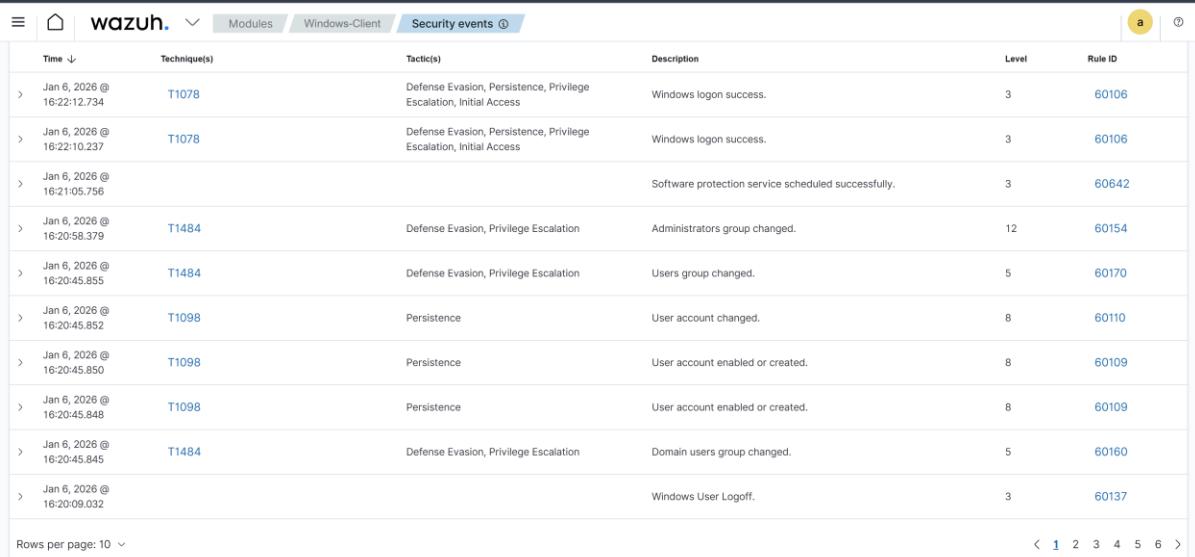
Administrator: Windows Pow
PS C:\Windows\system32>
PS C:\Windows\system32>
PS C:\Windows\system32> net user labuser P@ssw0rd! /add
The command completed successfully.

PS C:\Windows\system32> net localgroup administrators labuser /add
The command completed successfully.

PS C:\Windows\system32>

```

Hostname: EC2AMAZ-Q9MP463
 Instance ID: i-06488a88591728d16
 Private IPv4 address: 10.0.0.169
 Public IPv4 address: 50.17.106.163
 Instance size: t3.medium
 Availability Zone: us-east-1a
 Architecture: AMD64
 Total memory: 4096 MB
 Network: Up to 5 Gigabit



Time	Technique(s)	Tactic(s)	Description	Level	Rule ID
> Jan 6, 2026 @ 16:22:12.734	T1078	Defense Evasion, Persistence, Privilege Escalation, Initial Access	Windows logon success.	3	60106
> Jan 6, 2026 @ 16:22:10.237	T1078	Defense Evasion, Persistence, Privilege Escalation, Initial Access	Windows logon success.	3	60106
> Jan 6, 2026 @ 16:21:05.756			Software protection service scheduled successfully.	3	60642
> Jan 6, 2026 @ 16:20:58.379	T1484	Defense Evasion, Privilege Escalation	Administrators group changed.	12	60154
> Jan 6, 2026 @ 16:20:45.855	T1484	Defense Evasion, Privilege Escalation	Users group changed.	5	60170
> Jan 6, 2026 @ 16:20:45.852	T1098	Persistence	User account changed.	8	60110
> Jan 6, 2026 @ 16:20:45.850	T1098	Persistence	User account enabled or created.	8	60109
> Jan 6, 2026 @ 16:20:45.848	T1098	Persistence	User account enabled or created.	8	60109
> Jan 6, 2026 @ 16:20:45.845	T1484	Defense Evasion, Privilege Escalation	Domain users group changed.	5	60160
> Jan 6, 2026 @ 16:20:09.032			Windows User Logoff.	3	60137

9.2.3 Scénario 3— Option “EDR plus riche”: installer Sysmon

Installation Sysmon

- Invoke-WebRequest -Uri <https://download.sysinternals.com/files/Sysmon.zip> -OutFile C:\Sysmon.zip
- Expand-Archive -Path C:\Sysmon.zip -DestinationPath C:\Sysmon

50.17.106.163 - Remote Desktop Connection

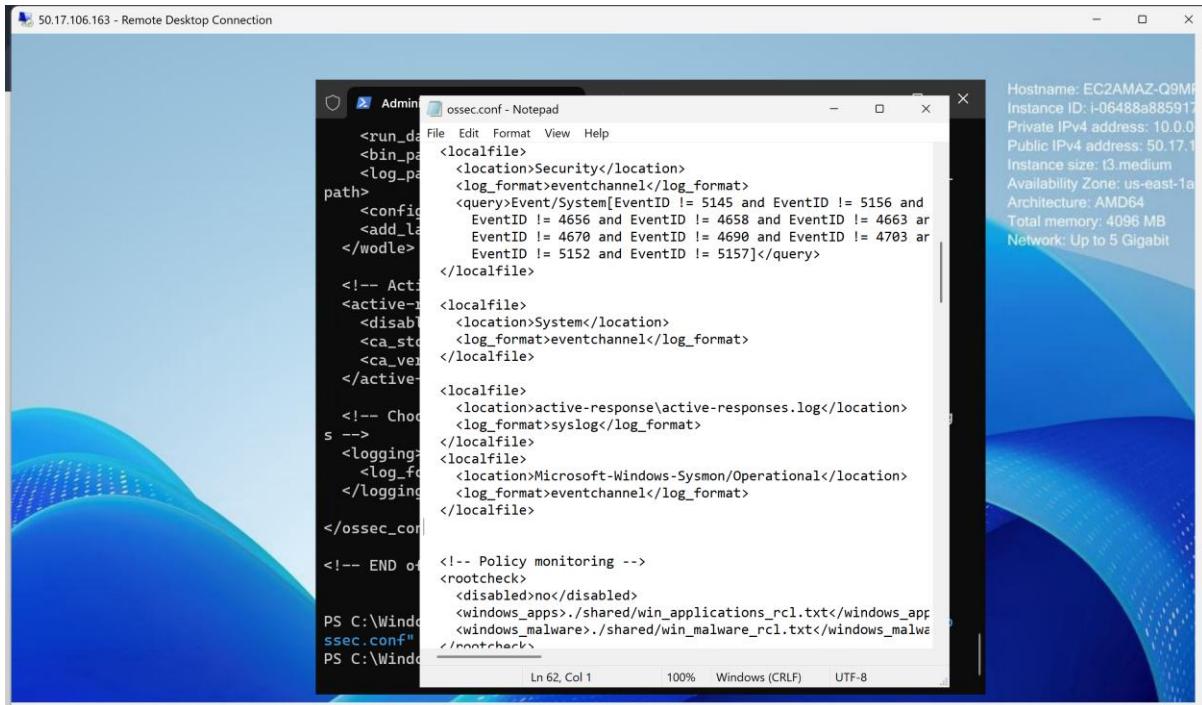
```
Administrator: Windows Pow x + v - □ ×  
PS C:\Windows\system32>  
PS C:\Windows\system32>  
PS C:\Windows\system32> net user labuser P@ssw0rd! /add  
The command completed successfully.  
  
PS C:\Windows\system32> net localgroup administrators labuser /add  
The command completed successfully.  
  
PS C:\Windows\system32> net user labuser P@ssw0rd! /add  
The account already exists.  
  
More help is available by typing NET HELPMSG 2224.  
  
PS C:\Windows\system32> net localgroup administrators labuser /add  
System error 1378 has occurred.  
  
The specified account name is already a member of the group.  
  
PS C:\Windows\system32> Invoke-WebRequest `>> -Uri https://download.sysinternals.com/files/Sysmon.zip `>> -OutFile $env:TEMP\Sysmon.zip  
PS C:\Windows\system32> Expand-Archive $env:TEMP\Sysmon.zip -DestinationPath C:\Sysmon  
PS C:\Windows\system32> Invoke-WebRequest `>> -Uri https://raw.githubusercontent.com/SwiftOnSecurity/sysmon-co`nfig/master/sysmonconfig-export.xml `>> -OutFile C:\Sysmon\sysmonconfig.xml  
PS C:\Windows\system32> |
```

50.17.106.163 - Remote Desktop Connection

```
Administrator: Windows Pow x + v - □ ×  
nfig/master/sysmonconfig-export.xml `>> -OutFile C:\Sysmon\sysmonconfig.xml  
PS C:\Windows\system32> C:\Sysmon\Sysmon64.exe -accepteula -i C:\Sys`on\sysmonconfig.xml  
  
System Monitor v15.15 - System activity monitor  
By Mark Russinovich and Thomas Garnier  
Copyright (C) 2014-2024 Microsoft Corporation  
Using libxml2. libxml2 is Copyright (C) 1998-2012 Daniel Veillard. Al`l Rights Reserved.  
Sysinternals - www.sysinternals.com  
  
Loading configuration file with schema version 4.50  
Sysmon schema version: 4.90  
Configuration file validated.  
Sysmon64 installed.  
SysmonDrv installed.  
Starting SysmonDrv.  
SysmonDrv started.  
Starting Sysmon64..  
Sysmon64 started.  
PS C:\Windows\system32> Get-Service Sysmon64  


| Status  | Name     | DisplayName |
|---------|----------|-------------|
| Running | Sysmon64 | Sysmon64    |

  
PS C:\Windows\system32> |
```



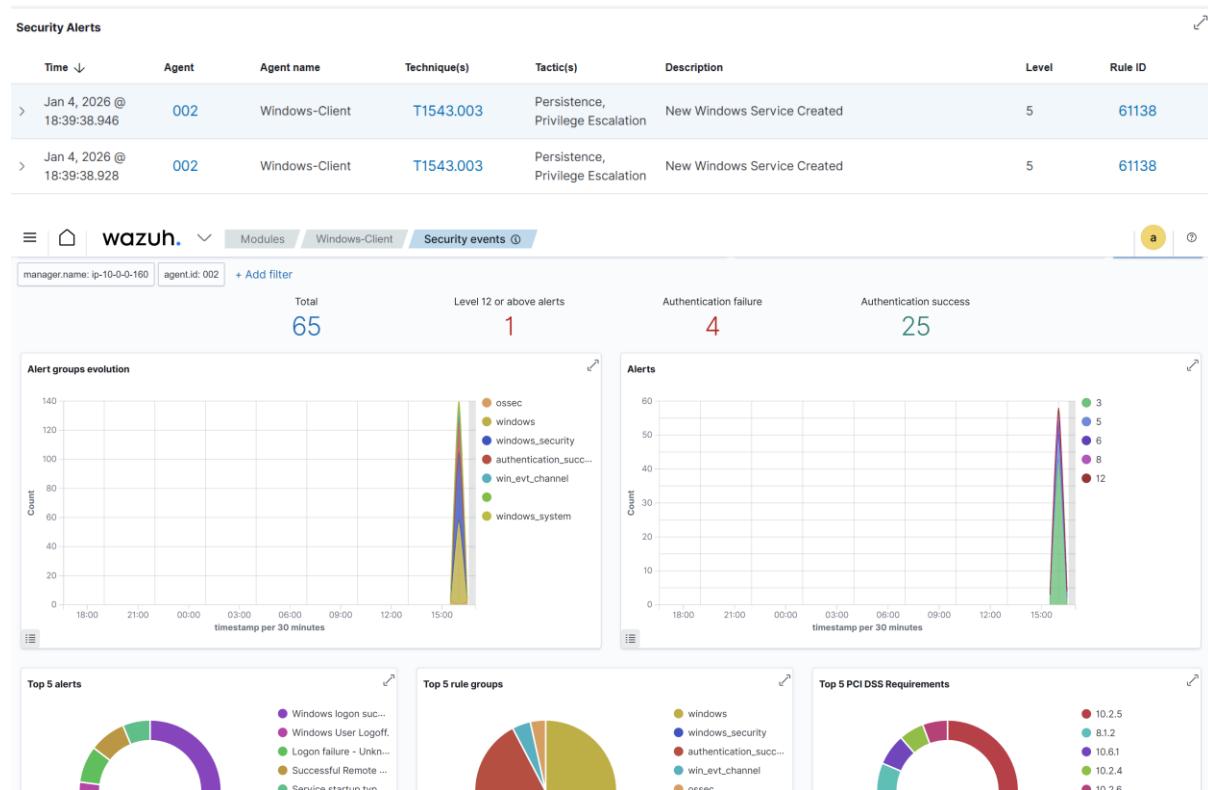
```

<run_daemon>
  <localfile>
    <location>Security</location>
    <log_format>eventchannel</log_format>
    <query>Event/System[EventID != 5145 and EventID != 5156 and EventID != 4656 and EventID != 4658 and EventID != 4663 and EventID != 4670 and EventID != 4690 and EventID != 4703 and EventID != 5152 and EventID != 5157]</query>
  </localfile>
  <!-- Active responses -->
  <active-response>
    <disabled>no</disabled>
    <ca_stop>
      <ca_version>1.0</ca_version>
    </ca_stop>
  </active-response>
  <!-- Chosen logs -->
  <logging>
    <log_file>
      <location>active-response\active-responses.log</location>
      <log_format>syslog</log_format>
    </log_file>
    <localfile>
      <location>Microsoft-Windows-Sysmon/Operational</location>
      <log_format>eventchannel</log_format>
    </localfile>
  </logging>
</ossec_conf>
<!-- END of configuration -->
<rootcheck>
  <disabled>no</disabled>
  <windows_apps>./shared/win_applications_rcl.txt</windows_apps>
  <windows_malware>./shared/win_malware_rcl.txt</windows_malware>
</rootcheck>

```

Ces scénarios démontrent la capacité de Wazuh à détecter des comportements suspects et à générer des alertes pertinentes.

10. Visualisation dans le Dashboard Wazuh



11. Conclusion

Cet atelier a permis de mettre en œuvre une plateforme SIEM & EDR complète, intégralement conçue et configurée en détail, depuis la création du réseau Cloud jusqu'à l'analyse avancée des événements de sécurité.

La création d'un VPC personnalisée, le déploiement multi-OS et la configuration fine des règles de sécurité illustrent une démarche professionnelle conforme aux standards des infrastructures modernes.

Wazuh s'impose comme une solution robuste, capable de fournir une vision centralisée, proactive et corrélée des événements de sécurité, renforçant ainsi significativement la posture de sécurité globale.



Webographie

<https://github.com/CHAIMAEMOUHSSINE/Security-Lab-Endpoints-Supervision-SIEM-Multi-OS-Linux-Windows-.git>