Beats

Stack

Windows 10
Professional

IMPLEMENTATION D'UN SIEM POUR GERER LES ALERTES DE MENACES AVEC WAZUH + ELK STACK

Présenté par :

Josias KONAN

INTRODUCTION

Aujourd'hui, les cyberattaques sont devenues une véritable menace pour les entreprises et les infrastructures informatiques. Les systèmes génèrent une énorme quantité de logs, mais sans un bon outil pour les analyser, ces données restent inutilisables. C'est là qu'intervient un **SIEM (Security Information and Event Management)**, une solution permettant de **centraliser, analyser et corréler** les événements de sécurité pour détecter rapidement les menaces et réagir efficacement.

Pourquoi ce projet ?

L'objectif de ce projet est de **mettre en place un SIEM basé sur Wazuh et la stack ELK** afin d'avoir une solution de surveillance en temps réel des événements de sécurité. L'idée est d'automatiser la détection des incidents, de générer des alertes pertinentes et de proposer une meilleure visibilité sur l'état de la sécurité d'un système d'information.

Technologies utilisées

- **Wazuh** : Une solution open-source qui fait office de SIEM et d'IDS (Intrusion Detection System). Il collecte et analyse les logs pour détecter les menaces et les activités suspectes.
- **ELK Stack** (Elasticsearch, Filebeat, Kibana):
    - **Elasticsearch** : Un moteur de recherche et d'indexation qui permet de stocker et d'interroger efficacement les logs.
    - **Filebeat** : Un agent léger qui collecte les logs directement depuis les serveurs et les envoie vers Elasticsearch sans nécessiter un traitement intermédiaire. Il est plus simple et plus performant que Logstash pour certaines architectures.
    - **Kibana** : Un outil de visualisation qui permet d'explorer les logs et de créer des tableaux de bord interactifs pour mieux comprendre les événements de sécurité.

Avantages d'une telle solution

✅ **Centralisation des logs** : Tous les événements de sécurité sont regroupés en un seul endroit.
✅ **Détection proactive des menaces** : Analyse automatique des logs pour repérer les comportements suspects.
✅ **Visualisation intuitive** : Grâce à Kibana, les données deviennent exploitables sous forme de graphiques et tableaux de bord.
✅ **Réduction du temps de réponse aux incidents** : Les alertes permettent d'agir rapidement en cas d'attaque.
✅ **Flexibilité et open-source** : Pas de coûts de licence élevés, et possibilité de personnalisation selon les besoins.
✅ **Optimisation des performances** : Avec **Filebeat**, la collecte des logs est plus légère et plus efficace qu'avec Logstash, ce qui améliore la réactivité du SIEM.

C'est cette solution que nous tenterons de **mettre en place tout au long de notre travail**, en explorant les différentes étapes de son déploiement et en évaluant son efficacité dans la gestion des menaces de sécurité.

I. PRÉRÉQUIS ET INITIALISATION DU PROJET

Environment:

- *1 Machine sous Ubuntu 22.04.5 LTS*
- *1 Machine sous Windows 10 Professionnel*
- *1 Machine sous Windows 11 Professionnel*
- *VMWare Workstation Pro 17*

Prérequis du projet:

-

Adressage Réseau:

N°	HÔTE	RÔLE	ADRESSE IP	ENVIRONNEMENT
1	Wazuh	Serveur Wazuh	192.168.1.114/24	Ubuntu 22.04.5 LTS
2	Client 1	Machine Cliente 1	192.168.1.65/24	Windows 11 Professionnel
3	Client 2	Machine Cliente 2	192.168.1.130/24	Windows 10 Professionnel

II. INSTALLATION ET CONFIGURATION DE BASE DU SERVEUR WAZUH

Dans cette partie, nous procéderons à l'installation des prérequis pour notre serveur Wazuh, de l'installation et la configuration de base du service « ***Elasticsearch*** » jusqu'à l'installation du service « ***Wazuh-manager*** »

1. Installation de Elastic Stack

- Nous allons nous connecter par ssh à notre serveur Wazuh depuis notre ligne de commande Windows. La première chose à faire sera de mettre à jour le système Ubuntu et les paquets si cela n'est pas déjà fait, en tapant la commande « ***sudo apt update && sudo apt upgrade -y*** ». Après quoi, nous allons installer les paquets requis pour l'élaboration de notre projet.

```
apt-get install apt-transport-https zip unzip lsb-release curl gnupg
```

- Après l'installation des prérequis, nous allons débuter l'installation de « ***Elastic Stack*** ». Pour cela, nous allons commencer par ajouter la « ***GPG Key*** » d'Elastic Stack qui est centré sur « ***elasticsearch*** » avec la commande : « ***curl -s https://artifacts.elastic.co/GPG-KEY-elasticsearch | gpg --no-default-keyring --keyring gnupg-ring:/usr/share/keyrings/elasticsearch.gpg --import*** » et accorder les permissions necéssaires au fichier avec cette commande : « ***chmod 644 /usr/share/keyrings/elasticsearch.gpg*** »

```
root@wazuh-server:/home/adminwazuh# curl -s https://artifacts.elastic.co/GPG-KEY-elasticsearch | gpg --no-default-keyrin
g --keyring gnupg-ring:/usr/share/keyrings/elasticsearch.gpg --import && chmod 644 /usr/share/keyrings/elasticsearch.gpg
gpg: keyring '/usr/share/keyrings/elasticsearch.gpg' created
gpg: key D27D666CD88E42B4: public key "Elasticsearch (Elasticsearch Signing Key) <dev_ops@elasticsearch.org>" imported
gpg: Total number processed: 1
gpg:               imported: 1
root@wazuh-server:/home/adminwazuh#
```

- Nous allons maintenant ajouter le repo d'Elasticsearch en tapant cette « ***echo "deb [signed-by=/usr/share/keyrings/elasticsearch.gpg] https://artifacts.elastic.co/packages/7.x/apt stable main" | tee /etc/apt/sources.list.d/elastic-7.x.list*** »

```
root@wazuh-server:/home/adminwazuh# echo "deb [signed-by=/usr/share/keyrings/elasticsearch.gpg] https://artifacts.elasti
c.co/packages/7.x/apt stable main" | tee /etc/apt/sources.list.d/elastic-7.x.list
deb [signed-by=/usr/share/keyrings/elasticsearch.gpg] https://artifacts.elastic.co/packages/7.x/apt stable main
root@wazuh-server:/home/adminwazuh#
```

- Nous allons mettre à jour les paquets d'Elasticsearch pour charger les dernières mises à jour et informations relative à elasticsearch avec cette commande : « ***apt-get update*** »

```
root@wazuh-server:/home/adminwazuh# apt-get update
Atteint :1 http://archive.ubuntu.com/ubuntu jammy InRelease
Réception de :2 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
Réception de :3 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Réception de :4 https://artifacts.elastic.co/packages/7.x/apt stable InRelease [13,7 kB]
Atteint :5 https://packages.wazuh.com/4.x/apt stable InRelease
Atteint :6 http://archive.ubuntu.com/ubuntu jammy-backports InRelease
Réception de :7 https://artifacts.elastic.co/packages/7.x/apt stable/main amd64 Packages [143 kB]
Réception de :8 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [2 390 kB]
2 803 ko réceptionnés en 5s (532 ko/s)
Lecture des listes de paquets… 76%
```

A présent, nous avons terminer les configurations de base pour l'installation de notre « ***Elastic Stack*** », nous allons maintenant débuter l'installation de « ***Elasticsearch*** ».

2. Installation d'Elasticsearch

- Nous allons télécharger le paquet d'installation d'« ***Elsaticsearch*** » en tapant cette commande : « ***apt-get install elasticsearch=7.17.13*** »

```
root@wazuh-server:/home/adminwazuh# apt-get install elasticsearch=7.17.13
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les NOUVEAUX paquets suivants seront installés :
  elasticsearch
0 mis à jour, 1 nouvellement installés, 0 à enlever et 3 non mis à jour.
Il est nécessaire de prendre 318 Mo dans les archives.
Après cette opération, 531 Mo d'espace disque supplémentaires seront utilisés.
Réception de :1 https://artifacts.elastic.co/packages/7.x/apt stable/main amd64 elasticsearch amd64 7.17.13 [318 MB]
0% [1 elasticsearch 1 188 kB/318 MB 0%]
```

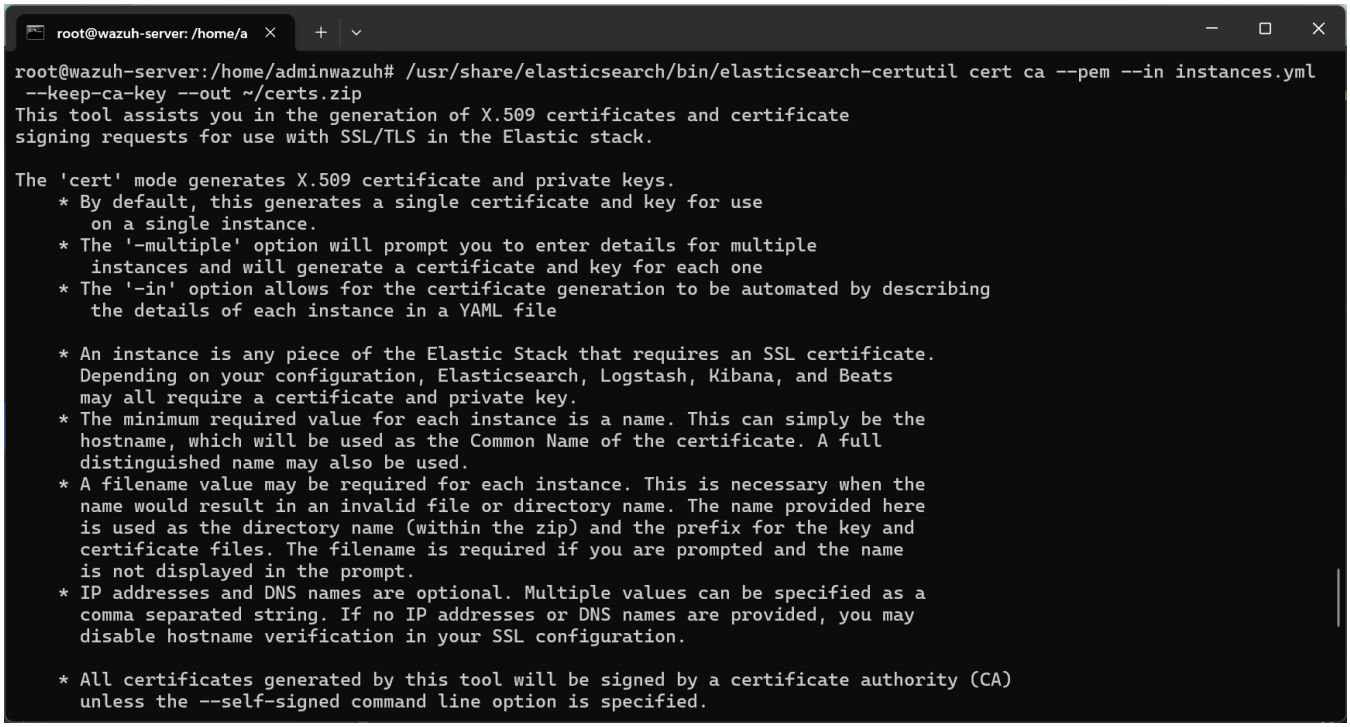
- Procédons au téléchargement du fichier de configuration et nous le stockerons dans le fichier « ***/etc/elasticserach/elasticsearch.yml*** » avec cette commande : « ***curl -so /etc/elasticsearch/elasticsearch.yml https://packages.wazuh.com/4.5/tpl/elastic-basic/elasticsearch_all_in_one.yml*** »

```
root@wazuh-server:/home/adminwazuh# curl -so /etc/elasticsearch/elasticsearch.yml https://packages.wazuh.com/4.5/tpl/ela
stic-basic/elasticsearch_all_in_one.yml
root@wazuh-server:/home/adminwazuh#
```

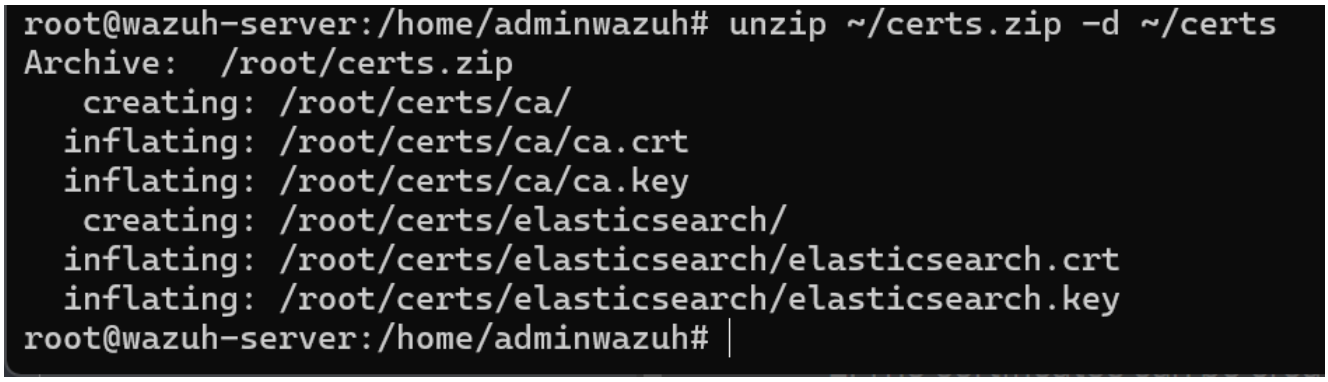
- A présent, nous allons télécharger le fichier de configuration qui servira à la création des certificats et il sera stocké dans ce fichier « ***/usr/share/elasticsearch/instances.yml*** » en exécutant la commande suivante : « ***curl -so /usr/share/elasticsearch/instances.yml https://packages.wazuh.com/4.5/tpl/elastic-basic/instances_aio.yml*** »

```
root@wazuh-server:/home/adminwazuh# curl -so /usr/share/elasticsearch/instances.yml https://packages.wazuh.com/4.5/tpl/e
lastic-basic/instances_aio.yml
root@wazuh-server:/home/adminwazuh#
```

- Nous allons maintenant créer des certificats en utilisant « ***elasticsearch-certutil-tool*** » avec la commande : « ***/usr/share/elasticsearch/bin/elasticsearch-certutil cert ca --pem --in instances.yml --keep-ca-key --out ~/certs.zip*** »



- Nous allons maintenant extraire le certificat généré qui a été créé dans « ***/usr/share/elasticsearch/certs.zip*** » en utilisant cette commande : « ***unzip ~/certs.zip -d ~/certs*** »



- A présent, nous procédons à la création d'un répertoire où nous allons copier le « ***CA File*** », le certificat ainsi que la clé. Nous exécuterons de manière successive ou simultannée ces commandes :

mkdir /etc/elasticsearch/certs/ca -p

cp -R ~/certs/ca/ ~/certs/elasticsearch/* /etc/elasticsearch/certs/

chown -R elasticsearch: /etc/elasticsearch/certs

chmod -R 500 /etc/elasticsearch/certs

chmod 400 /etc/elasticsearch/certs/ca/ca.* /etc/elasticsearch/certs/elasticsearch.*

rm -rf ~/certs/ ~/certs.zip

```
root@wazuh-server:/home/adminwazuh# mkdir /etc/elasticsearch/certs/ca -p
cp -R ~/certs/ca/ ~/certs/elasticsearch/* /etc/elasticsearch/certs/
chown -R elasticsearch: /etc/elasticsearch/certs
chmod -R 500 /etc/elasticsearch/certs
chmod 400 /etc/elasticsearch/certs/ca/ca.* /etc/elasticsearch/certs/elasticsearch.*
rm -rf ~/certs/ ~/certs.zip
root@wazuh-server:/home/adminwazuh#
```

- A présent, nous avons terminer l'installation de « ***Elasticsearch*** », nous allons activer et lancer le service en exécutant cette suite de commande :

systemctl daemon-reload

systemctl enable elasticsearch

systemctl start elasticsearch

systemctl status elasticseart

```
root@wazuh-server:/home/adminwazuh# systemctl daemon-reload
root@wazuh-server:/home/adminwazuh# systemctl start elasticsearch
root@wazuh-server:/home/adminwazuh# systemctl enable elasticsearch
Synchronizing state of elasticsearch.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable elasticsearch
root@wazuh-server:/home/adminwazuh# systemctl status elasticsearch
● elasticsearch.service - Elasticsearch
     Loaded: loaded (/lib/systemd/system/elasticsearch.service; enabled; vendor preset: enabled)
     Active: active (running) since Sat 2025-03-22 11:51:49 UTC; 27s ago
       Docs: https://www.elastic.co
   Main PID: 2151 (java)
      Tasks: 64 (limit: 4439)
     Memory: 2.2G
        CPU: 3min 5.886s
     CGroup: /system.slice/elasticsearch.service
             ├─2151 /usr/share/elasticsearch/jdk/bin/java -Xshare:auto -Des.networkaddress.cache.ttl=60 -Des.networkadd>
             └─2334 /usr/share/elasticsearch/modules/x-pack-ml/platform/linux-x86_64/bin/controller

mars 22 11:50:31 wazuh-server systemd[1]: Starting Elasticsearch...
mars 22 11:51:49 wazuh-server systemd[1]: Started Elasticsearch.
lines 1-14/14 (END)
```

- Nous allons générer maintenant les identifiants pour la « ***Elastic Stack*** » avec la commande suivante : « ***/usr/share/elasticsearch/bin/elasticsearch-setup-passwords auto*** » et nous tapons « ***y*** » pour valider l'opération.

```
root@wazuh-server:/home/adminwazuh# /usr/share/elasticsearch/bin/elasticsearch-setup-passwords auto
Initiating the setup of passwords for reserved users elastic,apm_system,kibana,kibana_system,logstash_system,beats_syste
m,remote_monitoring_user.
The passwords will be randomly generated and printed to the console.
Please confirm that you would like to continue [y/N]y


Changed password for user apm_system
PASSWORD apm_system = 9rYSTGXGMxuetVkNGJJO

Changed password for user kibana_system
PASSWORD kibana_system = DqY3jFtvSab6AvYC8Bla

Changed password for user kibana
PASSWORD kibana = DqY3jFtvSab6AvYC8Bla

Changed password for user logstash_system
PASSWORD logstash_system = SVfXRYgfEheaAk0CDaEC

Changed password for user beats_system
PASSWORD beats_system = sfXhK49vSI9ehpZXHgIi

Changed password for user remote_monitoring_user
PASSWORD remote_monitoring_user = ZaEFl4kL9nlp4e57lw44

Changed password for user elastic
PASSWORD elastic = 7YBFD0yI9bOO0V5RoFbj

root@wazuh-server:/home/adminwazuh#
```

- Nous allons maintenant vérifier que l'installation s'est passé correctement en utilisant la clé d' « ***Elasticsearch*** ». On exécute cette commande : « ***curl -XGET https://localhost:9200 -u elastic:<elastic_password> -k*** ».

<u>NB :</u> IL FAUT REMPLACER ''*<elastic_password>*'' PAR LA VRAI CLE GENEREE PRECEDEMENT. IL S'AGIRA DE LA CLE DE L'UTILISATEUR ''*elastic*''

```
root@wazuh-server:/home/adminwazuh# curl -XGET https://localhost:9200 -u elastic:7YBFD0yI9bOO0V5RoFbj -k
{
  "name" : "elasticsearch",
  "cluster_name" : "elasticsearch",
  "cluster_uuid" : "rHwcFZaYTeaj2xqf3NAN5Q",
  "version" : {
    "number" : "7.17.13",
    "build_flavor" : "default",
    "build_type" : "deb",
    "build_hash" : "2b211dbb8bfdecaf7f5b44d356bdfe54b1050c13",
    "build_date" : "2023-08-31T17:33:19.958690787Z",
    "build_snapshot" : false,
    "lucene_version" : "8.11.1",
    "minimum_wire_compatibility_version" : "6.8.0",
    "minimum_index_compatibility_version" : "6.0.0-beta1"
  },
  "tagline" : "You Know, for Search"
}
root@wazuh-server:/home/adminwazuh#
```

Nous avons terminé l'installation de « ***Elasticsearch*** », nous passerons maintenant à l'installation de « ***wazuh-manager*** » qui est l'interface d'administration du SIEM Wazuh.

3. Installation de Wazuh-manager

- Pour débuter l'installation de « ***wazuh-manager*** », nous allons d'abord installer la clé « ***GPG Key*** » de « ***wazuh*** » en exécutant cette commande : « ***curl -s https://packages.wazuh.com/key/GPG-KEY-WAZUH | gpg --no-default-keyring --keyring gnupg-ring:/usr/share/keyrings/wazuh.gpg --import && chmod 644 /usr/share/keyrings/wazuh.gpg*** »

```
root@wazuh-server:/home/adminwazuh# curl -s https://packages.wazuh.com/key/GPG-KEY-WAZUH | gpg --no-default-keyring --ke
yring gnupg-ring:/usr/share/keyrings/wazuh.gpg --import && chmod 644 /usr/share/keyrings/wazuh.gpg
gpg: key 96B3EE5F29111145: "Wazuh.com (Wazuh Signing Key) <support@wazuh.com>" not changed
gpg: Total number processed: 1
gpg:              unchanged: 1
root@wazuh-server:/home/adminwazuh#
```

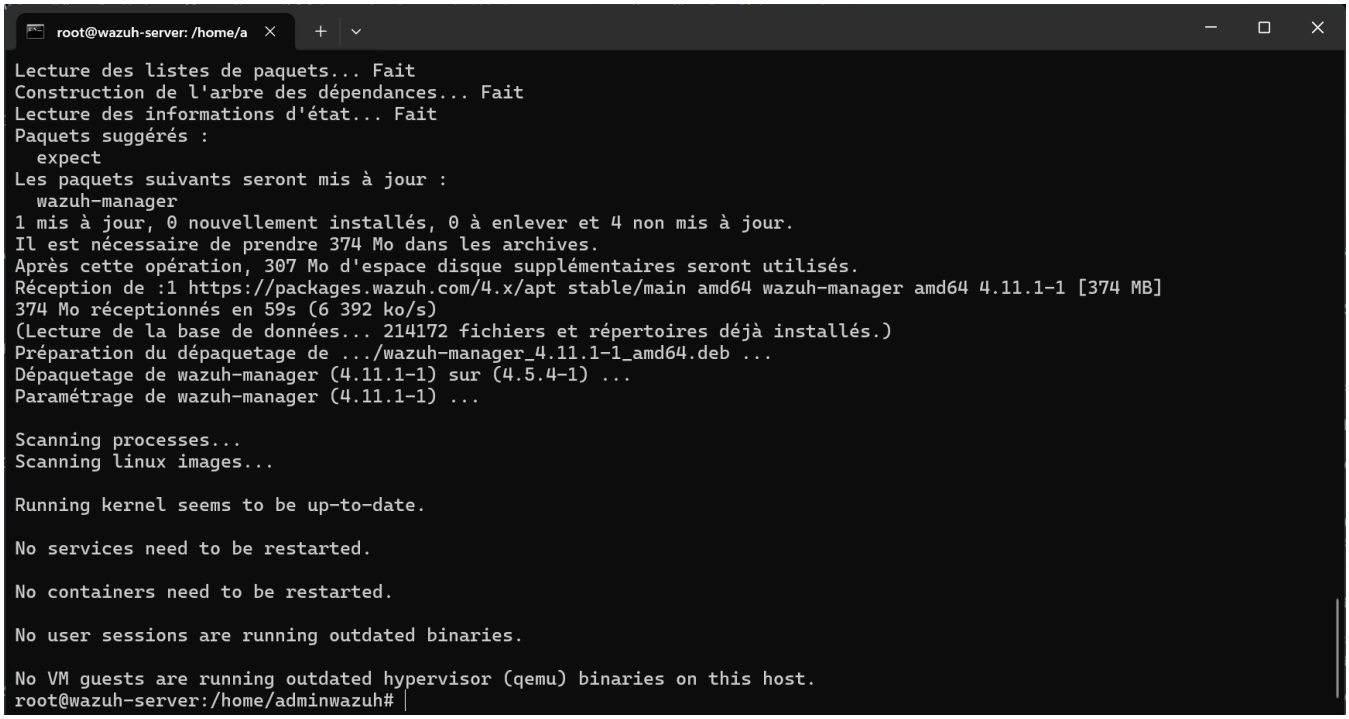
- Après avoir installé notre clé « ***GPG Key*** », nous allons ajouter le repo de « ***Wazuh Server*** » en tapant la commande qui suit : « ***echo "deb [signed-by=/usr/share/keyrings/wazuh.gpg] https://packages.wazuh.com/4.x/apt/ stable main" | tee -a /etc/apt/sources.list.d/wazuh.list*** »

```
root@wazuh-server:/home/adminwazuh# echo "deb [signed-by=/usr/share/keyrings/wazuh.gpg] https://packages.wazuh.com/4.x/a
pt/ stable main" | tee -a /etc/apt/sources.list.d/wazuh.list
deb [signed-by=/usr/share/keyrings/wazuh.gpg] https://packages.wazuh.com/4.x/apt/ stable main
root@wazuh-server:/home/adminwazuh#
```

- Il nous faut maintenant mettre à jour les informations de notre paquet « ***Wazuh Server*** » avec la commande : « ***apt-get update*** »

```
root@wazuh-server:/home/adminwazuh# apt-get update
Atteint :1 http://security.ubuntu.com/ubuntu jammy-security InRelease
Atteint :2 http://archive.ubuntu.com/ubuntu jammy InRelease
Atteint :3 https://packages.wazuh.com/4.x/apt stable InRelease
Atteint :4 http://archive.ubuntu.com/ubuntu jammy-updates InRelease
Atteint :5 https://artifacts.elastic.co/packages/7.x/apt stable InRelease
Atteint :6 http://archive.ubuntu.com/ubuntu jammy-backports InRelease
Lecture des listes de paquets... Fait
root@wazuh-server:/home/adminwazuh#
```

- A ce niveau, les prérequis sont installés. Nous allons maintenant pouvoir passer à l'installation de « ***Wazuh-manager*** ». Pour se faire, nous exécutons cette commande : « ***apt-get install wazuh-manager=4.5.4-1*** ». Après avoir lancé l'installation, nous aurons cette sortie qui nous montre que l'installation de notre paquet « ***wazuh-manager*** » s'est bien déroulé.

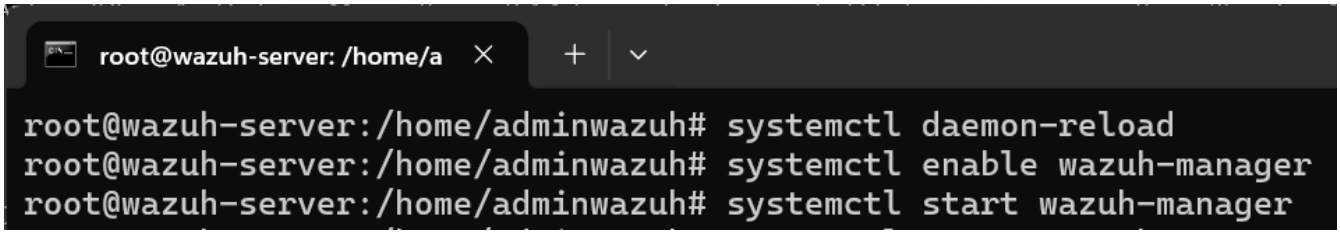


- Une fois l'installation de « ***wazuh-manager*** » terminé, nous devons exécuter une suite de commande pour activer et lancer le service « ***wazuh-manager*** »

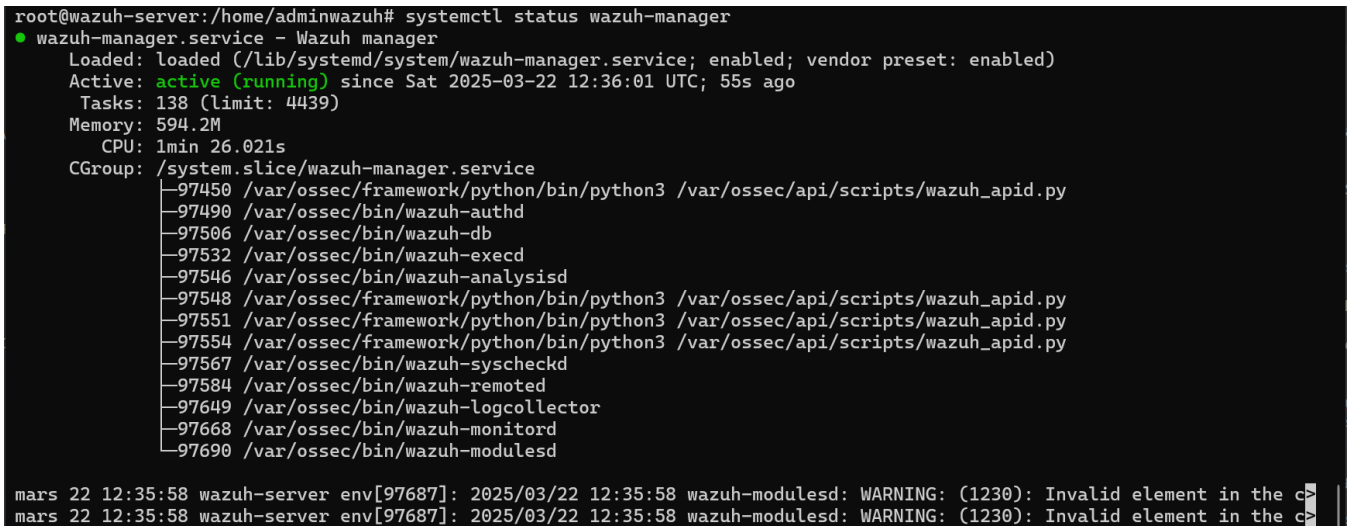
systemctl daemon-reload

systemctl enable wazuh-manager

systemctl start wazuh-manager

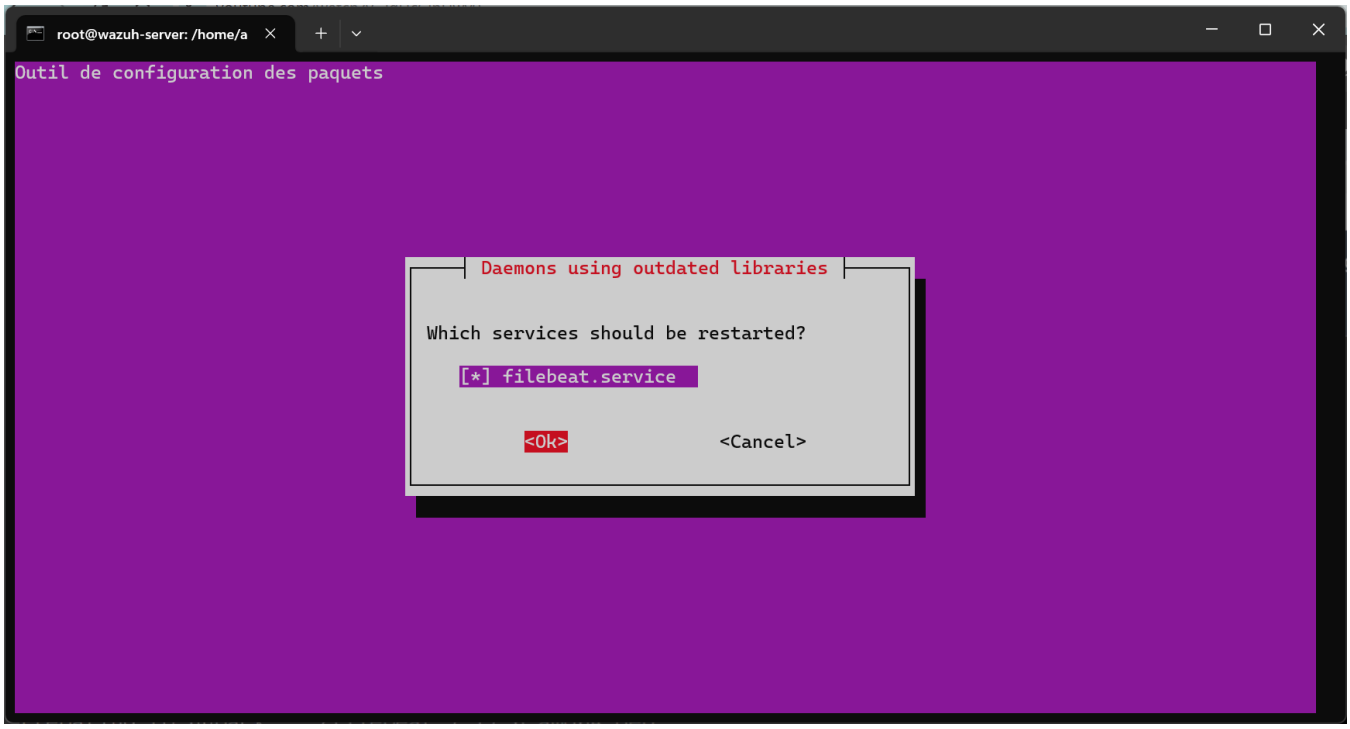


- Nous allons maintenant vérifier que notre service a démarré correctement avec la commande « ***systemctl status wazuh-manager*** ». La sortie de la commande nous affiche bien « ***active (running)*** », ce qui signifie que notre service est actif et a bien démarré

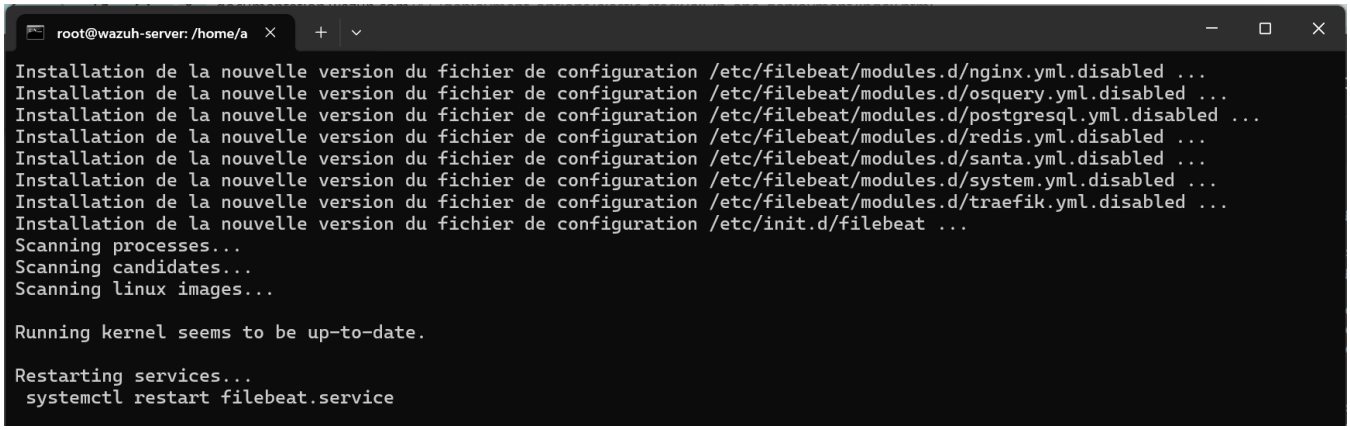


4. Installation de Filebeat

- A ce niveau, nous allons devoir installer le service « ***Filebeat*** » qui remplace ici, le service « ***Logstash*** » de la pile ELK. Pour se faire, nous allons exécuter la commande « ***apt-get install filebeat=7.17.13*** ». Après avoir lancer cette commande, nous aurons une fenêtre dans notre terminal, nous tapons juste « ***Entrer*** »



- Ensuite, nous pouvons voir que l'installation c'est bien terminé et que notre service redémarre correctement.

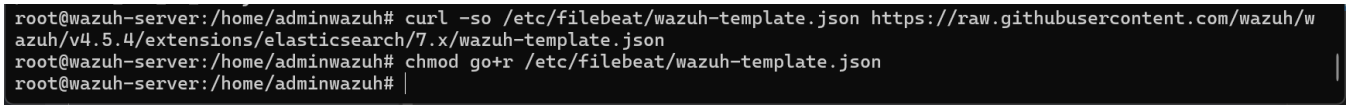


- Nous allons maintenant télécharger le fichier de configuration préconfiguré de ***« Filebeat*** » qui sera utilisé pour transférer les alertes ***Wazuh*** vers ***Elasticsearch***. Pour se faire, on exécute la commande suivante : « ***curl -so /etc/filebeat/filebeat.yml https://packages.wazuh.com/4.5/tpl/elastic-basic/filebeat_all_in_one.yml*** »

- Nous téléchargeons maintenant le template des alertes pour ***Elasticsearch*** et accorder les droits nécessaires au fichier en exécutant les commandes suivantes :

curl -so /etc/filebeat/wazuh-template.json https://raw.githubusercontent.com/wazuh/wazuh/v4.5.4/extensions/elasticsearch/7.x/wazuh-template.json

chmod go+r /etc/filebeat/wazuh-template.json



- À ce niveau, nous téléchargerons le module ***Wazuh*** pour ***Filebeat*** avec cette commande « ***curl -s https://packages.wazuh.com/4.x/filebeat/wazuh-filebeat-0.2.tar.gz | tar -xvz -C /usr/share/filebeat/module*** »

```
root@wazuh-server:/home/adminwazuh# curl -s https://packages.wazuh.com/4.x/filebeat/wazuh-filebeat-0.2.tar.gz | tar -xvz
 -C /usr/share/filebeat/module
wazuh/alerts/
wazuh/alerts/config/
wazuh/alerts/config/alerts.yml
wazuh/alerts/manifest.yml
wazuh/alerts/ingest/
wazuh/alerts/ingest/pipeline.json
wazuh/archives/
wazuh/archives/config/
wazuh/archives/config/archives.yml
wazuh/archives/manifest.yml
wazuh/archives/ingest/
wazuh/archives/ingest/pipeline.json
wazuh/module.yml
root@wazuh-server:/home/adminwazuh#
```

- Nous devons maintenant éditer le fichier de configuration de ***Filebeat*** ''***/etc/filebeat/filebeat.yml***'' pour ajouter/modifier cette ligne « ***output.elasticsearch.password : <elasticsearch_password>*** », il ne faudra pas oublier de remplacer ''<elasticsearch_password>'' par la vraie valeur. Nous exécuterons la commande « ***nano /etc/filebeat/filebeat.yml*** ».

```
root@wazuh-server: /home/a
  GNU nano 6.2                                /etc/filebeat/filebeat.yml *
# Wazuh - Filebeat configuration file
output.elasticsearch.hosts: ["127.0.0.1:9200"]
output.elasticsearch.password: 7YBFD0yI9bOO0V5RoFbj

filebeat.modules:
  - module: wazuh
    alerts:
      enabled: true
    archives:
      enabled: false

setup.template.json.enabled: true
setup.template.json.path: /etc/filebeat/wazuh-template.json
setup.template.json.name: wazuh
setup.template.overwrite: true
setup.ilm.enabled: false

output.elasticsearch.protocol: https
output.elasticsearch.ssl.certificate: /etc/elasticsearch/certs/elasticsearch.crt
output.elasticsearch.ssl.key: /etc/elasticsearch/certs/elasticsearch.key
output.elasticsearch.ssl.certificate_authorities: /etc/elasticsearch/certs/ca/ca.crt
output.elasticsearch.ssl.verification_mode: strict
output.elasticsearch.username: elastic

logging.metrics.enabled: false


^G Help        ^O Write Out   ^W Where Is    ^K Cut         ^T Execute     ^C Location    M-U Undo       M-A Set Mark
^X Exit        ^R Read File   ^\ Replace     ^U Paste       ^J Justify     ^/ Go To Line  M-E Redo       M-6 Copy
```

- Nous copions maintenant tous les certificats de ***Filebeat*** dans « ***/etc/filebeat/certs*** » en exécutant les commandes suivantes :

cp -r /etc/elasticsearch/certs/ca/ /etc/filebeat/certs/

cp /etc/elasticsearch/certs/elasticsearch.crt /etc/filebeat/certs/filebeat.crt

cp /etc/elasticsearch/certs/elasticsearch.key /etc/filebeat/certs/filebeat.key

```
root@wazuh-server:/home/adminwazuh# cp -r /etc/elasticsearch/certs/ca/ /etc/filebeat/certs/
root@wazuh-server:/home/adminwazuh# cp /etc/elasticsearch/certs/elasticsearch.crt /etc/filebeat/certs/filebeat.crt
root@wazuh-server:/home/adminwazuh# cp /etc/elasticsearch/certs/elasticsearch.key /etc/filebeat/certs/filebeat.key
root@wazuh-server:/home/adminwazuh#
```

- Une fois l'installation de « ***filebeat*** » terminé, nous devons exécuter une suite de commande pour activer, lancer et vérifier le status du service « ***filebeat*** »

systemctl daemon-reload

systemctl enable filebeat

systemctl start filebeat

systemctl status filebeat

```
root@wazuh-server:/home/adminwazuh# systemctl daemon-reload
root@wazuh-server:/home/adminwazuh# systemctl enable filebeat
Synchronizing state of filebeat.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable filebeat
root@wazuh-server:/home/adminwazuh# systemctl start filebeat
root@wazuh-server:/home/adminwazuh# systemctl status filebeat
● filebeat.service - Filebeat sends log files to Logstash or directly to Elasticsearch.
     Loaded: loaded (/lib/systemd/system/filebeat.service; enabled; vendor preset: enabled)
     Active: active (running) since Sat 2025-03-22 12:40:03 UTC; 6min ago
       Docs: https://www.elastic.co/beats/filebeat
   Main PID: 99494 (filebeat)
      Tasks: 8 (limit: 4439)
     Memory: 27.6M
        CPU: 848ms
     CGroup: /system.slice/filebeat.service
             └─99494 /usr/share/filebeat/bin/filebeat --environment systemd -c /etc/filebeat/filebeat.yml --path.home />

mars 22 12:42:07 wazuh-server filebeat[99494]: 2025-03-22T12:42:07.342Z        INFO        [monitoring]        log/log.>
mars 22 12:42:37 wazuh-server filebeat[99494]: 2025-03-22T12:42:37.343Z        INFO        [monitoring]        log/log.>
mars 22 12:43:07 wazuh-server filebeat[99494]: 2025-03-22T12:43:07.342Z        INFO        [monitoring]        log/log.>
mars 22 12:43:37 wazuh-server filebeat[99494]: 2025-03-22T12:43:37.343Z        INFO        [monitoring]        log/log.>
mars 22 12:44:07 wazuh-server filebeat[99494]: 2025-03-22T12:44:07.343Z        INFO        [monitoring]        log/log.>
mars 22 12:44:37 wazuh-server filebeat[99494]: 2025-03-22T12:44:37.343Z        INFO        [monitoring]        log/log.>
mars 22 12:45:07 wazuh-server filebeat[99494]: 2025-03-22T12:45:07.351Z        INFO        [monitoring]        log/log.>
mars 22 12:45:37 wazuh-server filebeat[99494]: 2025-03-22T12:45:37.341Z        INFO        [monitoring]        log/log.>
mars 22 12:46:07 wazuh-server filebeat[99494]: 2025-03-22T12:46:07.342Z        INFO        [monitoring]        log/log.>
mars 22 12:46:37 wazuh-server filebeat[99494]: 2025-03-22T12:46:37.347Z        INFO        [monitoring]        log/log.>
lines 1-21/21 (END)
```

- Nous pouvons aussi nous assurer que ***Filebeat*** a bien été installé, ainsi que toutes ses dépendances avec cette commande « ***filebeat test output*** »

```
root@wazuh-server:/home/adminwazuh# filebeat test output
elasticsearch: https://127.0.0.1:9200...
  parse url... OK
  connection...
    parse host... OK
    dns lookup... OK
    addresses: 127.0.0.1
    dial up... OK
  TLS...
    security: server's certificate chain verification is enabled
    handshake... OK
    TLS version: TLSv1.3
    dial up... OK
  talk to server... OK
  version: 7.17.13
root@wazuh-server:/home/adminwazuh#
```

5. Installation de Kibana

- Une fois l'installation de ***Elasticsearch, Wazuh-manager*** et ***Filebeat*** terminée, il ne nous reste plus qu'à installer « ***Kibana*** ». Pour entamer l'installation, nous allons lancer la commande « ***apt-get install kibana=7.17.13*** »

```
root@wazuh-server: /home/a
root@wazuh-server:/home/adminwazuh# apt-get install kibana=7.17.13
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les NOUVEAUX paquets suivants seront installés :
  kibana
0 mis à jour, 1 nouvellement installés, 0 à enlever et 4 non mis à jour.
Il est nécessaire de prendre 307 Mo dans les archives.
Après cette opération, 826 Mo d'espace disque supplémentaires seront utilisés.
Réception de :1 https://artifacts.elastic.co/packages/7.x/apt stable/main amd64 kibana amd64 7.17.13 [307 MB]
307 Mo réceptionnés en 48s (6 425 ko/s)
Sélection du paquet kibana précédemment désélectionné.
(Lecture de la base de données... 218608 fichiers et répertoires déjà installés.)
Préparation du dépaquetage de .../kibana_7.17.13_amd64.deb ...
Dépaquetage de kibana (7.17.13) ...
Paramétrage de kibana (7.17.13) ...
Creating kibana group... OK
Creating kibana user... OK
```

- Vu que ***Kibana*** va traiter directement avec ***Elasticsearch***, il va falloir les synchroniser en copiant les certificats d'***Elasticseach*** dans le dossier de configuration de ***Kibana*** en exécutant les commandes suivantes :

mkdir /etc/kibana/certs/ca -p

cp -R /etc/elasticsearch/certs/ca/ /etc/kibana/certs/

cp /etc/elasticsearch/certs/elasticsearch.key /etc/kibana/certs/kibana.key

cp /etc/elasticsearch/certs/elasticsearch.crt /etc/kibana/certs/kibana.crt

chown -R kibana:kibana /etc/kibana/

chmod -R 500 /etc/kibana/certs

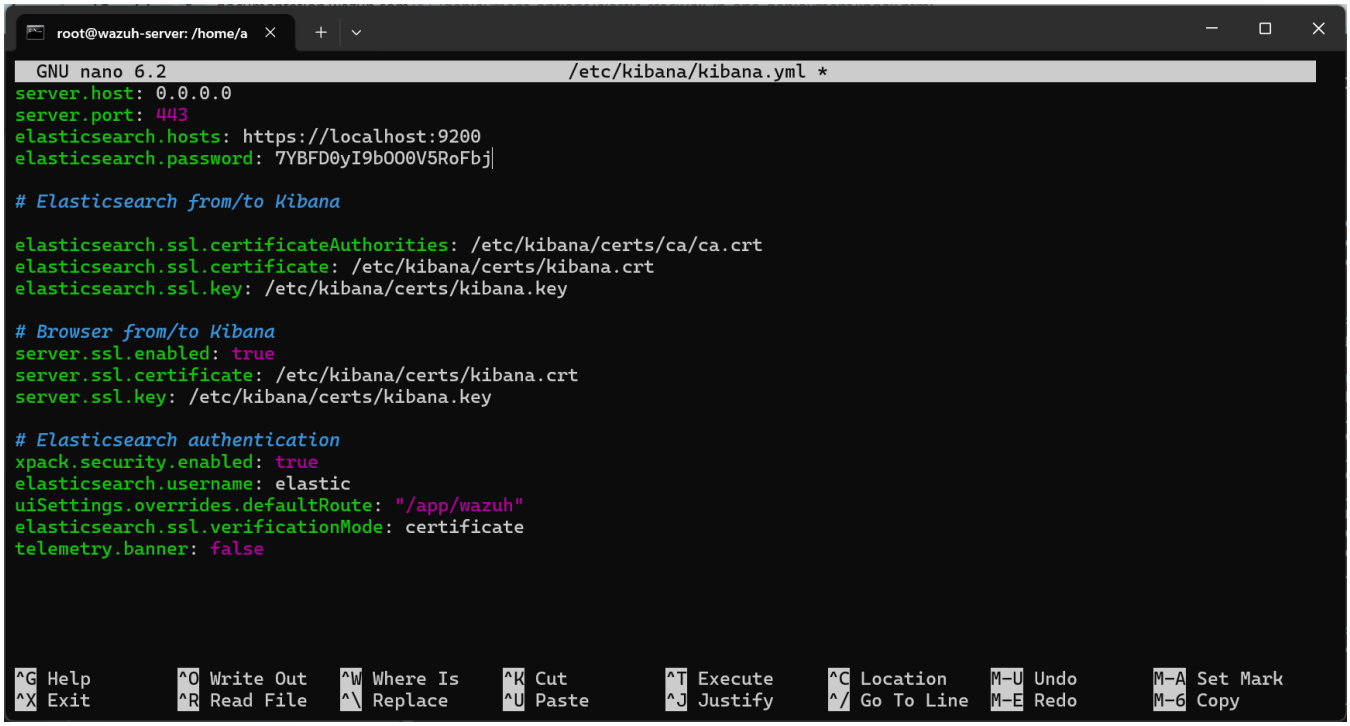
chmod 440 /etc/kibana/certs/ca/ca.* /etc/kibana/certs/kibana.*

```
root@wazuh-server:/home/adminwazuh# mkdir /etc/kibana/certs/ca -p
cp -R /etc/elasticsearch/certs/ca/ /etc/kibana/certs/
cp /etc/elasticsearch/certs/elasticsearch.key /etc/kibana/certs/kibana.key
cp /etc/elasticsearch/certs/elasticsearch.crt /etc/kibana/certs/kibana.crt
chown -R kibana:kibana /etc/kibana/
chmod -R 500 /etc/kibana/certs
chmod 440 /etc/kibana/certs/ca/ca.* /etc/kibana/certs/kibana.*
root@wazuh-server:/home/adminwazuh#
```

- A présent nous allons télécharger le fichier de configuration de ***Kibana*** en tapant cette commande « ***curl -so /etc/kibana/kibana.yml https://packages.wazuh.com/4.5/tpl/elastic-basic/kibana_all_in_one.yml*** »

```
root@wazuh-server:/home/adminwazuh# curl -so /etc/kibana/kibana.yml https://packages.wazuh.com/4.5/tpl/elastic-basic/kib
ana_all_in_one.yml
root@wazuh-server:/home/adminwazuh#
```

- Nous devons maintenant éditer le fichier de configuration de ***Kibana*** ''***/etc/kibana/kibana.yml***'' pour ajouter/modifier cette ligne « ***elasticsearch.password: <elasticsearch_password>*** », il ne faudra pas oublier de remplacer ''<elasticsearch_password>'' par la vraie valeur. Nous exécuterons la commande « ***nano /etc/kibana/kibana.yml*** ».



- Nous allons créer maintenant un répertoire pour les données de ***Kibana*** dans « ***/usr/share/kibana/data*** » et lui attribuerons les privilèges nécessaires en exécutant les commandes suivantes :

mkdir /usr/share/kibana/data

chown -R kibana:kibana /usr/share/kibana

```
root@wazuh-server:/home/adminwazuh# mkdir /usr/share/kibana/data
root@wazuh-server:/home/adminwazuh# chown -R kibana:kibana /usr/share/kibana
root@wazuh-server:/home/adminwazuh#
```

- Nous installerons le plugin wazuh de ***Kibana***. Cette installation doit se faire depuis le répertoire principal de ***Kibana*** en exécutant les commandes suivantes :

cd /usr/share/kibana

sudo -u kibana /usr/share/kibana/bin/kibana-plugin install
https://packages.wazuh.com/4.x/ui/kibana/wazuh_kibana-4.5.4_7.17.13-1.zip

```
root@wazuh-server:/home/adminwazuh# cd /usr/share/kibana
root@wazuh-server:/usr/share/kibana# sudo -u kibana /usr/share/kibana/bin/kibana-plugin install https://packages.wazuh.c
om/4.x/ui/kibana/wazuh_kibana-4.5.4_7.17.13-1.zip
Plugin installation was unsuccessful due to error "ENOENT: no such file or directory, mkdir '/usr/share/kibana/plugins/.
plugin.installing'"
root@wazuh-server:/usr/share/kibana#
```

- Nous allons devoir lier le socket de ***Kibana*** au port privilégié ***443*** avec cette commande « ***setcap 'cap_net_bind_service=+ep' /usr/share/kibana/node/bin/node*** »

```
root@wazuh-server:/usr/share/kibana# setcap 'cap_net_bind_service=+ep' /usr/share/kibana/node/bin/node
root@wazuh-server:/usr/share/kibana#
```

- L'installation de ***Kibana*** est à présent terminée. Nous allons activer, lancer et vérifier le status du service, pour nous assurer que tout s'est bien passé en exécutant les commandes qui suivent :

systemctl daemon-reload

systemctl enable kibana

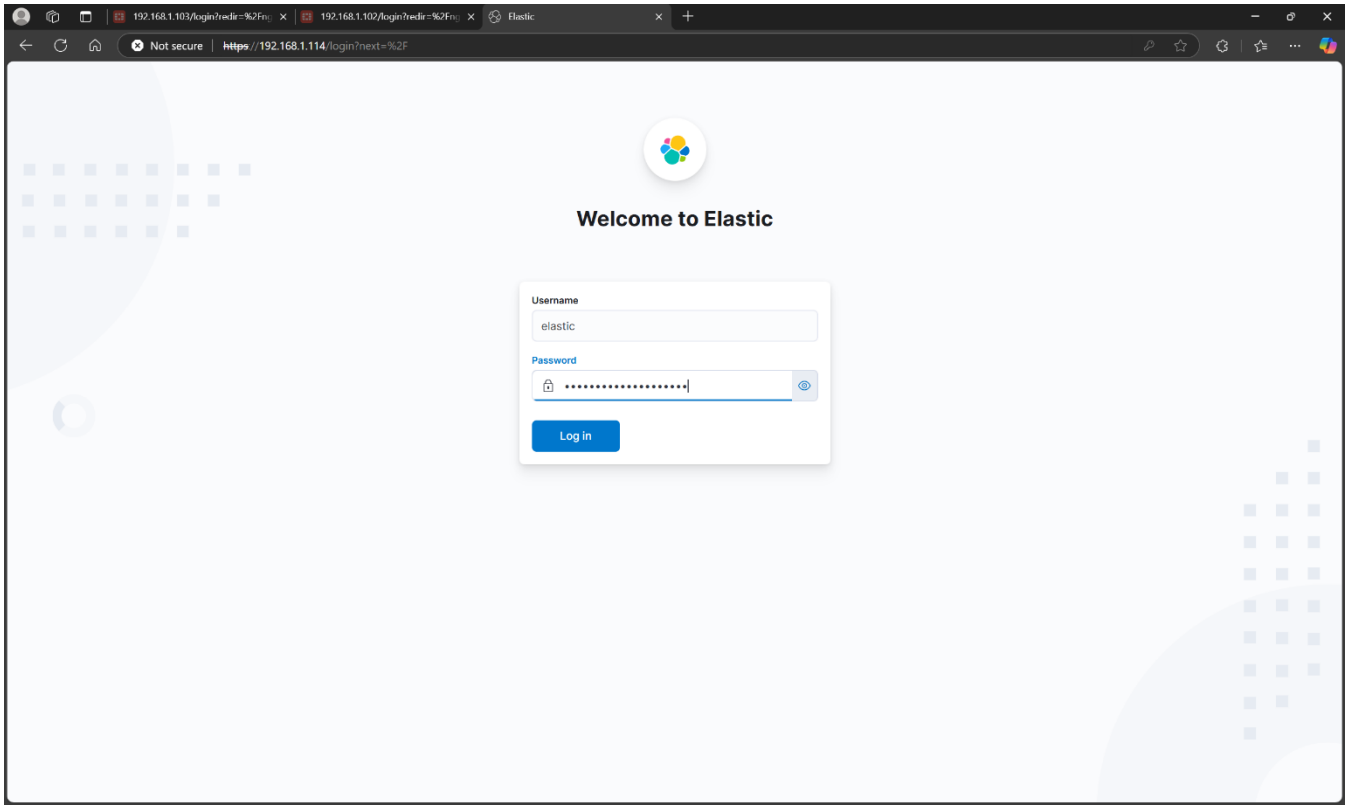
systemctl start kibana

systemctl status kibana

```
root@wazuh-server:/usr/share/kibana# systemctl daemon-reload
root@wazuh-server:/usr/share/kibana# systemctl enable kibana
Synchronizing state of kibana.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable kibana
root@wazuh-server:/usr/share/kibana# systemctl start kibana
root@wazuh-server:/usr/share/kibana# systemctl status kibana
● kibana.service - Kibana
     Loaded: loaded (/etc/systemd/system/kibana.service; enabled; vendor preset: enabled)
     Active: active (running) since Sat 2025-03-22 12:58:32 UTC; 2s ago
       Docs: https://www.elastic.co
   Main PID: 100270 (node)
      Tasks: 7 (limit: 4439)
     Memory: 52.2M
        CPU: 2.868s
     CGroup: /system.slice/kibana.service
             └─100270 /usr/share/kibana/bin/../node/bin/node /usr/share/kibana/bin/../src/cli/dist --logging.dest=/var/>

mars 22 12:58:32 wazuh-server systemd[1]: Started Kibana.
mars 22 12:58:32 wazuh-server kibana[100270]: Kibana is currently running with legacy OpenSSL providers enabled! For de>
lines 1-13/13 (END)
```

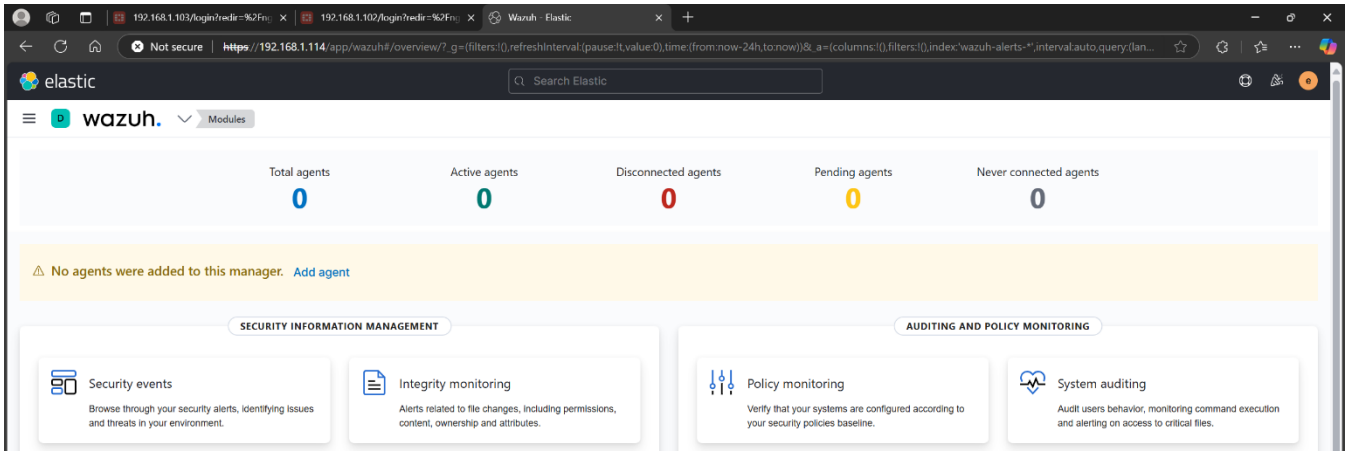
- Notre système ***Wazuh*** a bien été installé et intégrer avec la ***Pile ELK.*** Nous allons à présent pouvoir nous connecter à l'interface web d'administration en utilisant l'adresse IP de notre serveur ***Wazuh.*** Le login est « ***elastic*** » et le password est le credential généré lors de l'installation d'***Elasticsearch***.



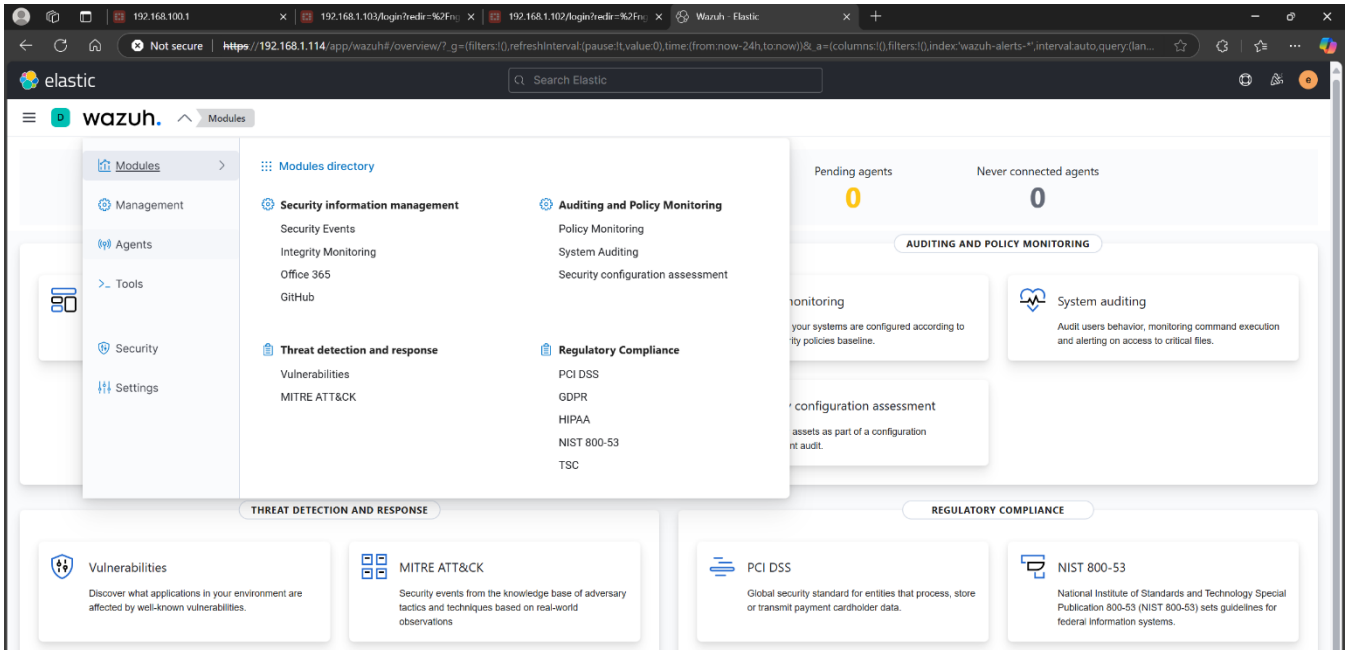
III. TESTS DE SIMULATION

1. Déploiement des Agents Wazuh

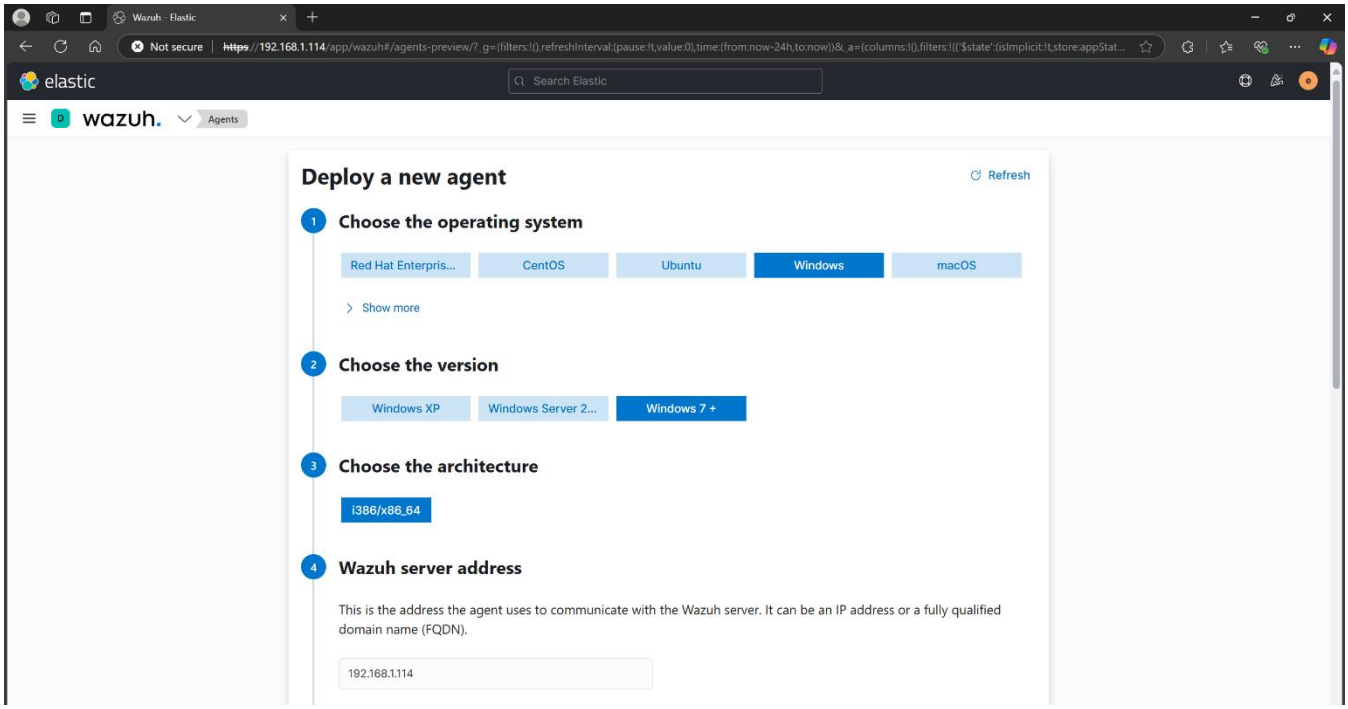
➢ Après l'installation et la configuration de ***Wazuh*** couplé à la ***Pile ELK***, nous allons installer les « ***wazuh-agent*** » et observer les altertes de wazuh.



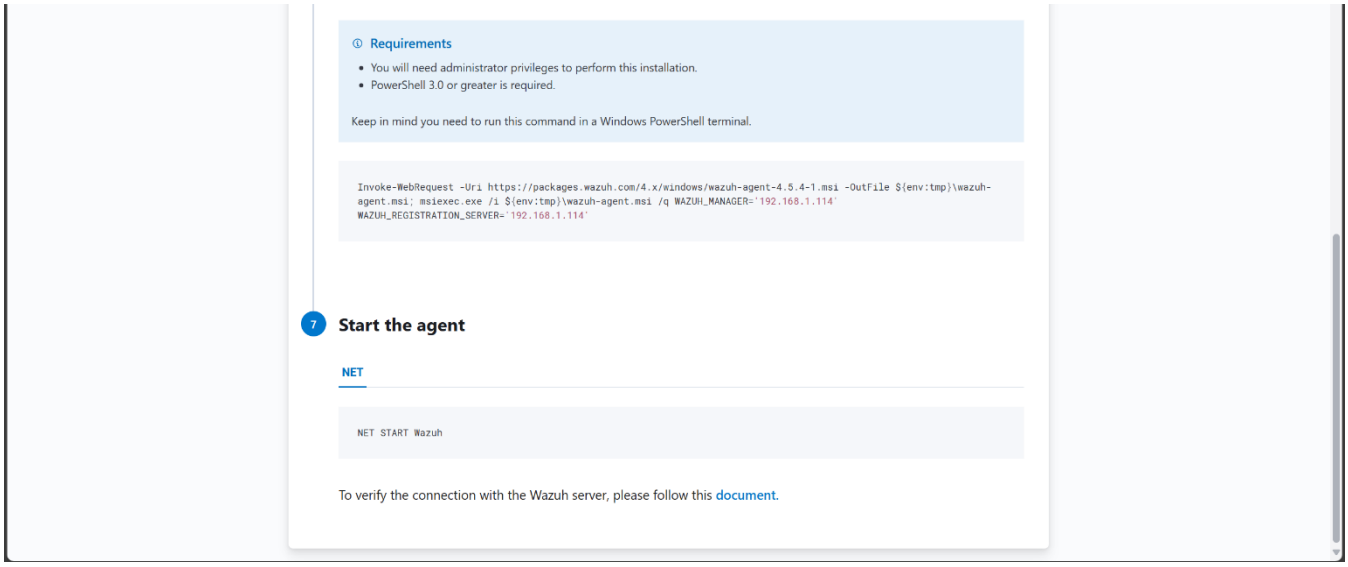
➢ Nous allons maintenant, dans ***« Agents »*** après avoir fait défiler le menu tout en haut à gauche.



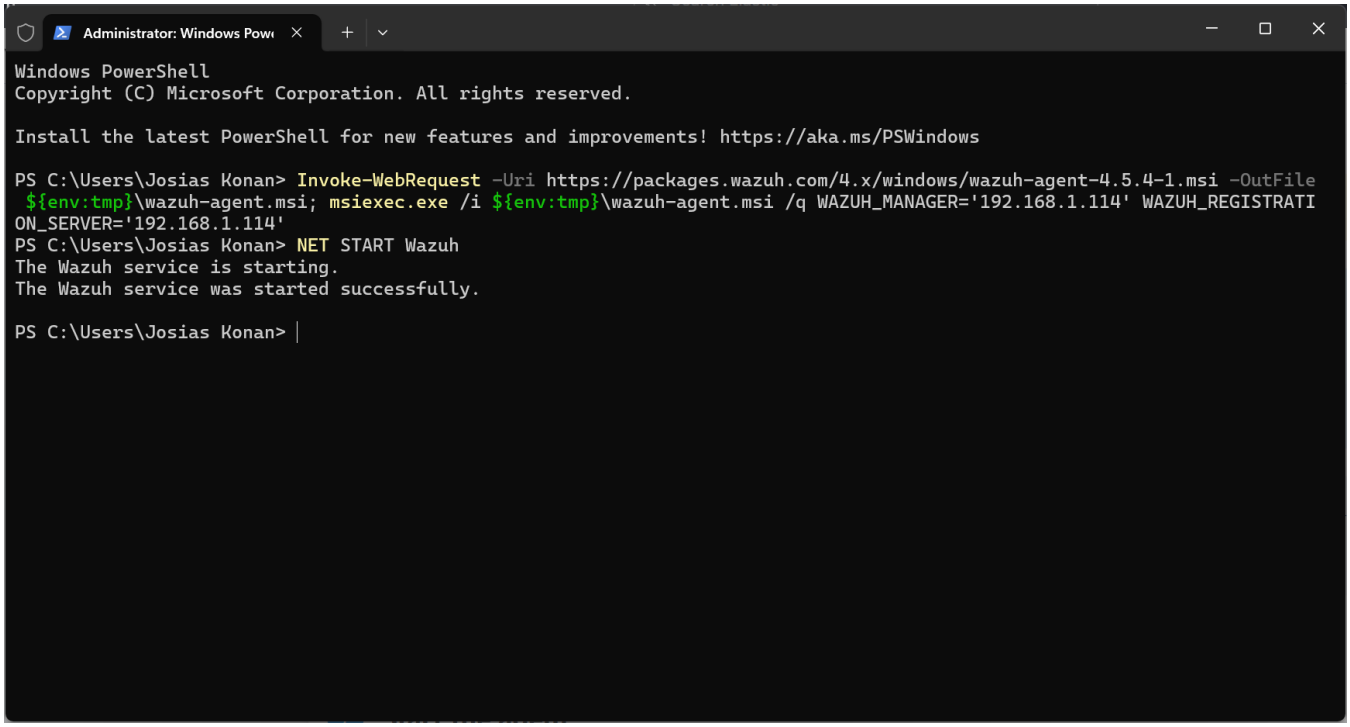
- Nous allons venir déployer un nouvel agent sur notre Windows 11. Nous sélectionnons donc « ***Windows*** », « ***Windows 7+*** » et nous renseignons l'adresse IP du serveur Wazuh, « ***192.168.1.114*** » dans notre cas



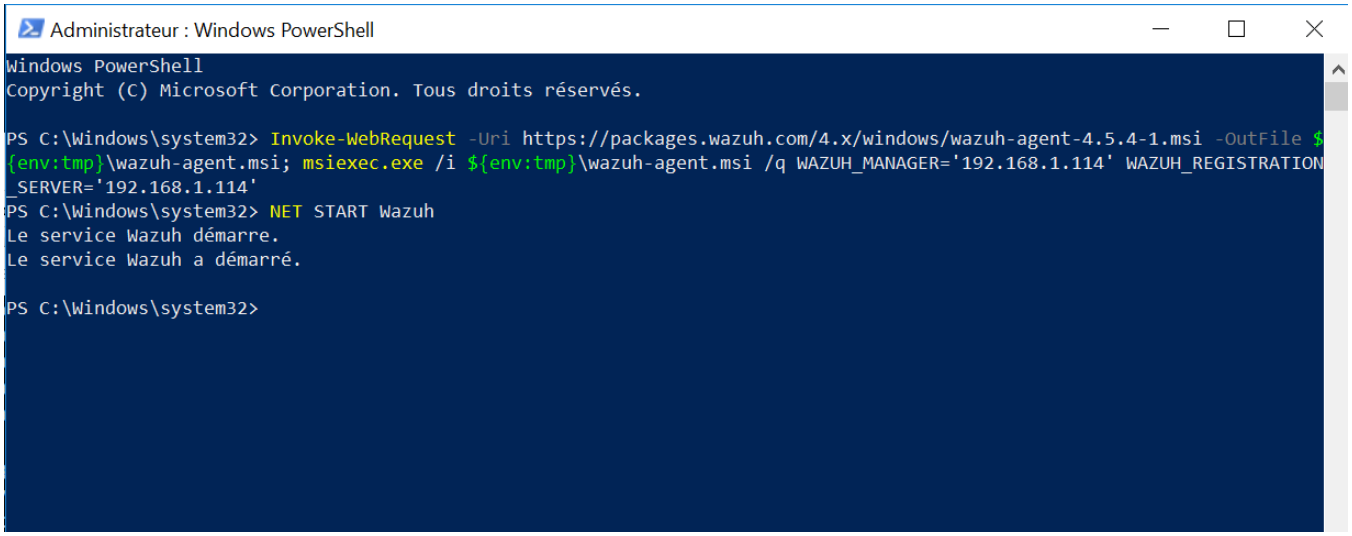
- A présent, nous allons copier la commande pour installer l'agent en ligne de commande PowerShell sur Windows, ensuite faudra le démarrer avec la commande suivante.



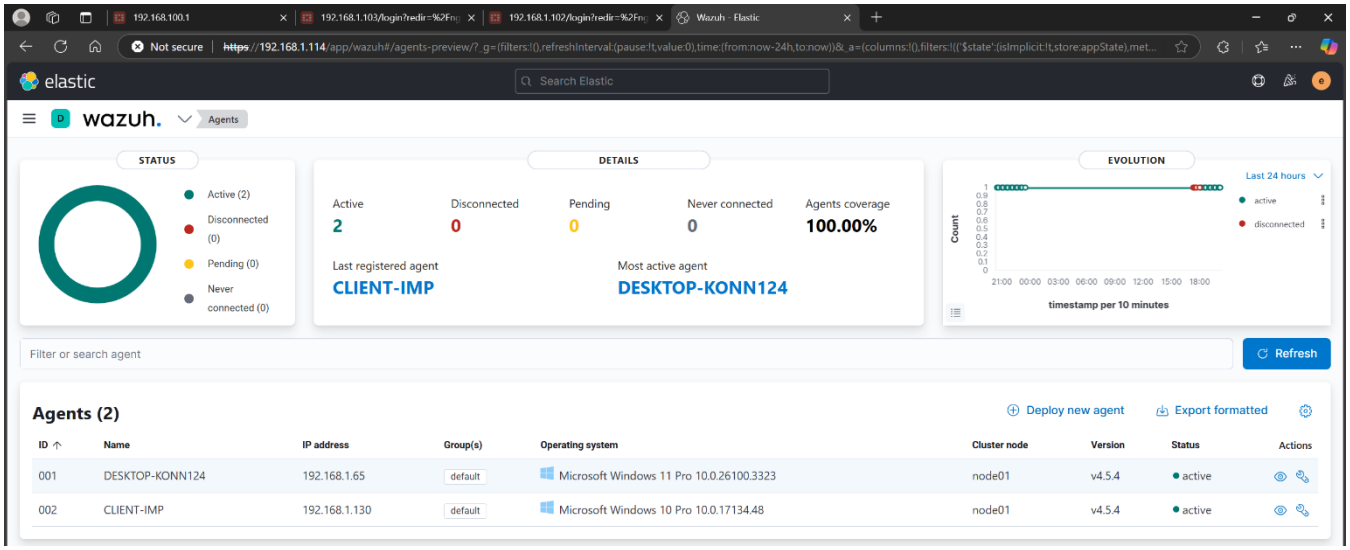
- Nous exécutons ces deux commandes sur notre machine cliente. Et nous pouvons constater que notre « ***wazuh-agent*** » a été installé et a bien démarré



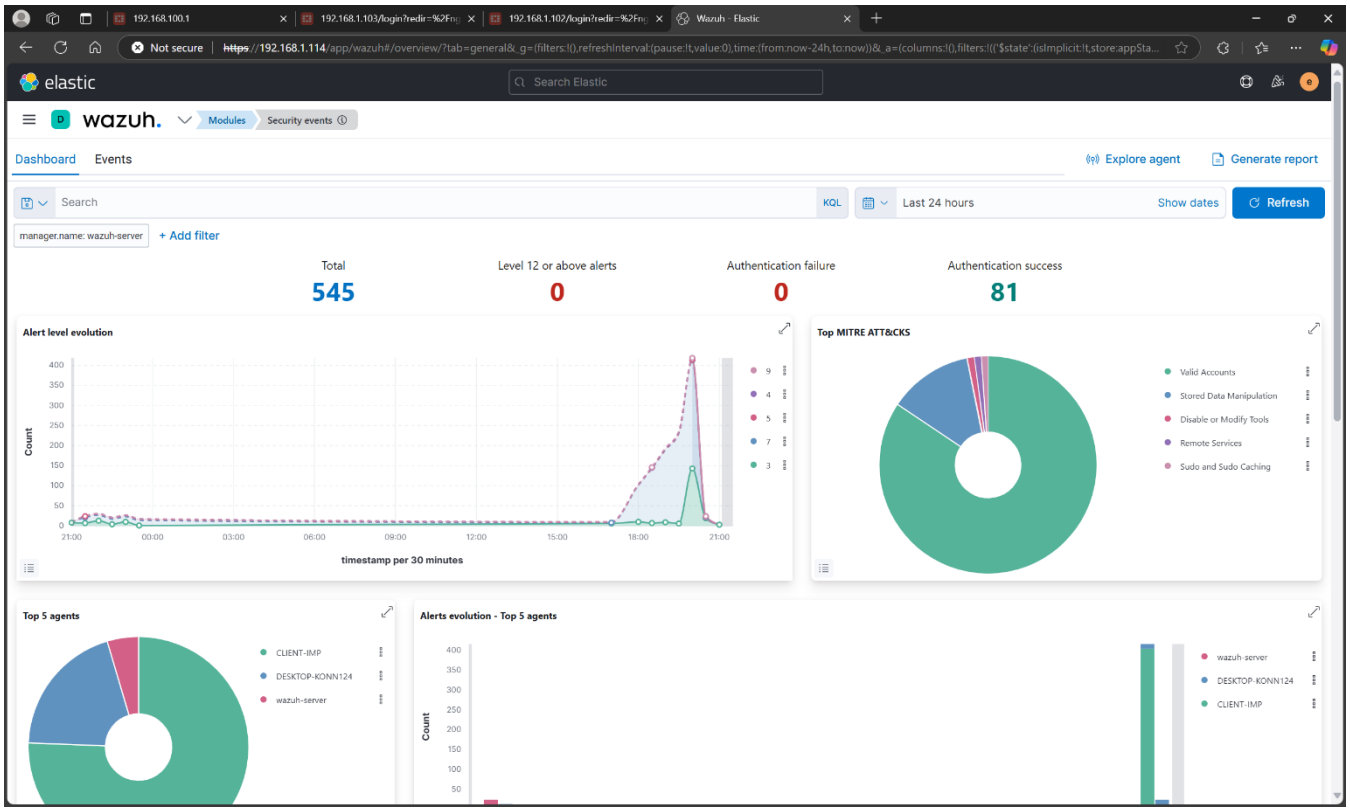
- Nous ferons la même chose sur notre deuxième machine cliente sous Windows 10 avec les mêmes commandes



- Une fois nos agents installés, nous pourrons voir sur notre interface d'administration que le « ***Wazuh Server*** » les a bien détecté



- Nous pouvons maintenant observer les évènements depuis notre interface d'administration, dans « ***Modules*** » > « ***Security Events*** »



CONCLUSION

L'**implémentation d'un SIEM avec Wazuh et la stack ELK** nous a permis de centraliser, analyser et corréler les alertes de sécurité de manière efficace. Grâce à cette solution, nous avons amélioré la **visibilité sur les menaces**, renforcé la **détection d'intrusions** et optimisé la **gestion des incidents**.

Toutefois, pour aller plus loin dans la **protection proactive**, l'intégration d'un **IDS/IPS comme Suricata** devient essentielle. Cette prochaine étape permettra d'**analyser le trafic réseau en temps réel**, détecter les menaces avant qu'elles ne causent des dommages, et les bloquer si nécessaire. Ainsi, l'ajout de Suricata à notre SIEM Wazuh complétera notre approche en **renforçant la prévention et la réponse aux cyberattaques**.