

WAZUH





Sommaire

١.	Ir	ntroduction3
1)	Présentation3
2)	Prérequis3
II.	Ir	nstallation4
1)	Service ssh4
2)	Création des certificats5
3)	Installation du nœud6
4	.)	Initialisation du cluster8
5)	Installation du server9
6)	Configuration du server node 'Master' (Seulement pour une configuration mult-node)13
7)	Installation du Dashboard15
8)	Agent Wazuh17
	1	.1. Linux
	1	.2. Windows19
111.		Configuration19
1)	Active-response
2)	Quoi de plus23
	2	.1. Vulnerabilty-dectetion
	2	.2 intégrations de slack



I. Introduction

1) Présentation

Wazuh est une plateforme open-source host-based intrusion detection system (HIDS) dédiée à la détection, à la prévention et à la réponse aux menaces dans divers environnements informatiques

2) Prérequis

- 📌 1. Configuration Matérielle Recommandée
- Pour une installation "All-in-One" (Wazuh Manager + OpenSearch + Dashboard)

Ressource Minimum Recommandé

CPU	4 vCPU	8+ vCPU					
RAM	8 Go	16+ Go					
Stockage	80 Go	200+ Go (selon les logs)					
Réseau	1 Gbps	1 Gbps+					
 Pour une installation en cluster (grandes infrastructures) 							

- Wazuh Manager, OpenSearch et Dashboard doivent être installés sur des machines séparées.
- Chaque composant a besoin d'un minimum de 4 vCPU, 8 Go RAM, et 100+ Go de stockage.
- 2. Systèmes d'Exploitation Supportés
- Serveur Wazuh

Wazuh est compatible avec :

- 🗹 Ubuntu 20.04+ / Debian 11+ (Recommandé)
- CentOS 7 / AlmaLinux 8+ / Rocky Linux 8+
- 🗹 RHEL 7+
- 🗹 Amazon Linux 2
- Agents Wazuh

Les agents peuvent être installés sur :

- 🗹 Windows (7, 10, 11, Server 2012+), Linux (Debian, Ubuntu, CentOS, etc.)
- MacOS, AIX, Solaris, FreeBSD

📌 3. Prérequis Logiciels



• Paquets à installer avant Wazuh (sur Ubuntu/Debian)

« sudo apt update && sudo apt install -y curl unzip software-properties-common »

📌 4. Autres Prérequis

Un accès Internet pour télécharger les paquets

Une IP fixe (éviter DHCP sur le serveur Wazuh)

- II. Installation
- 1) Service ssh

Si vous avez installé une version graphique (aussi valable pour la version en ligne de commande) je conseille d'installer le service ssh pour prendre à distance votre serveur.

Pour commencer, il faut faire la commande « (sudo si pas en administrateur) apt install ssh-server ».

root@wazuh:/home/wazuh# apt install openssh-server

Une fois installé il faut modifier le fichier de configuration avec « nano /etc/ssh/sshd_config ».

root@wazuh:/home/wazuh# nano /etc/ssh/sshd_config

Dans le fichier, écrire deux lignes :

- Port 22
- PermitRootLogin yes



Et pour finir, faire la commande : « systemctl restart ssh.service ».

root@wazuh:/etc/ssh# systemctl restart ssh.service

Pour accéder à la machine en ssh on a plusieurs options :

- Utilisez un logiciel tel que Putty.



- Utiliser le terminal avec la commande « ssh user@machine ».

2) Création des certificats

On commence par installer l'assistant d'installation Wazuh et le fichier de configuration. (Cela crée les certificats qui chiffrent les communications entre les composants centraux de Wazuh.) avec les commandes :

« curl -sO https://packages.wazuh.com/4.11/wazuh-cert-tool.sh ».

```
root@wazuh:/home/wazuh# curl -s0 https://packages.wazuh.com/4.11/wazuh-cert-tool.sh
```

« curl -sO https://packages.wazuh.com/4.11/config.yml ».

```
root@wazuh:/home/wazuh# curl -s0 https://packages.wazuh.com/4.11/config.yml
```

Puis on vient modifier le fichier config.yml avec la commande « nano config.yml ».

```
root@wazuh:/home/wazuh# nano config.yml
```

Une fois dans le fichier il faut mettre l'adresse IP du ou des serveur(s) et le nom du nœud où c'est écrit «- name : », « ip : ». Dans ce cas on peut voir :

- Name: node-1
- ip: « 192.168.60.20 »

```
GNU nano 7.2
nodes:
# Wazuh indexer nodes
indexer:
    name: node-1
    ip: "192.168.60.20"
    #- name: node-2
    # ip: "<indexer-node-ip>"
    #- name: node-3
    # ip: "<indexer-node-ip>"
# Wazuh server nodes
# If there is more than one Wazuh server
# node, each one must have a node_type
server:
    - name: wazuh-1
    ip: "192.168.60.20"
    # node_type: master
    #- name: wazuh-2
    # ip: "<wazuh-manager-ip>"
    # node_type: worker
    #- name: wazuh-3
    # ip: "<wazuh-manager-ip>"
    # node_type: worker
# node_type: worker
# node_type: worker
# wazuh dashboard nodes
dashboard:
    - name: dashboard
    ip: "192.168.60.20"
```



Maintenant pour créer les certificats, il faut exécuter le fichier ./wazuh-certs-tool.sh . (Pour les clusters multi-nœuds, ces certificats doivent être déployés ultérieurement sur toutes les instances Wazuh de votre cluster). Pour ce faire il faut faire la commande « bash ./wazuh-certs-tool.sh -A ».

root@wazuh:/home/wazuh# bash ./wazuh-certs-tool.sh -A

Compressez les fichiers nécessaires avec la commande « tar -cvf ./wazuh-certificates.tar -C ./wazuh-certificates/ . ».

root@wazuh:/home/wazuh# tar -cvf ./wazuh-certificates.tar -C ./wazuh-certificates/ .

Puis « rm -rf ./wazuh-certificates ».

root@wazuh:/home/wazuh# rm -rf ./wazuh-certificates

À savoir : Si vous avez plusieurs serveurs il faut copier le fichier « wazuh-certificates.tar » sur tous les wazuh indexer, wazuh dashboard et wazuh server.

3) Installation du nœud

Pour commencer l'installation du nœud on commence par installer des dépendances des paquets avec la commande suivante « apt-get install debconf adduser procps ».

root@wazuh:/home/wazuh# apt-get install debconf adduser procps

Maintenant, on va ajourner le référentiel Wazuh, en commençant par installer des paquets avec la commande « apt-get install gnupg apt-transport-https ».

root@wazuh:/home/wazuh# apt-get install gnupg apt-transport-https

Installer la clé GPG avec « curl -s https://packages.wazuh.com/key/GPG-KEY-WAZUH | gpg --nodefault-keyring --keyring gnupg-ring:/usr/share/keyrings/wazuh.gpg --import && chmod 644 /usr/share/keyrings/wazuh.gpg ».

root@wazuh:/home/wazuh# curl -s https://packages.wazuh.com/key/GPG-KEY-WAZUH | gpg --no-default-keyring --keyring gnupg-ring:/usr/share/keyrings/wazuh.gp import && chmod 644 /usr/share/keyrings/wazuh.gpg

Ajoutez le référentiel avec « 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:~# 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.lis

Mettre à jour les informations des packages. « apt-get update ».

root@wazuh:/home/wazuh# apt-get update

Maintenant, on installe le package Wazuh indexer. « apt-get -y install wazuh-indexer ».

root@wazuh:/home/wazuh# apt-get -y install wazuh-indexer

Une fois le package installé il faut aller modifier le fichier de configuration avec la commande « nano /etc/wazuh-indexer/opensearch.yml ».

root@wazuh:/home/wazuh# nano /etc/wazuh-indexer/opensearch.yml



Dans le fichier de configuration, il faut remplacer les valeurs suivantes :

- Network.host : « adresse IP ou nom d'hôte du serveur
- Node.name : « nom du nœud de ce serveur ».
- Cluster.initial_master_nodes : « Liste des noms des nœuds éligibles au master ici seulement –« node-1 » car il n'y a pas d'autre machine ».
- Discovery.seed_host : « Liste des noms des nœuds éligibles au master (si seulement un serveur le laisse en commentaire).



Maintenant il faut déployer les certificats. Pour ce faire, commencer d'abord par définir une variable avec la commande, « NODE_NAME=nom-du-nœud-définit-précédemment »

root@wazuh:/home/wazuh# NODE NAME=node-1

Maintenant il faut exécuter les commandes suivantes une à une (possible de toutes les lancer et même temps mais moins fiable) pour déployer les certificats avec le nom du nœud.

« mkdir /etc/wazuh-indexer/certs ».

root@wazuh:~# mkdir /etc/wazuh-indexer/certs

« tar -xf ./wazuh-certificates.tar -C /etc/wazuh-indexer/certs/ ./\$NODE_NAME.pem ./\$NODE_NAME-key.pem ./admin.pem ./admin-key.pem ./root-ca.pem ».

root@wazuh:~# tar -xf ./wazuh-certificates.tar -C /etc/wazuh-indexer/certs/ ./\$NODE_NAME.pem ./\$NODE_NAME-key.pem ./admin.pem ./admin-key.pem ./root-ca.pem

« mv -n /etc/wazuh-indexer/certs/\$NODE_NAME.pem /etc/wazuh-indexer/certs/indexer.pem ».

root@wazuh:~# mv -n /etc/wazuh-indexer/certs/\$NODE_NAME.pem /etc/wazuh-indexer/certs/indexer.pem

« mv -n /etc/wazuh-indexer/certs/\$NODE_NAME-key.pem /etc/wazuh-indexer/certs/indexer-key.pem ».

root@wazuh:~# mv -n /etc/wazuh-indexer/certs/\$NODE_NAME-key.pem /etc/wazuh-indexer/certs/indexer-key.pem

« chmod 500 /etc/wazuh-indexer/certs ».

root@wazuh:~# chmod 500 /etc/wazuh-indexer/certs

« chmod 400 /etc/wazuh-indexer/certs/* ».

root@wazuh:~# chmod 400 /etc/wazuh-indexer/certs*

« chown -R wazuh-indexer:wazuh-indexer /etc/wazuh-indexer/certs ».



root@wazuh:~# chown -R wazuh-indexer:wazuh-indexer /etc/wazuh-indexer/certs

À savoir :

 Si aucun autre composant Wazuh ne doit être installé sur ce nœud, supprimez le fichier wazuh-certificates.tar en exécutant pour augmenter la sécurité. « rm -f ./wazuhcertificates.tar »

Recharger les daemons avec « systemctl daemon-reload ».

root@wazuh:~# systemctl daemon-reload

Activez au démarrage le service avec « systemctl enable wazuh-indexer ».

root@wazuh:~# systemctl enable wazuh-indexer

Démarrer le service avec « systemctl start wazuh-indexer ».

root@wazuh:~# systemctl start wazuh-indexer

À savoir :

- Il faut répéter ces étapes pour chaque nœud wazuh indexer du cluster. Ensuite, initialisez votre cluster mono-nœud ou multi-nœuds à l'étape suivante.

4) Initialisation du cluster

Pour initialiser le cluster, il faut exécuter le script wazuh-indexer sur n'importe quel nœud wazuhindexer pour charger les nouvelles informations de certificats et démarrer le cluster à nœud unique ou à nœuds multiples, avec la commande « /usr/share/wazuh-indexer/bin/indexer-security-init.sh ».

Après avoir installé le cluster, il faut le tester si l'installation a réussi avec la commande « curl -k -u admin:admin https://<adresse-ip-de-l'indexer>:9200 ».

root@wazuh:~# curl -k -u admin:admin https://192.168.60.20:9200

Le résultat doit être sensible à ceci :

```
{
```

```
"name" : "node-1",
```

"cluster_name" : "wazuh-cluster",

```
"cluster_uuid" : "095jEW-oRJSFKLz5wmo5PA",
```



"version" : {

"number" : "7.10.2",

"build_type" : "rpm",

"build_hash" : "db90a415ff2fd428b4f7b3f800a51dc229287cb4",

"build_date" : "2023-06-03T06:24:25.112415503Z",

"build_snapshot" : false,

"lucene_version" : "9.6.0",

"minimum_wire_compatibility_version" : "7.10.0",

"minimum_index_compatibility_version" : "7.0.0"

},

"tagline" : "The OpenSearch Project: https://opensearch.org/"

}

Puis-je on vérifier si le cluster à nœud unique ou à nœuds multiples fonctionne correctement. « curl - k -u admin:admin https://< adresse-ip-de-l'indexer >:9200/_cat/nodes?V ».

root@wazuh:~# curl -k -u admin:admin https://192.168.60.20:9200/_cat/nodes?v

5) Installation du server

Comparer à la documentation trouvable sur le site de Wazuh, on ne commence pas en ajoutant le répertoire Wazuh car on installe tous les services sur la même machine.

Si c'était le cas, voici les différentes commandes à effectuer :

- apt-get install gnupg apt-transport-https
- 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
- 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
- apt-get update

Il faut commencer par installer le package Wazuh Manager avec la commande « apt-get -y install wazuh-manager ».

root@wazuh:~# apt-get -y install wazuh-manager



Installer le package filebeat « apt-get -y install filebeat ».

root@wazuh:~# apt-get -y install filebeat

Télécharger le fichier préconfiguré de filebeat « curl -so /etc/filebeat/filebeat.yml https://packages.wazuh.com/4.11/tpl/wazuh/filebeat/filebeat.yml ».

root@wazuh:~# curl -so /etc/filebeat/filebeat.yml https://packages.wazuh.com/4.11/tpl/wazuh/filebeat/filebeat.yml

Éditer le fichier de configuration « nano /etc/filebeat/filebeat.yml ».

```
root@wazuh:~# nano /etc/filebeat/filebeat.yml
```

Dans le fichier il faut modifier les valeurs suivantes :

- La partie « hosts : ["127.0.0.1 :9200] » il faut changer l'adresse IP 127... par celle de la machine où est le nœud wazuh indexer (si vous avez plusieurs nœuds il faut ajouter les adresses IP dans ce sens ["192.168.60.1:9200", "192.168.60.2:9200", "192.168.60.3:9200"].



Crée un 'Filebeat keystore (magasin de clé)' pour stocker en toute sécurité les informations d'authentification « filebeat keystore create ».

root@wazuh:~# filebeat keystore create

Ajouter un nom d'utilisateur dans le 'keystore' « echo admin | filebeat keystore add username --stdin –force ».

root@wazuh:~# echo admin | filebeat keystore add username --stdin --force

Ajouter un nom d'un mot de passe dans le 'keystore' « echo admin | filebeat keystore add password --stdin –force ».

root@wazuh:~# echo admin | filebeat keystore add password --stdin --force

Téléchargez le modèle d'alertes pour le Wazuh indexer :



« curl -so /etc/filebeat/wazuh-template.json

https://raw.githubusercontent.com/wazuh/wazuh/v4.11.1/extensions/elasticsearch/7.x/wazuhtemplate.json ».

oot@wazuh:~# curl -so /etc/filebeat/wazuh-template.json https://raw.githubusercontent.com/wazuh/wazuh/v4.11.1/extensions/elasticsearch/7.x/wazuh-template.j

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

root@wazuh:~# chmod go+r /etc/filebeat/wazuh-template.json

Installez le module Wazuh pour Filebeat. « curl -s https://packages.wazuh.com/4.x/filebeat/wazuh-filebeat-0.4.tar.gz | tar -xvz -C /usr/share/filebeat/module ».

root@wazuh:~# curl -s https://packages.wazuh.com/4.x/filebeat/wazuh-filebeat-0.4.tar.gz | tar -xvz -C /usr/share/filebeat/module

Encore une fois, on va définir la variable NODE_NAME qui doit être le nom de notre nœud pour déployer les certificats « NODE_NAME=<SERVER_NODE_NAME> ».

root@wazuh:~# NODE NAME=node-1

Et comme déjà fait précédemment on fait les commandes suivantes une à une (possible de toutes les lancer en même temps mais moins fiable) pour déployer les certificats avec le nom du nœud.

« mkdir /etc/filebeat/certs ».

root@wazuh:~# mkdir /etc/filebeat/cert

« tar -xf ./wazuh-certificates.tar -C /etc/filebeat/certs/ ./\$NODE_NAME.pem ./\$NODE_NAMEkey.pem ./root-ca.pem ».

root@wazuh:~# tar -xf ./wazuh-certificates.tar -C /etc/filebeat/certs/ ./\$NOD E NAME.pem ./\$NODE NAME-key.pem ./root-ca.pem

« mv -n /etc/filebeat/certs/\$NODE_NAME.pem /etc/filebeat/certs/filebeat.pem ».

root@wazuh:~# mv -n /etc/filebeat/certs/\$NODE_NAME.pem /etc/filebeat/certs/filebeat.pem

« mv -n /etc/filebeat/certs/\$NODE_NAME-key.pem /etc/filebeat/certs/filebeat-key.pem ».

root@wazuh:~# mv -n /etc/filebeat/certs/\$NODE_NAME-key.pem /etc/filebeat/cert
s/filebeat-key.pem

« chmod 500 /etc/filebeat/certs ».

root@wazuh:~# chmod 500 /etc/filebeat/certs

« chmod 400 /etc/filebeat/certs/* ».

root@wazuh:~# chmod 400 /etc/filebeat/certs/*

« chown -R root:root /etc/filebeat/certs ».

root@wazuh:~# chown -R root:root /etc/filebeat/certs

Une fois les commandes faites il faut enregistrer le nom d'utilisateur et le mot de passe du wazuh indexer dans le magasin de clés avec (par défaut les informations d'identification sont admin : admin) :



« echo '<INDEXER_USERNAME>' | /var/ossec/bin/wazuh-keystore -f indexer -k username ».

« echo '<INDEXER_PASSWORD>' | /var/ossec/bin/wazuh-keystore -f indexer -k password ».

root@wazuh:~# echo '<INDEXER_USERNAME>' | /var/ossec/bin/wazuh-keystore -f in
dexer -k username
root@wazuh:~# echo '<INDEXER_PASSWORD>' | /var/ossec/bin/wazuh-keystore -f in
dexer -k password

Maintenant il faut aller modifier le fichier ossec.conf pour configurer la connexion de l'indexeur « nano /var/ossec/etc/ossec.conf » .

root@wazuh:~# nano /var/ossec/etc/ossec.conf

Dans le fichier il faut modifier la valeur <host> qui par défaut est 0.0.0.0 en la remplaçant par l'adresse IP du nœud wazuh indexer.

```
<indexer>
  <enabled>yes</enabled>
  <hosts>
    <hosts>
    <host>https://192.168.60.20:9200</host>
    </hosts>
    <ssl>
      <certificate_authorities>
        <ca>/etc/filebeat/certs/root-ca.pem</ca>
      </certificate_authorities>
        <certificate_authorities>
        <certificate_authorities>
        <certificate/etc/filebeat/certs/filebeat.pem</certificate>
        <key>/etc/filebeat/certs/filebeat-key.pem</key>
      </ssl>
</indexer>
```

À savoir :

- Si vous avez un cluster, il faut ajouter une ligne <host>adresseIP :9200</host> pour chaque nœud, exemple :
 - o <hosts>
 - o <host>https://10.0.0.1:9200</host>
 - o <host>https://10.0.0.2:9200</host>
 - o </hosts>

Une fois la configuration du serveur faite, il faut démarrer les services (wazuh-manager, Filebeat).

Recharger les daemons « systemctl daemon-reload ».

root@wazuh:~# systemctl daemon-reload

Activer le service au démarrage « systemctl enable wazuh-manager ».

root@wazuh:~# systemctl enable wazuh-manager

Et démarrer le service « systemctl start wazuh-manager ».

root@wazuh:~# systemctl start wazuh-manager

On fait la même chose pour Filebeat

Recharger les daemons « systemctl daemon-reload »

root@wazuh:~# systemctl daemon-reload



Activer le service au démarrage « systemctl enable filebeat »

root@wazuh:~# systemctl enable filebeat

Et démarrer le service « systemctl start filebeat »

root@wazuh:~# systemctl start filebeat

En faisant la commande suivante on teste si le service filebeat est bien installé « filebeat test output »

root@wazuh:~# filebeat test output

Le résultat doit être sensible à ceci :

elasticsearch: https://192.168.60.20:9200...

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: 7.10.2

6) Configuration du server node 'Master' (Seulement pour une configuration multnode)

Après avoir fait l'installation du Wazuh server sur tous les nœuds il faut configurer un serveur nœud en tant que maître et le reste sont des travailleurs.

Sur la serveuse choisie pour être le maître il faut aller modifier le fichier ossce.conf avec « nano /var/ossec/etc/ossec.conf »

Pour ce qu'il faut changer dans le fichier, le site de wazuh nous donne un tableau avec ce à quoi tout correspond :

name	It indicates the name of the cluster.



node name	It indicates the name of the current node. Each node of the cluster must have a unique name.
node type	It specifies the role of the node. It has to be set as worker.
<u>key</u>	The key created previously for the master node. It has to be the same for all the nodes.
nodes	It has to contain the address of the master node and can be either an IP or a DNS.
disabled	It indicates whether the node is enabled or disabled in the cluster. It has to be set to no.

Relancer le service wazuh-manager avec « systemctl restart wazuh-manager ».

Sur les Serveurs travailleurs il faut aussi modifier le fichier ossec.conf avec « nano /var/ossec/etc/ossec.conf »

De même que pour le maître, tout ce qu'il faut changer dans le fichier, le site de wazuh nous donne un tableau avec ce à quoi tout correspond:

<u>name</u>	It indicates the name of the cluster.
node name	It indicates the name of the current node. Each node of the cluster must have a unique name.
node_type	It specifies the role of the node. It has to be set as worker.
<u>key</u>	The key created previously for the master node. It has to be the same for all the nodes.
nodes	It has to contain the address of the master node and can be either an IP or a DNS.
<u>disabled</u>	It indicates whether the node is enabled or disabled in the cluster. It has to be set to no.

Et sur eux aussi il faut relancer le service wazuh-manager avec « systemctl restart wazuh-manager ».

Pour vérifier que tout marche il faut faire la commande « /var/ossec/bin/cluster_control -l »

Si tout se passe bien, cela doit donner ceci (Les informations vont changer suivant vos informations):

NAMETYPEVERSION ADDRESSmaster-node master4.12.010.0.0.3worker-node1 worker4.12.010.0.0.4worker-node2 worker4.12.010.0.0.5



7) Installation du Dashboard

Pour commencer l'installation du Wazuh Dashboard, il faut installer les paquets manquants « apt-get install debhelper tar curl libcap2-bin #debhelper version 9 or later ».

root@wazuh:~# apt-get install debhelper tar curl libcap2-bin #debhelper versi
on 9 or later

Comparer à la documentation trouvable sur le site de Wazuh on n'ajoute pas le répertoire Wazuh car on installe tous les services sur la même machine.

Si c'était le cas, voici les différentes commandes à effectuer :

- apt-get install gnupg apt-transport-https
- 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
- 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
- apt-get update

Installez les packages wazuh-dashboard « apt-get -y install wazuh-dashboard ».

root@wazuh:~# apt-get -y install wazuh-dashboard

Il faut modifier le fichier opensearch_dashboards.yml « nano /etc/wazuhdashboard/opensearch_dashboards.yml ».

root@wazuh:~# nano /etc/wazuh-dashboard/opensearch_dashboards.yml

Dans le fichier il faut remplacer les valeurs suivantes :

- server.host : 0.0.0.0, Ce paramètre spécifie l'hôte du serveur Wazuh-Dashboard. Pour permettre aux utilisateurs distants de se connecter, définissez la valeur de l'adresse IP ou le nom DNS du serveur Wazuh-Dashboard. La valeur 0.0.0.0 acceptera toutes les adresses IP disponibles de l'hôte.
- Opensearch.host : <u>https://localhost:9200</u>, ce paramètre spécifie le nœud wazuh-indexer. Si vous avez un seul serveur comme moi vous pouvez laisser localhost ou mettre l'adresse IP de la machine, Le Wazuh-dashboard peut être configuré pour se connecter à plusieurs nœuds d'indexeur Wazuh d'un même cluster ["https://10.0.0.2:9200", "https://10.0.0.3:9200"]).





Encore une fois, on va définir la variable NODE_NAME qui doit être le nom de notre nœud pour déployer les certificats « NODE_NAME=<SERVER_NODE_NAME> ».

root@wazuh:~# NODE NAME=node-1

Et comme déjà fait précédemment on fait les commandes suivantes une à une (possible de toutes les lancer en même temps mais moins fiable) pour déployer les certificats avec le nom du nœud.

« mkdir /etc/wazuh-dashboard/certs ».

root@wazuh:~# mkdir /etc/wazuh-dashboard/certs

« tar -xf ./wazuh-certificates.tar -C /etc/wazuh-dashboard/certs/ ./\$NODE_NAME.pem ./\$NODE_NAME-key.pem ./root-ca.pem »

root@wazuh:~# tar -xf ./wazuh-certificates.tar -C /etc/wazuh-dashboard/certs/ ./\$NODE_NAME.pem ./\$NODE_NAME-key.pem ./root-ca.pem

« mv -n /etc/wazuh-dashboard/certs/\$NODE_NAME.pem /etc/wazuhdashboard/certs/dashboard.pem »

root@wazuh:~# mv -n /etc/wazuh-dashboard/certs/\$NODE_NAME.pem /etc/wazuh-dashboard/certs/dashboard.pem

« mv -n /etc/wazuh-dashboard/certs/\$NODE_NAME-key.pem /etc/wazuhdashboard/certs/dashboard-key.pem »

root@wazuh:~# mv -n /etc/wazuh-dashboard/certs/\$NODE_NAME-key.pem /etc/wazuh-dashboard/certs/dashboard-key.pem

« chmod 500 /etc/wazuh-dashboard/certs »

root@wazuh:~# chmod 500 /etc/wazuh-dashboard/certs

« chmod 400 /etc/wazuh-dashboard/certs/* »

root@wazuh:~# chmod 400 /etc/wazuh-dashboard/certs*

« chown -R wazuh-dashboard:wazuh-dashboard /etc/wazuh-dashboard/certs »

root@wazuh:~# chown -R wazuh-dashboard:wazuh-dashboard /etc/wazuh-dashboard/certs

Une fois la configuration du serveur faite, il faut démarrer le service (wazuh-dashboard).

Recharger les daemons « systemctl daemon-reload »

root@wazuh:~# systemctl daemon-reload



Activer le service au démarrage « systemctl enable wazuh-dashboard »

root@wazuh:~# systemctl enable wazuh-dashboard

Et démarrer le service « systemctl start wazuh-dashboard »

root@wazuh:~# systemctl start wazuh-dashboard

Modifier le fichier wazuh.yml « nano /usr/share/wazuh-dashboard/data/wazuh/config/wazuh.yml »

```
root@wazuh:~# nano /usr/share/wazuh-dashboard/data/wazuh/config/wazuh.yml
```

Dans le fichier il faut changer la valeur suivante :

- url : <u>https://<WAZUH_SERVER_IP_ADDRESS</u>> en mettant l'adresse IP du wazuh dashboard (https://192.168.60.20).



Accédez à l'interface web de Wazuh avec vos identifiants utilisateur. Il s'agit du compte administrateur par défaut de l'indexeur Wazuh et il vous permet d'accéder au tableau de bord Wazuh.

- URL :https://<WAZUH_DASHBOARD_IP_ADDRESS>
- Nom d'utilisateur :admin
- Mot de passe :admin

w.	Overview				a
	AGENTS SUMMARY		LAST 24 HOU	IRS ALERTS	
	Active (6) Disconnected (3)	Critical severity 0 Rule level 15 or higher	High severity	Medium severity 1,271 Rule level 7 to 11	Low severity 2,094 Rule level 0 to 6
	ENDPOIN	T SECURITY		THREAT INTE	ILLIGENCE
£02	Configuration Assessment Scan your assets as part of a configuration assessment audit.	Check indicators of compromise triggered by malware infections or cyberattacks.	Threat Hunting Browse through you issues and threats in	ur security alerts, identifying n your environment.	Vulnerability Detection Discover what applications in your environment are affected by well-known vulnerabilities.
	File Integrity Monitoring Alerts related to file changes, including permissions, content, ownership, and attributes.		MITRE ATT&CK Explore security ale and techniques for I	rts mapped to adversary tactics better threat understanding.	
	SECURITY	OPERATIONS		CLOUD SE	CURITY
-1 [] -	PCI DSS Global security standard for entities that process, store, or transmit payment cardholder data.	GDPR General Data Protection Regulation (GDPR) sets guidelines for processing of personal data.	Monitor and collect containers such as a stopping or pausing	the activity from Docker creation, running, starting, events.	Amazon Web Services Security events related to your Amazon AWS services, collected directly via AWS API.
ŵ	HIPAA Health Insurance Portability and Accountability Act of 1996 (HIPAA) provides data privacy and security provisions for safeguarding medical information.	NIST 800-53 National Institute of Standards and Technology Special Publication 800-53 (NIST 800-53) sets guidelines for federal information systems.	Google Cloud Security events rela Platform services, c	ted to your Google Cloud ollected directly via GCP API.	GItHub Monitoring events from audit logs of your GitHub organizations.

8) Agent Wazuh



1.1. Linux

i. RPM amd64

Exécutez les commandes suivantes pour télécharger et installer l'agent :

curl -o wazuh-agent-4.11.2-1.x86_64.rpm https://packages.wazuh.com/4.x/yum/wazuh-agent-4.11.2-1.x86_64.rpm && sudo WAZUH_MANAGER=' IP-DU-SERVER-WAZUH ' WAZUH_AGENT_GROUP=' groupe-d'agent-wazuh ' WAZUH_AGENT_NAME=' Nom-de-l'agent-Wazuh ' rpm -ihv wazuh-agent-4.11.2-1.x86_64.rpm

Démarrer l'agent :

sudo systemctl daemon-reload sudo systemctl enable wazuh-agent

sudo systemctl start wazuh-agent

ii. DEB amd64

Exécutez les commandes suivantes pour télécharger et installer l'agent :

wget https://packages.wazuh.com/4.x/apt/pool/main/w/wazuh-agent/wazuh-agent_4.11.2-1_amd64.deb && sudo WAZUH_MANAGER=' IP-DU-SERVER-WAZUH ' WAZUH_AGENT_GROUP=' groupe-d'agent-wazuh ' WAZUH_AGENT_NAME=' Nom-de-l'agent-Wazuh ' dpkg -i ./wazuhagent_4.11.2-1_amd64.deb

Démarrer l'agent :

sudo systemctl daemon-reload

sudo systemctl enable wazuh-agent

sudo systemctl start wazuh-agent

iii. RPM aarch64

Exécutez les commandes suivantes pour télécharger et installer l'agent :

curl -o wazuh-agent-4.11.2-1.aarch64.rpm https://packages.wazuh.com/4.x/yum/wazuh-agent-4.11.2-1.aarch64.rpm && sudo WAZUH_MANAGER=' IP-DU-SERVER-WAZUH ' WAZUH_AGENT_GROUP=' groupe-d'agent-wazuh ' WAZUH_AGENT_NAME=' Nom-de-l'agent-Wazuh ' rpm -ihv wazuh-agent-4.11.2-1.aarch64.rpm

Démarrer l'agent :

sudo systemctl daemon-reload

sudo systemctl enable wazuh-agent



sudo systemctl start wazuh-agent

iv. DEB aarch64

Exécutez les commandes suivantes pour télécharger et installer l'agent :

wget https://packages.wazuh.com/4.x/apt/pool/main/w/wazuh-agent/wazuh-agent_4.11.2-1_arm64.deb && sudo WAZUH_MANAGER=' IP-DU-SERVER-WAZUH ' WAZUH_AGENT_GROUP=' groupe-d'agent-wazuh ' WAZUH_AGENT_NAME=' Nom-de-l'agent-Wazuh ' dpkg -i ./wazuhagent_4.11.2-1_arm64.deb

Démarrer l'agent : sudo systemctl daemon-reload sudo systemctl enable wazuh-agent sudo systemctl start wazuh-agent

1.2. Windows

Exécutez les commandes suivantes pour télécharger et installer l'agent :

Invoke-WebRequest -Uri https://packages.wazuh.com/4.x/windows/wazuh-agent-4.11.2-1.msi -OutFile \$env:tmp\wazuh-agent; msiexec.exe /i \$env:tmp\wazuh-agent /q WAZUH_MANAGER='IP-DU-SERVER-WAZUH' WAZUH_AGENT_GROUP='groupe-d'agent-wazuh' WAZUH_AGENT_NAME='Nom-de-l'agent-Wazuh'

Démarrer l'agent :

NET START WazuhSvc

III. Configuration

1) Active-response

Avec le logiciel Wazuh, il est possible de créer des règles permettant d'initialiser une réponse à un événement spécifique.

Dans mon cas, je vais montrer comment créer une règle qui met un blocage pare-feu pendant un temps donné.



Pour ce faire il faut d'abord connaître quelle règle nous voulons appliquer. Sur le site de wazuh il nous est mis à disposition quelques scripts utiles (scripts Linux et Windows diffèrents) :

https://documentation.wazuh.com/current/user-manual/capabilities/active-response/defaultactive-response-scripts.html

Puis il faut savoir quel ID nous devons utiliser pour détecter l'événement. Pour ce fait il faut aller dans « AGENTS SUMMARY » et aller sur « Active »



Aller dans un agent.

w.	Endpoints								a
		Active (6) Disconnected (3 Pending (0) Never connecte	3) rd (0)	toP 5 os debian (5) windows (3) ubuntu (1)			Linux (5) Windows (3) default (1)		
Agents (9)	Show only outda	ited			Deploy new agent	C Refresh	也 Export formatted	More ~	0
Search									WQL
□ ID ↑	Name	IP address	Group(s)	Operating system	Cluster node	Version	Status		Actions
001	VM-USER-WIN	192.168.60.60	Windows	Microsoft Windows 10 Pro 10.0.19045.5854	node01	v4.11.2	 disconnected () 	<	٥ ····
004	VM-USER-DEB	192.168.60.67	Linux	Debian GNU/Linux 12	node01	v4.11.2	active	0	۰۰۰ ا
005	TWINGATE	192.168.60.10	Linux	Debian GNU/Linux 12	node01	v4.11.2	active	c	©
007	SRV-WIN-2K25	192.168.60.40	Windows	Microsoft Windows Server 2025 Standard 10.0.26100.1742	node01	v4.11.2	 disconnected () 	0	©
008	VM-NEXTCLOUD	192.168.60.12	Linux	👌 Ubuntu 22.04.5 LTS	node01	v4.11.2	active	0	©
009	HomePage	192.168.60.21	Linux	A Debian GNU/Linux 12	node01	v4.11.2	active	(©
011	SRV-MDT-2K25	192.168.60.41	Windows	Microsoft Windows Server 2025 Standard 10.0.26100.1742	node01	v4.11.2	 disconnected () 	(©
012	Test-Debian 🖓	192.168.60.22	Linux	👌 Debian GNU/Linux 12	node01	v4.11.2	active	(ooo ()
013	proxmox	192.168.60.2	default	👌 Debian GNU/Linux 12	node01	v4.11.2	active	(<u>ه</u>
Rows per pag	ge: 10 🗸							<	1 >

Puis dans l'agent, il faut aller dans le menu « Threat Hunting »





Dans Threat Hunting, il faut aller dans le sous-menu « Events ».



Maintenant on peut voir tous les événements sur l'agent, notamment les « rule.id » à droite quand il y a une connexion SSH, l'id est 5710.

≡	E W. Threat Hunting Test-Debian										
Da	Dashboard Events										
	Search				DQL	🗮 🖌 Last 24 hours	Sh	ow dates	C Refresh		
man	manager.name: wazuh agent.id: 012 💿 🙃 Add filter										
Count	400 200 1500 1800 2100 0000 0000 1200										
۵	Export Formatted 🛛 👼 841 ava	ailable fields ۞ ≡ Columns ⊟ Density Φ	May 21 1 fields sorted Full screen	1,068 hits , 2025 @ 13:59:56.454 - May 22, 2025 @	13:59:56.454						
	ψ timestamp \lor	agent.name v	rule.description				~	rule.level \sim	rule.id \sim		
R	May 22, 2025 @ 13:15:16.3	Test-Debian	Listened ports status (netstat) cha	anged (new port opened or closed).				7	533		
ାର୍	May 22, 2025 @ 11:20:24.4	Test-Debian	Host-based anomaly detection ev	ent (rootcheck).				7	510		
R	May 22, 2025 @ 11:20:24.4	Test-Debian	Host-based anomaly detection ev	ent (rootcheck).				7	510		
R	May 22, 2025 @ 11:20:24.4	Test-Debian	Host-based anomaly detection ev	ent (rootcheck).				7	510		
R	May 22, 2025 @ 11:20:24.4	Test-Debian	Host-based anomaly detection ev	ent (rootcheck).				7	510		
ାର୍ଯ୍	May 22, 2025 @ 11:20:24.4	Test-Debian	Host-based anomaly detection ev	ent (rootcheck).				7	510		
ାର୍	May 22, 2025 @ 11:20:24.4	Test-Debian	Host-based anomaly detection ev	ent (rootcheck).				7	510		
R	May 22, 2025 @ 11:20:24.4	Test-Debian	Host-based anomaly detection ev	ent (rootcheck).				7	510		
ାର୍	May 22, 2025 @ 11:20:24.4	Test-Debian	Host-based anomaly detection ev	ent (rootcheck).				7	510		
Q	May 22, 2025 @ 11:20:24.4	Test-Debian	Host-based anomaly detection ev	ent (rootcheck).				7	510		
ଘ	May 22, 2025 @ 11:20:24.4	Test-Debian	Host-based anomaly detection ev	ent (rootcheck).				7	510		

Maintenant, il faut aller dans le menu « Server management », puis dans le sous-menu « Settings ».



Dans les Settings, il faut en haut à droite dans « Edit configuration ».

Configuration Main configurations		G Refresh & Edit configuration @
Name	Description	
Global Configuration	Global and remote settings	
Cluster	Master node configuration	
Registration Service	Automatic agent registration service	
Alerts and output management		
Name	Description	
Alerts	Settings related to the alerts and their format	
Integrations	Slack, VirusTotal and PagerDuty Integrations with external APIs	
Auditing and policy monitoring		
Name	Description	
Policy monitoring	Configuration to ensure compliance with security policies, standards and hardening guides	
OpenSCAP	Configuration assessment and automation of compliance monitoring using SCAP checks	
CIS-CAT	Configuration assessment using CIS scanner and SCAP checks	

Est chercher la partie où il est écrit « active-response ». Une fois au bon endroit, il faut écrire le texte suivant.



<active-response> <command>firewall-drop</command> <location>local</location> <rules_id>5710</rules_id> <timeout>180</timeout> </active-response>

Après avoir écrit le texte il faut « Save » puis « Restart Manager ».



Maintenant les machines avec des agent dessus sont protéger contre les tentatives de brut force en SSH.

2) Quoi de plus

2.1. Vulnerabilty-dectetion

La detection de vulnérabilité permet d'identifier les failles de sécurité connues présentes sur les systèmes surveillés. Grâce à son module dédié, Wazuh analyse les paquets logiciels installés sur les agents (serveurs ou postes clients) et les compare à des bases de données publiques de vulnérabilités (comme NVD, OVAL, etc.).

Objectifs principaux :



- Détecter les logiciels obsolètes ou vulnérables.
- Fournir une visibilité centralisée sur les risques présents dans l'environnement.
- Aider les équipes IT à prioriser les mises à jour de sécurité.
- Renforcer la posture de sécurité globale de l'infrastructure.

Cette fonctionnalité est essentielle pour **prévenir les attaques** exploitant des failles connues, et s'inscrit dans une démarche proactive de **gestion des risques**.

2.2 intégrations de slack

🔔 Pourquoi utiliser Slack avec Wazuh ?

Intégrer **Slack** avec **Wazuh** permet de **recevoir en temps réel les alertes de sécurité** directement dans vos canaux de discussion. Cela transforme Slack en **centre de supervision réactif**, sans avoir à surveiller constamment l'interface Wazuh.

Avantages de l'intégration Wazuh + Slack :

- **• Notifications instantanées** : Recevez des alertes critiques (intrusions, vulnérabilités, changements de fichiers) dans Slack dès qu'elles sont détectées.
- **L Réaction rapide** : L'équipe peut discuter et prendre des décisions immédiatement sur l'alerte, sans changer d'outil.
- **Automatisation** : Grâce à des scripts ou Webhooks, vous pouvez configurer quelles alertes sont transmises, pour ne voir que l'essentiel.
- Historique centralisé : Slack garde une trace des alertes échangées, pratique pour les audits ou les suivis.

The Second Seco

Loïc Corneloup SIO25

