BRETON Théo

PROJET N°10 : MON HOMELAB HEBERGER SOUS PROXMOX SUPERVISER PAR WAZUH

Qu’est que Wazuh ?

Wazuh est une plateforme de sécurité informatique complète, open-source qui combine les fonctions de gestion des informations et des événements de sécurité) et de détection et réponse étendues.

Elle dispose de nombreuse fonctionalités :

- **Analyse des journaux (Logs) : Collecte et analyse en temps réel des données et des journaux d'activité provenant des systèmes d'exploitation et des applications.**
- **Détection d'intrusions (HIDS) : Surveillance des fichiers du système pour repérer les modifications non autorisées ou suspectes (surveillance de l'intégrité des fichiers ou FIM).**
- **Gestion des vulnérabilités :**
- **Analyse continue des logiciels installés pour identifier les failles de sécurité connues (CVE).**
- **Conformité réglementaire : Aide à respecter les normes de sécurité en vérifiant la configuration des équipements.**

Pourquoi j’ai décidé d’intégrer Wazuh à mon homelab ?

Ce que Wazuh va apporter dans mon homelab :

Détection d'intrusion et d'anomalies

C’est tout d’abord une analyse des logs système et applicatifs en temps réel (connexions SSH, authentifications échouées, escalades de privilèges). Particulièrement utile sur votre VLAN 11 DMZ (marquée "SERVEUR SENSIBLE") et le VLAN 10 SiteWeb, exposés à internet via votre tunnel chiffré.

File Integrity Monitoring (FIM)

Alerte si un fichier système critique est modifié, ajouté ou supprimé sans autorisation. Très pertinent sur mes serveurs Docker (VLAN 6, VLAN 11) où des conteneurs compromis pourraient altérer des fichiers hôtes

Détection de vulnérabilités

Scanne les paquets installés sur mes serveurs (Ubuntu, Windows, Debian) et les compare à des bases de CVE connues. Utile sur tout votre parc : WindowsServer BDD, SERV-WEB, SERV-DOCKER, etc.

Conformité et durcissement (CIS Benchmarks)

Audits automatiques de configuration : mots de passe faibles, services inutiles actifs, permissions trop larges. Génère des rapports PCI DSS, GDPR, NIST si besoin.

Détection au niveau réseau/DMZ

Sur mon Nginx Proxy Manager et le VLAN 11, Wazuh peut détecter des attaques (scans de ports, tentatives de brute-force, injections) en analysant les logs Nginx.

Etape 1 : Création sous interface sur le Fortigate– VLAN 12 – Wazuh :

Status	Name	Members	IP/Netmask	Type	Access	Ref.
Physical (4)						
	dmz		[redacted]	Physical	PING HTTPS HTTP FMG-Access CAPWAP	3
	wan1		[redacted]	Physical	PING FMG-Access AUTO-IPSEC	18
	VPN_Homelab		[redacted]	VPN Tunnel		1
	wan2		[redacted]	Physical	PING FMG-Access AUTO-IPSEC	0
Software Switch (10)						
	lan	wifi (SSID: fortinet), internal	[redacted]	Software Switch (2)	PING HTTPS SSH SNMP HTTP FMG-Access CAPWAP	17
	VLAN 4 - BDD		192.168.4.1 255.255.255.0	VLAN		9
	VLAN 5 - TEL.		192.168.5.1 255.255.255.0	VLAN		6
	VLAN 6 - DOCKER		192.168.6.1 255.255.255.0	VLAN		10
	VLAN 7 - ZABBIX		192.168.7.1 255.255.255.0	VLAN		8
	VLAN 8 - HA		192.168.8.1 255.255.255.0	VLAN		6
	VLAN 9_ADGuard		192.168.9.1 255.255.255.0	VLAN		8
	VLAN 11 DMZ		192.168.11.1 255.255.255.0	VLAN		12
	VLAN 12 - Wazuh		192.168.12.1 255.255.255.0	VLAN		1
	VLAN10 SiteWeb		192.168.10.1 255.255.255.0	VLAN		10

Etape 2 : Création des règles de filtrage

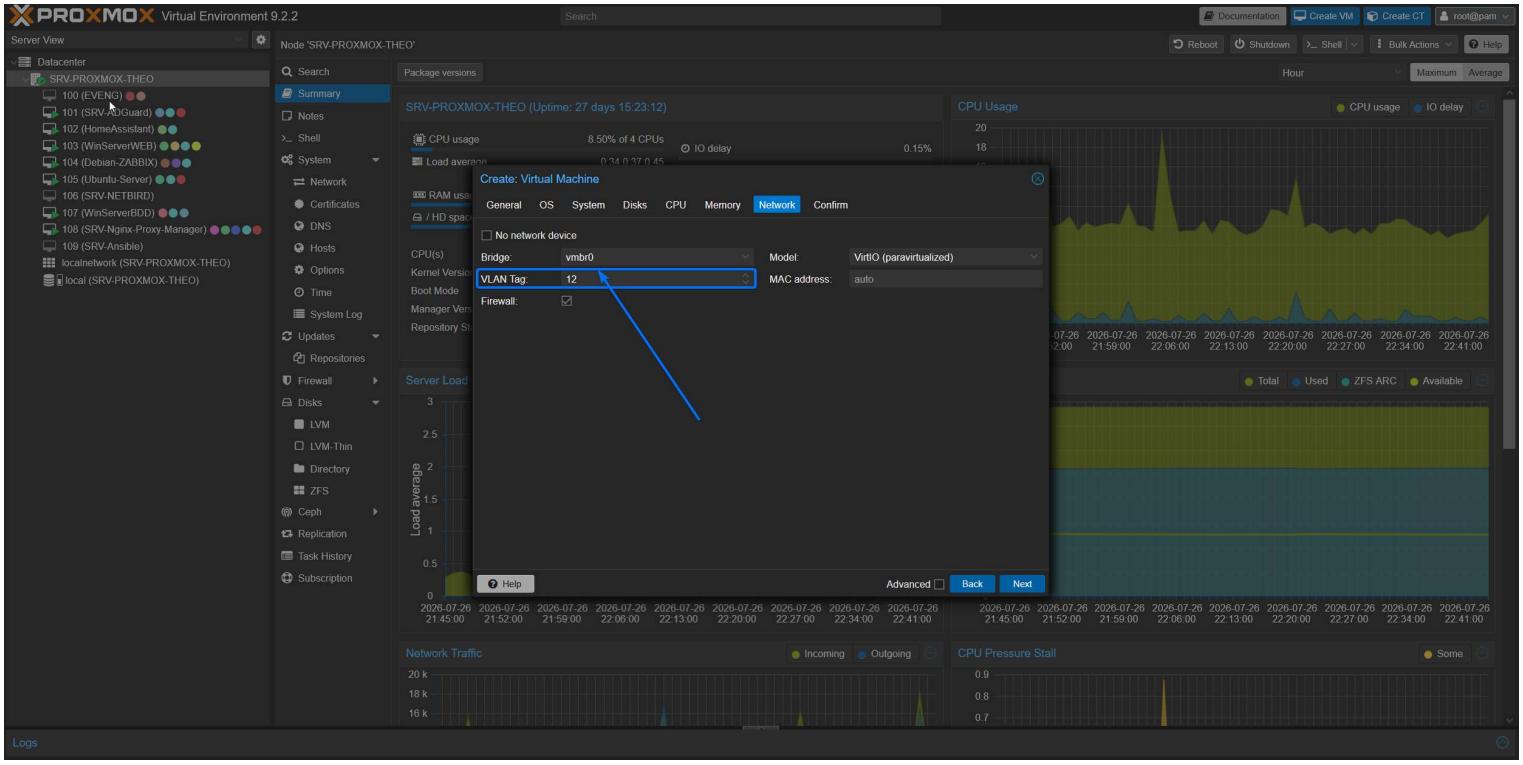
Soit 4 règles de filtrage à mettre en place

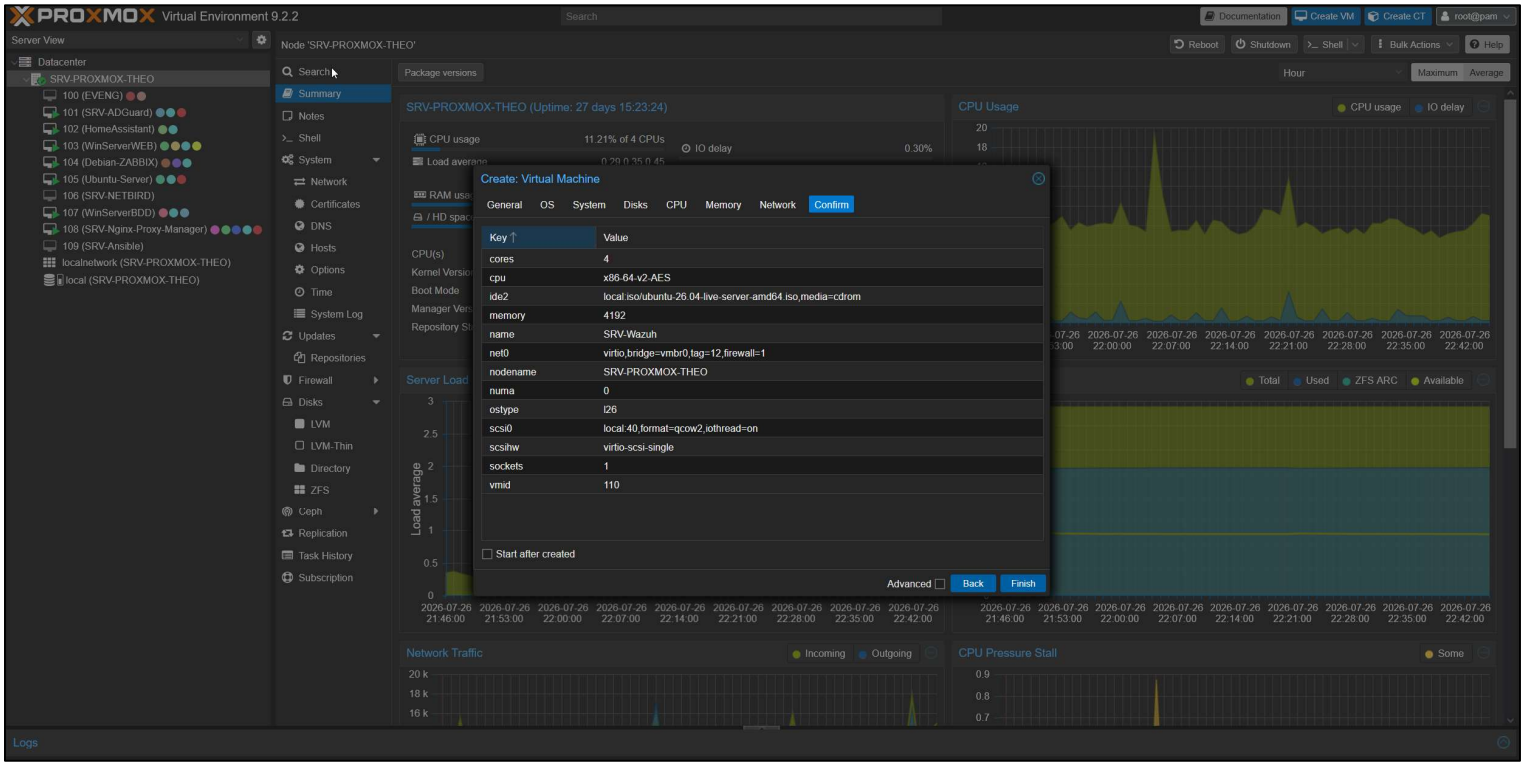
- Accès au WAN (Aller-retour)
- Accès à l'ensemble des VLANs (Aller-retour)

Etape 3 : Création du vlan sur le Switch Cisco :

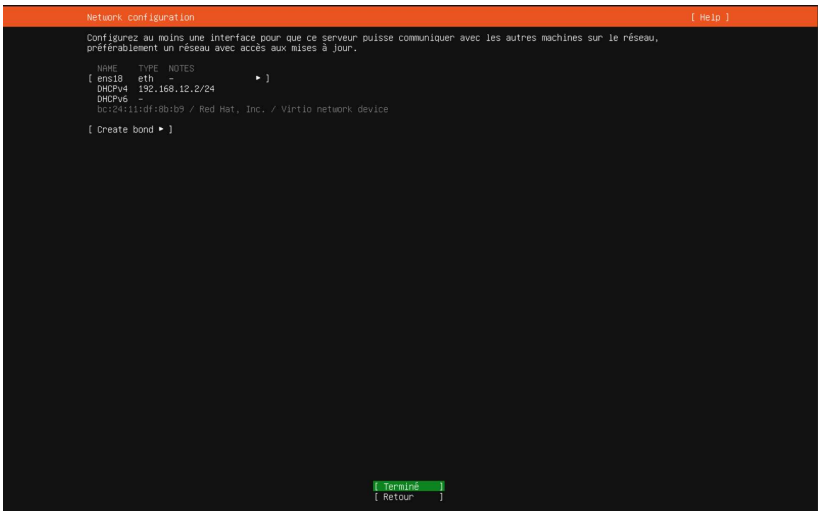
```
SW-HOMELAB_THEO(config)#vlan 12
SW-HOMELAB_THEO(config-vlan)#name Reseau_Wazuh
SW-HOMELAB_THEO(config-vlan)#exit
```

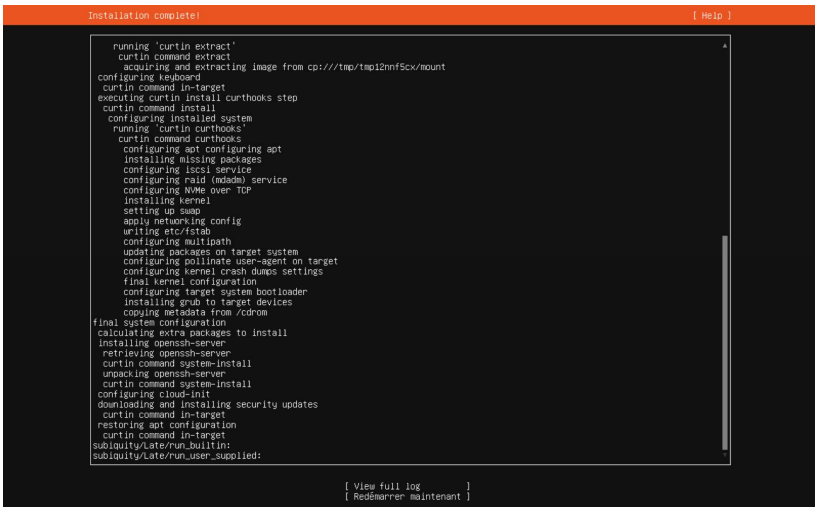
Etape 4 : Création de la machine virtuelle sur Proxmox :





Etape 5 : Installation de la machine virtuelle :





Etape 6 : Installation manuelle de Wazuh :

Installation du dépôt :

```
curl -s https://packages.wazuh.com/key/GPG-KEY-WAZUH | sudo gpg --no-default-keyring
--keyring gnupg-ring:/usr/share/keyrings/wazuh.gpg --import
sudo 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" | sudo tee
/etc/apt/sources.list.d/wazuh.list
sudo apt update
```

Installation de l'indexer :

```
sudo apt install wazuh-indexer -y
```

Installation de wazuh-manager :

```
sudo apt install wazuh-manager -y
```

Installation de flibeat :

```
sudo apt install filebeat -y
```

Installation de dashbord :

```
sudo apt install wazuh-dashboard -y
```

Si une commande échoue (erreur réseau/disque ponctuelle), la relancer seule suffit :

```
sudo apt install --reinstall <nom-du-paquet> -y
```

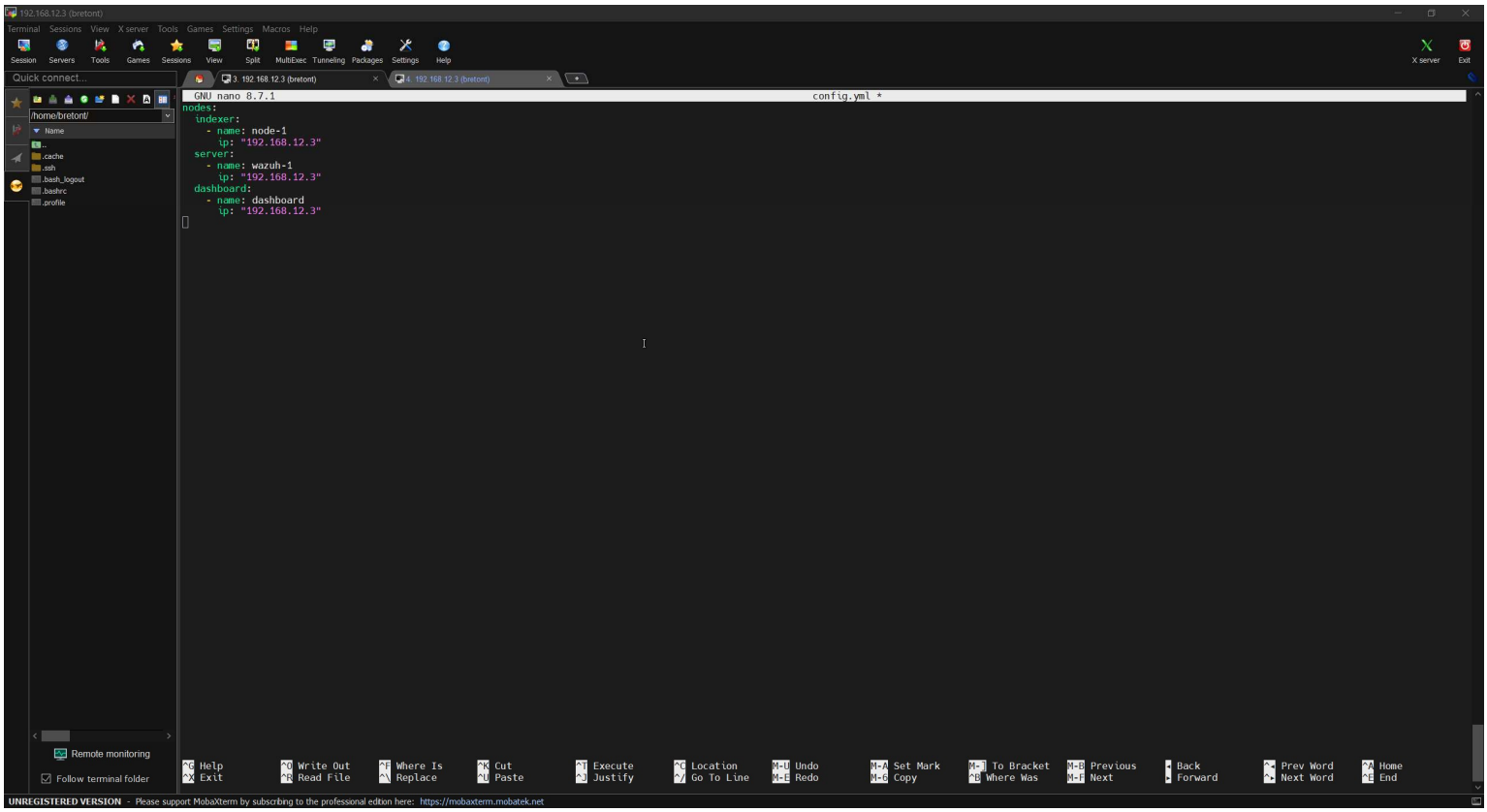
Etape 7 : Installation manuelle de Wazuh – Certificats :

Généré les certificats :

```
cd ~
curl -sO https://packages.wazuh.com/4.14/wazuh-certs-tool.sh
curl -sO https://packages.wazuh.com/4.14/config.yml
```

Éditer config.yml avec l'IP réelle du serveur (remplacer <indexer-node-ip> etc. par la même IP pour un déploiement single-node) :

```
nodes:
  indexer:
    - name: node-1
      ip: "192.168.12.3"
  server:
    - name: wazuh-1
      ip: "192.168.12.3"
  dashboard:
    - name: dashboard
      ip: "192.168.12.3"
```



Avertissement : le script wazuh-certs-tool.sh **doit être lancé avec l'option -A**, sinon il n'affiche que l'aide et ne génère aucun certificat.

```
sudo bash ./wazuh-certs-tool.sh -A
```

Vérifier que le dossier ./wazuh-certificates/ contient bien :

- node-1.pem
- node-1-key.pem
- wazuh-1.pem
- wazuh-1-key.pem
- dashboard.pem
- dashboard-key.pem
- admin.pem
- admin-key.pem
- root-ca.pem
- root-ca.key.

```
sudo ls -la ./wazuh-certificates/
```

```
bretont@srv-wazuh:~$ sudo ls -la ./wazuh-certificates/
[sudo: authenticate] Password:
total 48
drwx------ 2 root    root    4096 Jul 26 21:57 .
drwxr-x--- 6 bretont bretont 4096 Jul 26 21:57 ..
-rw------- 1 root    root    1704 Jul 26 21:57 admin-key.pem
-rw------- 1 root    root    1220 Jul 26 21:57 admin.pem
-rw------- 1 root    root    1704 Jul 26 21:57 dashboard-key.pem
-rw------- 1 root    root    1281 Jul 26 21:57 dashboard.pem
-rw------- 1 root    root    1704 Jul 26 21:57 node-1-key.pem
-rw------- 1 root    root    1277 Jul 26 21:57 node-1.pem
-rw------- 1 root    root    1704 Jul 26 21:57 root-ca.key
-rw------- 1 root    root    1204 Jul 26 21:57 root-ca.pem
-rw------- 1 root    root    1704 Jul 26 21:57 wazuh-1-key.pem
-rw------- 1 root    root    1277 Jul 26 21:57 wazuh-1.pem
bretont@srv-wazuh:~$
```

Déployer les certificats — Indexer :

```
NODE_NAME=node-1
sudo mkdir -p /etc/wazuh-indexer/certs
sudo cp ~/wazuh-certificates/$NODE_NAME.pem /etc/wazuh-indexer/certs/indexer.pem
sudo cp ~/wazuh-certificates/$NODE_NAME-key.pem /etc/wazuh-indexer/certs/indexer-
key.pem
sudo cp ~/wazuh-certificates/admin.pem /etc/wazuh-indexer/certs/
sudo cp ~/wazuh-certificates/admin-key.pem /etc/wazuh-indexer/certs/
sudo cp ~/wazuh-certificates/root-ca.pem /etc/wazuh-indexer/certs/
sudo chmod 500 /etc/wazuh-indexer/certs
sudo chmod 400 /etc/wazuh-indexer/certs/*
sudo chown -R wazuh-indexer:wazuh-indexer /etc/wazuh-indexer/certs
```

Note : Le chmod 400 .../certs/* renvoie impossible d'accéder à '.../certs/*' : Aucun fichier ou répertoire de ce type, ce n'est généralement pas grave. En collant plusieurs commandes d'un coup, le chmod peut s'exécuter une fraction de seconde avant que le dernier cp ait fini d'écrire son fichier — le * ne le "voit" alors pas encore.

Vérifier avec **sudo ls -la /etc/wazuh-indexer/certs/** : si tous les fichiers attendus sont présents avec les permissions 400 et le bon propriétaire, tout est en ordre malgré le message d'erreur.

Démarrer l'indexer et initialiser la sécurité :

```
sudo systemctl daemon-reload
sudo systemctl enable wazuh-indexer
sudo systemctl start wazuh-indexer
sudo systemctl status wazuh-indexer
sudo /usr/share/wazuh-indexer/bin/indexer-security-init.sh
```

Test :

```
curl -k -u admin:admin https://localhost:9200
```

```
bretont@srv-wazuh:~$ curl -k -u admin:admin https://localhost:9200
{
  "name" : "node-1",
  "cluster_name" : "wazuh-cluster",
  "cluster_uuid" : "1dJXEkvLREa9QGgZFXilYQ",
  "version" : {
    "number" : "7.10.2",
    "build_type" : "deb",
    "build_hash" : "3eb218fe481c2ffce6aaf7ec381d1f3ec709102a",
    "build_date" : "2026-06-25T17:02:43.553291169Z",
    "build_snapshot" : false,
    "lucene_version" : "9.12.3",
    "minimum_wire_compatibility_version" : "7.10.0",
    "minimum_index_compatibility_version" : "7.0.0"
  },
  "tagline" : "The OpenSearch Project: https://opensearch.org/"
}
bretont@srv-wazuh:~$
```

Déployer les certificats — Manager (Filebeat) :

```
NODE_NAME=wazuh-1
sudo mkdir -p /etc/filebeat/certs
sudo cp ~/wazuh-certificates/$NODE_NAME.pem /etc/filebeat/certs/filebeat.pem
sudo cp ~/wazuh-certificates/$NODE_NAME-key.pem /etc/filebeat/certs/filebeat-key.pem
sudo cp ~/wazuh-certificates/root-ca.pem /etc/filebeat/certs/root-ca.pem
sudo chmod 500 /etc/filebeat/certs
sudo chmod 400 /etc/filebeat/certs/*
sudo chown -R root:root /etc/filebeat/certs
```

Note : même remarque qu'à l'étape 4 si chmod renvoie une erreur ici — vérifier avec sudo ls -la /etc/filebeat/certs/ avant.

Etape 7 : Démarrage du manager :

```
sudo systemctl daemon-reload
sudo systemctl enable wazuh-manager
sudo systemctl start wazuh-manager
sudo systemctl status wazuh-manager
```

Etape 8 : Configurer Filebeat :

```
sudo curl -so /etc/filebeat/filebeat.yml https://packages.wazuh.com/4.14/tpl/wazuh/filebeat/filebeat.yml
```

 **Avertissement :** le chemin correct utilise le numéro de version exact (4.14), pas 4.x. **Une mauvaise URL** renvoie **une page d'erreur XML** au lieu du fichier YAML — vérifier systématiquement avec **sudo cat /etc/filebeat/filebeat.yml après le téléchargement.**

```
sudo filebeat keystore create
echo admin | sudo filebeat keystore add username --stdin --force
echo admin | sudo filebeat keystore add password --stdin --force

sudo curl -so /etc/filebeat/wazuh-template.json
https://raw.githubusercontent.com/wazuh/wazuh/v4.14.6/extensions/elasticsearch/7.x/wazuh-
template.json
sudo chmod go+r /etc/filebeat/wazuh-template.json

sudo curl -s https://packages.wazuh.com/4.x/filebeat/wazuh-filebeat-0.5.tar.gz | sudo tar -xvz -C
/usr/share/filebeat/module
```

Configurer les identifiants de test indexer côté manager (nouveauté Wazuh 4.14 : communication directe manager ↔ indexer) :

```
echo 'admin' | sudo /var/ossec/bin/wazuh-keystore -f indexer -k username
echo 'admin' | sudo /var/ossec/bin/wazuh-keystore -f indexer -k
password
```

Avertissement : Point d'attention IP/certificat : le **fichier filebeat.yml** pointe **par défaut vers 127.0.0.1:9200**, mais le certificat de l'indexer n'est **valide que pour l'IP indiquée dans config.yml**. Si les deux ne correspondent pas, le handshake TLS échoue (x509: certificate is valid for <IP>, not 127.0.0.1). Corriger si besoin :

```
sudo sed -i 's/127.0.0.1:9200/192.168.12.3:9200/'
/etc/filebeat/filebeat.yml
```

Démarrer et tester :

```
sudo systemctl daemon-reload
sudo systemctl enable filebeat
sudo systemctl start filebeat
sudo filebeat test output
```

→ Toutes les lignes doivent afficher OK, notamment handshake... OK et talk to server... OK.

Etape 9 : Déployer le dashbord :

```
NODE_NAME=dashboard
sudo mkdir -p /etc/wazuh-dashboard/certs
sudo cp ~/wazuh-certificates/$NODE_NAME.pem /etc/wazuh-
dashboard/certs/dashboard.pem
sudo cp ~/wazuh-certificates/$NODE_NAME-key.pem /etc/wazuh-
dashboard/certs/dashboard-key.pem
sudo cp ~/wazuh-certificates/root-ca.pem /etc/wazuh-
dashboard/certs/root-ca.pem
sudo chmod 500 /etc/wazuh-dashboard/certs
sudo chmod 400 /etc/wazuh-dashboard/certs/*
sudo chown -R wazuh-dashboard:wazuh-dashboard /etc/wazuh-
dashboard/certs
```

Note : même remarque qu'à l'étape 4 si chmod renvoie une erreur ici — vérifier avec sudo ls -la /etc/filebeat/certs/ avant.

Adapter la configuration du dashboard (même piège IP/certificat)

```
sudo grep -E "server.host|server.port|opensearch.hosts"
/etc/wazuh-dashboard/opensearch_dashboards.yml
```

Si opensearch.hosts pointe vers localhost alors que le certificat est émis pour une IP spécifique, corriger :

```
sudo sed -i 's|https://localhost:9200|https://192.168.12.3:9200|'
/etc/wazuh-dashboard/opensearch_dashboards.yml
```

Etape 9 : Démarrer le dashbord :

```
sudo systemctl daemon-reload
sudo systemctl enable wazuh-dashboard
sudo systemctl start wazuh-dashboard
sudo systemctl status wazuh-dashboard
```

Vérification effectuée pour s'assurer qu'il ne redémarre pas en boucle (restart counter qui augmente = problème). En cas de souci, consulter les logs détaillés :

```
sudo journalctl -u wazuh-dashboard -n 100 --no-pager
```

Etape 10 : Accéder à l'interface :

```
curl -k -I https://localhost
```

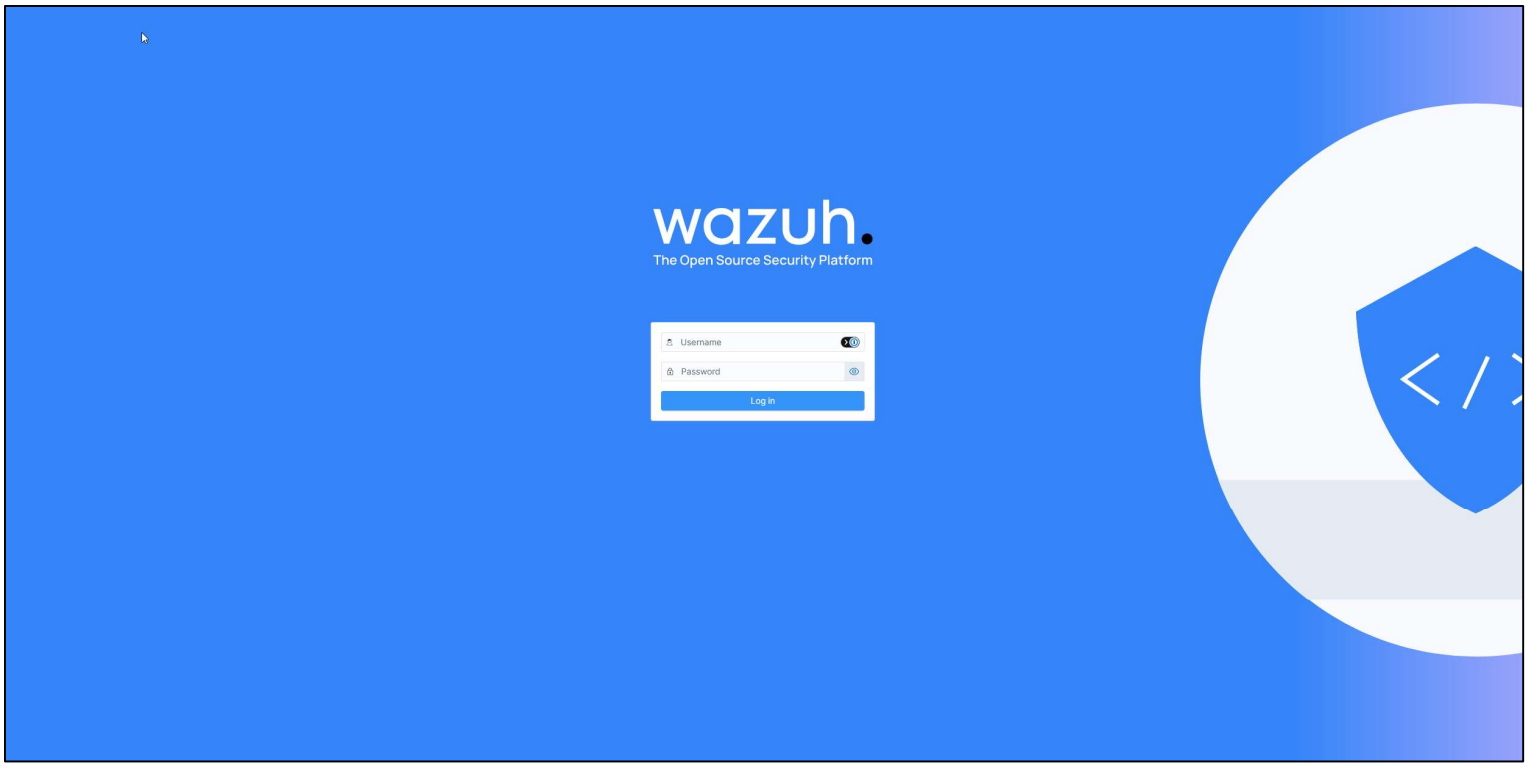
→ Doit renvoyer HTTP/1.1 302 Found avec location: /app/login.

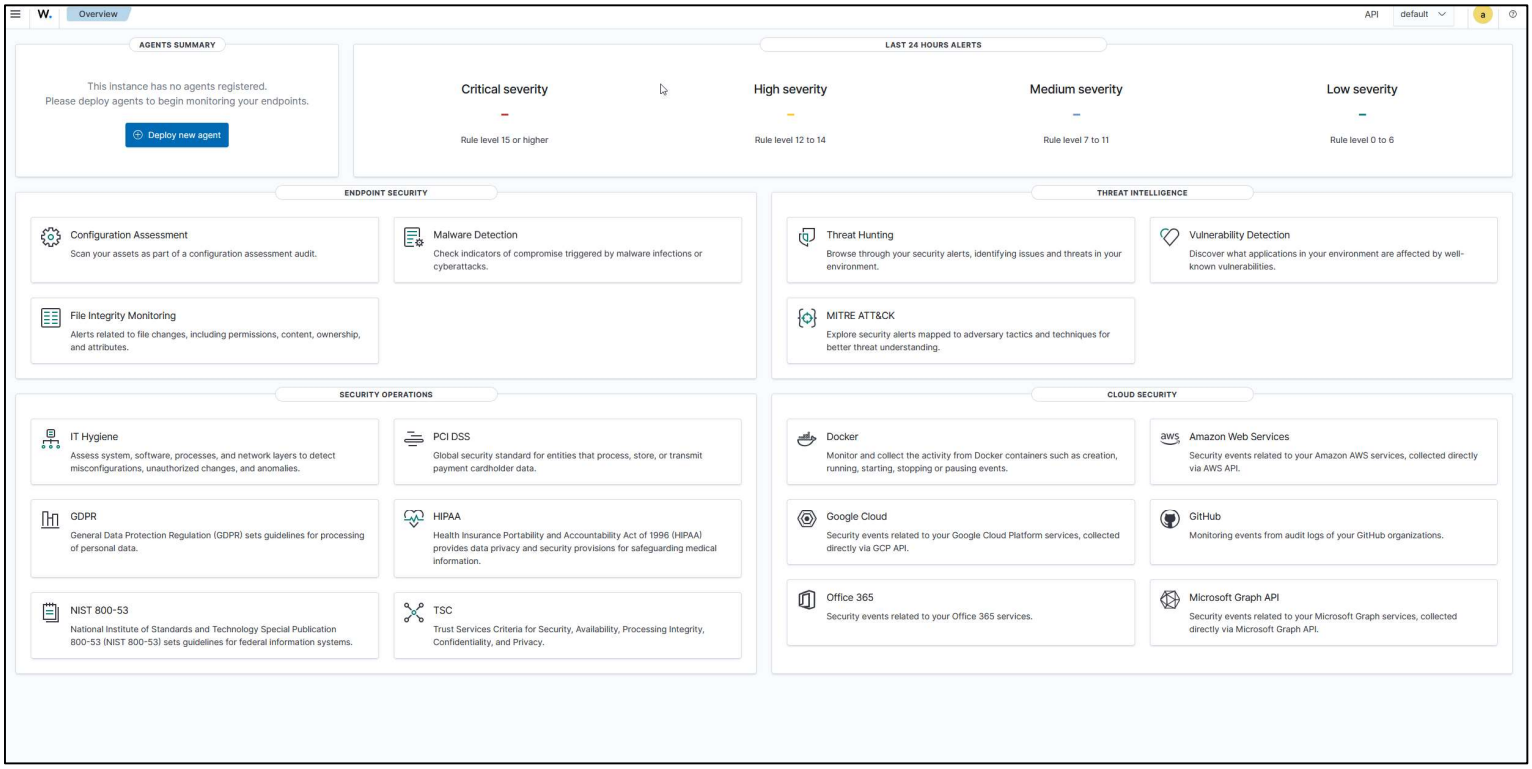
Puis dans un navigateur :

https://<IP_DU_SERVEUR>

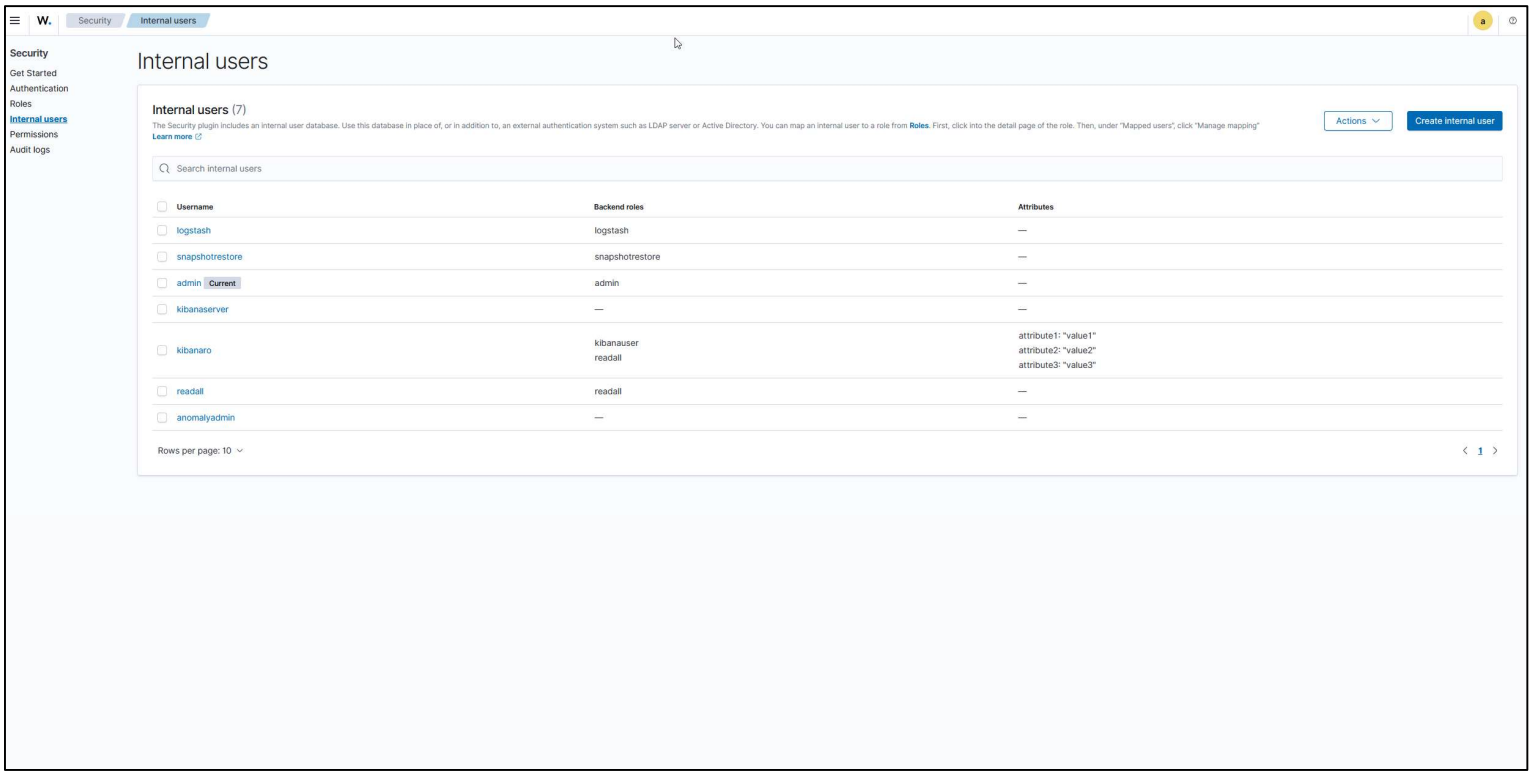
Identifiants par défaut :

- **Utilisateur** : admin
- **Mot de passe** : admin


wazuh.
The Open Source Security Platform
Username
Password
Log in


Overview
API
default
AGENTS SUMMARY
This instance has no agents registered.
Please deploy agents to begin monitoring your endpoints.
Deploy new agent
LAST 24 HOURS ALERTS
Critical severity
Rule level 15 or higher
High severity
Rule level 12 to 14
Medium severity
Rule level 7 to 11
Low severity
Rule level 0 to 6
ENDPOINT SECURITY
Configuration Assessment
Scan your assets as part of a configuration assessment audit.
Malware Detection
Check indicators of compromise triggered by malware infections or cyberattacks.
File Integrity Monitoring
Alerts related to file changes, including permissions, content, ownership, and attributes.
THREAT INTELLIGENCE
Threat Hunting
Browse through your security alerts, identifying issues and threats in your environment.
Vulnerability Detection
Discover what applications in your environment are affected by well-known vulnerabilities.
MITRE ATT&CK
Explore security alerts mapped to adversary tactics and techniques for better threat understanding.
SECURITY OPERATIONS
IT Hygiene
Assess system, software, processes, and network layers to detect misconfigurations, unauthorized changes, and anomalies.
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.
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.
TSC
Trust Services Criteria for Security, Availability, Processing Integrity, Confidentiality, and Privacy.
CLOUD SECURITY
Docker
Monitor and collect the activity from Docker containers such as creation, running, starting, stopping or pausing events.
Amazon Web Services
Security events related to your Amazon AWS services, collected directly via AWS API.
Google Cloud
Security events related to your Google Cloud Platform services, collected directly via GCP API.
GitHub
Monitoring events from audit logs of your GitHub organizations.
Office 365
Security events related to your Office 365 services.
Microsoft Graph API
Security events related to your Microsoft Graph services, collected directly via Microsoft Graph API.

Depuis l'interface web nous pouvons changer le mot des passe des utilisateurs et en ajouter.



Etape 11 : Déploiement massif des agents sur l'ensemble des serveurs du homelab :

Windows

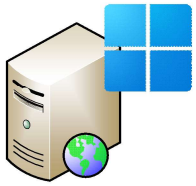
Un fichier .msi, installable en silencieux via PowerShell :

```
Invoke-WebRequest -Uri https://packages.wazuh.com/4.x/windows/wazuh-agent-4.14.6-1.msi -OutFile $env:tmp\wazuh-agent.msi
msiexec.exe /i $env:tmp\wazuh-agent.msi /q WAZUH_MANAGER="192.168.12.3" WAZUH_AGENT_NAME="win-serveur01"
NET START WazuhSvc
```

Linux :

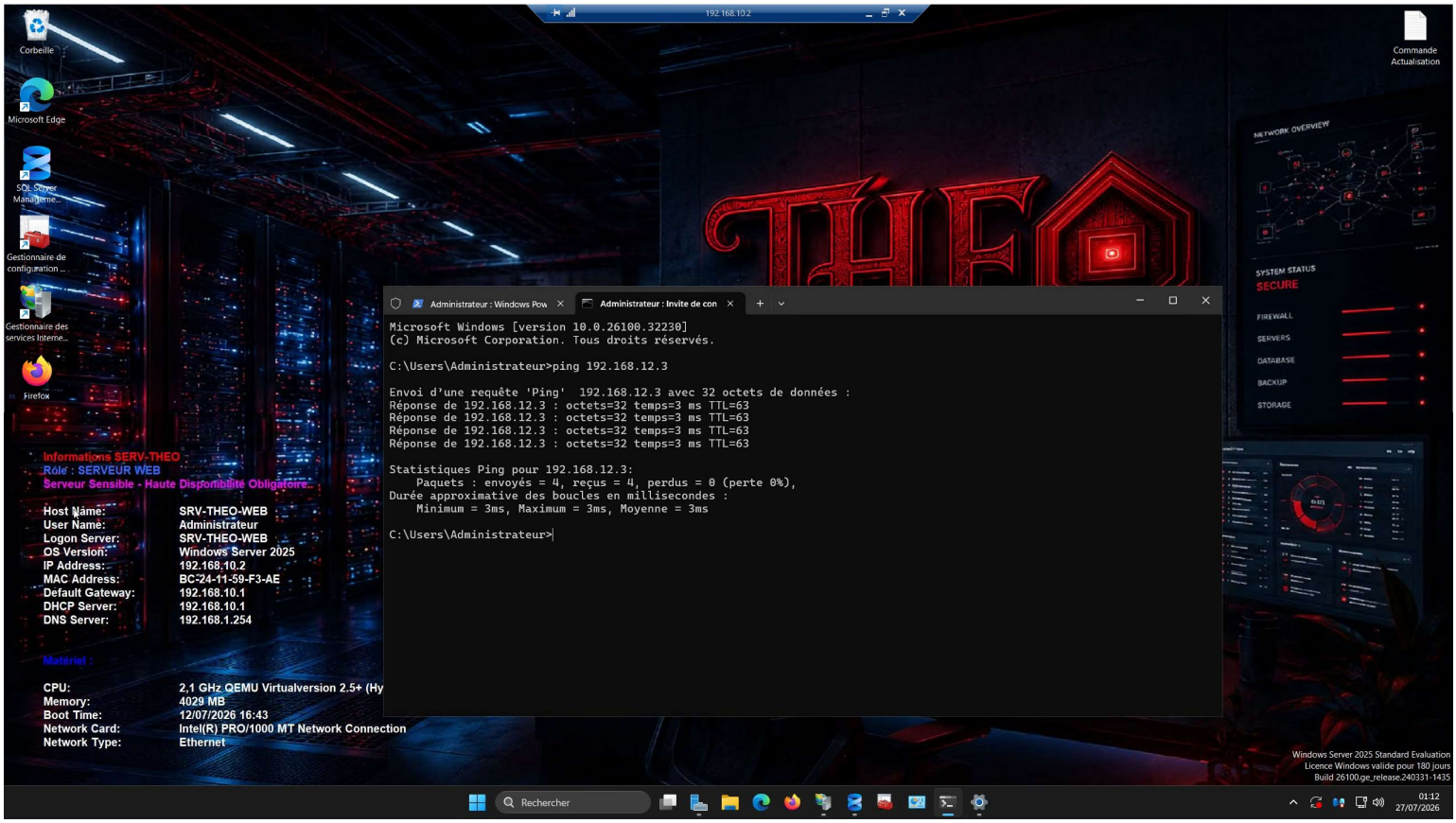
```
wget https://packages.wazuh.com/4.x/apt/pool/main/w/wazuh-agent/wazuh-agent_4.14.6-1_amd64.deb
sudo WAZUH_MANAGER='192.168.12.3' WAZUH_AGENT_NAME='NOMSERVEUR' dpkg -i ./wazuh-agent_4.14.6-1_amd64.deb
sudo systemctl daemon-reload
sudo systemctl enable wazuh-agent
sudo systemctl start wazuh-agent
```

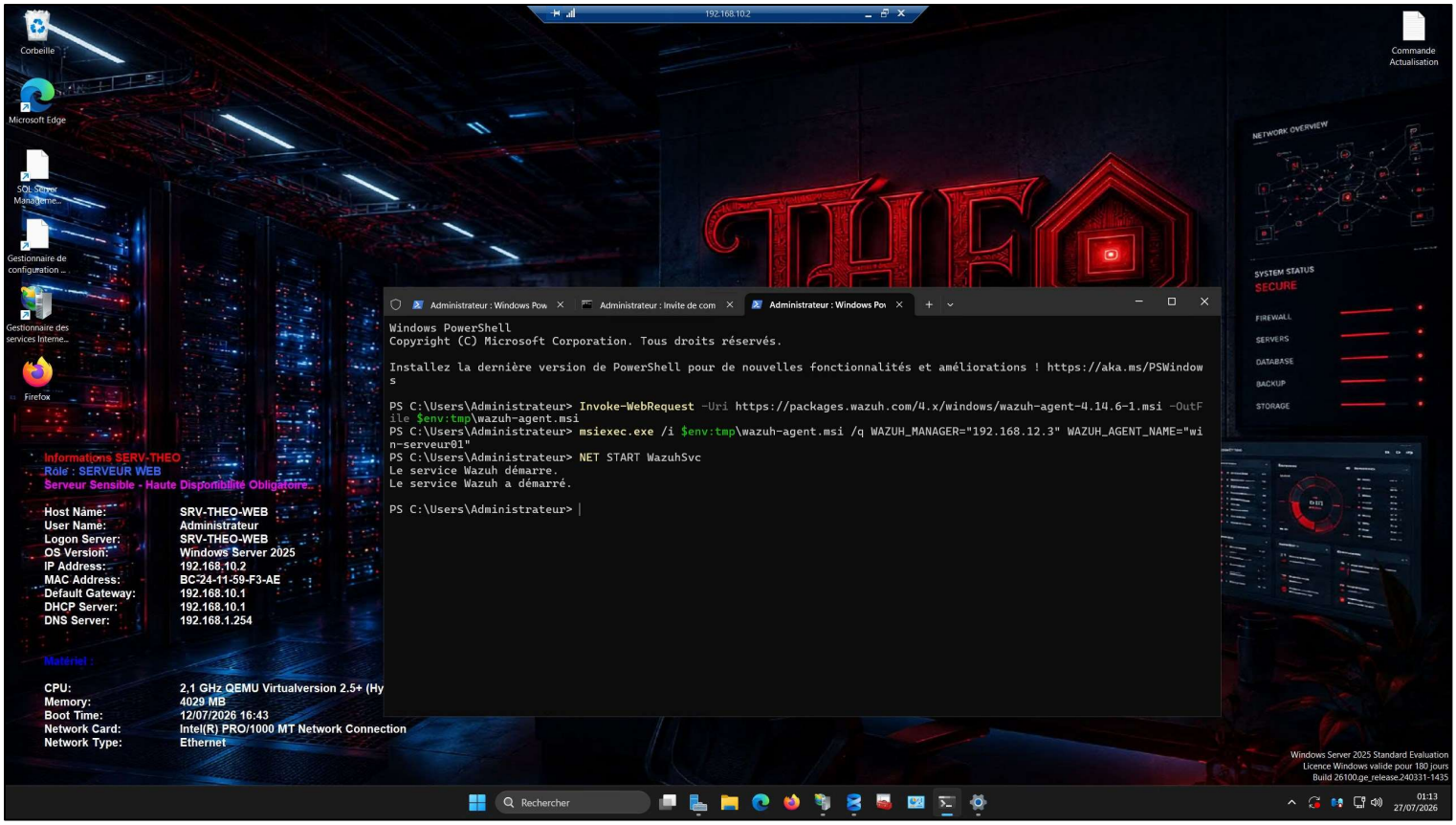
SERVEURS WINDOWS :



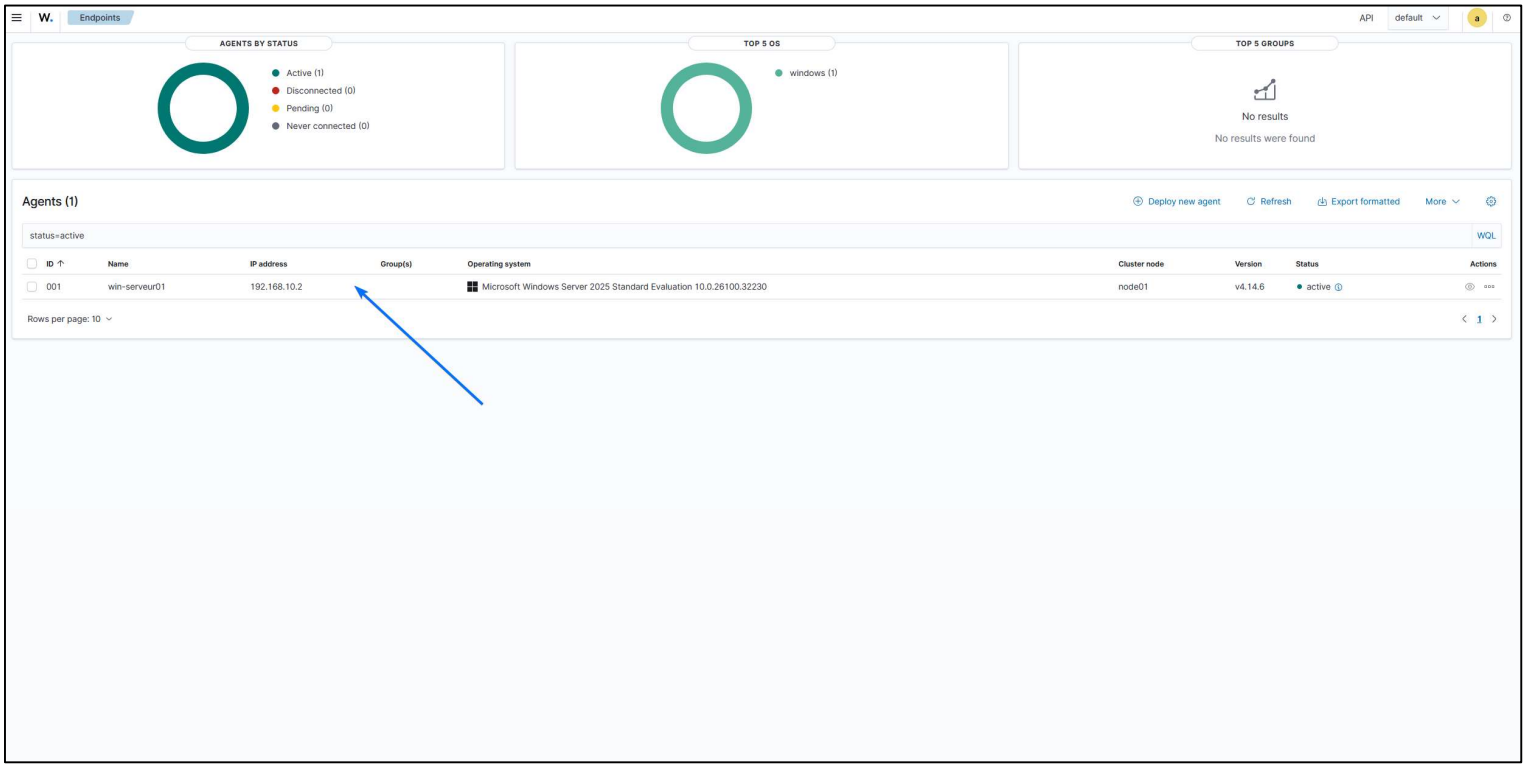
SRV-WEB :

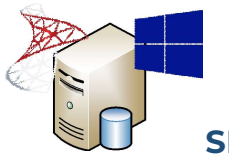
Vérifier la bonne communication avec le SRV-WAZUH :



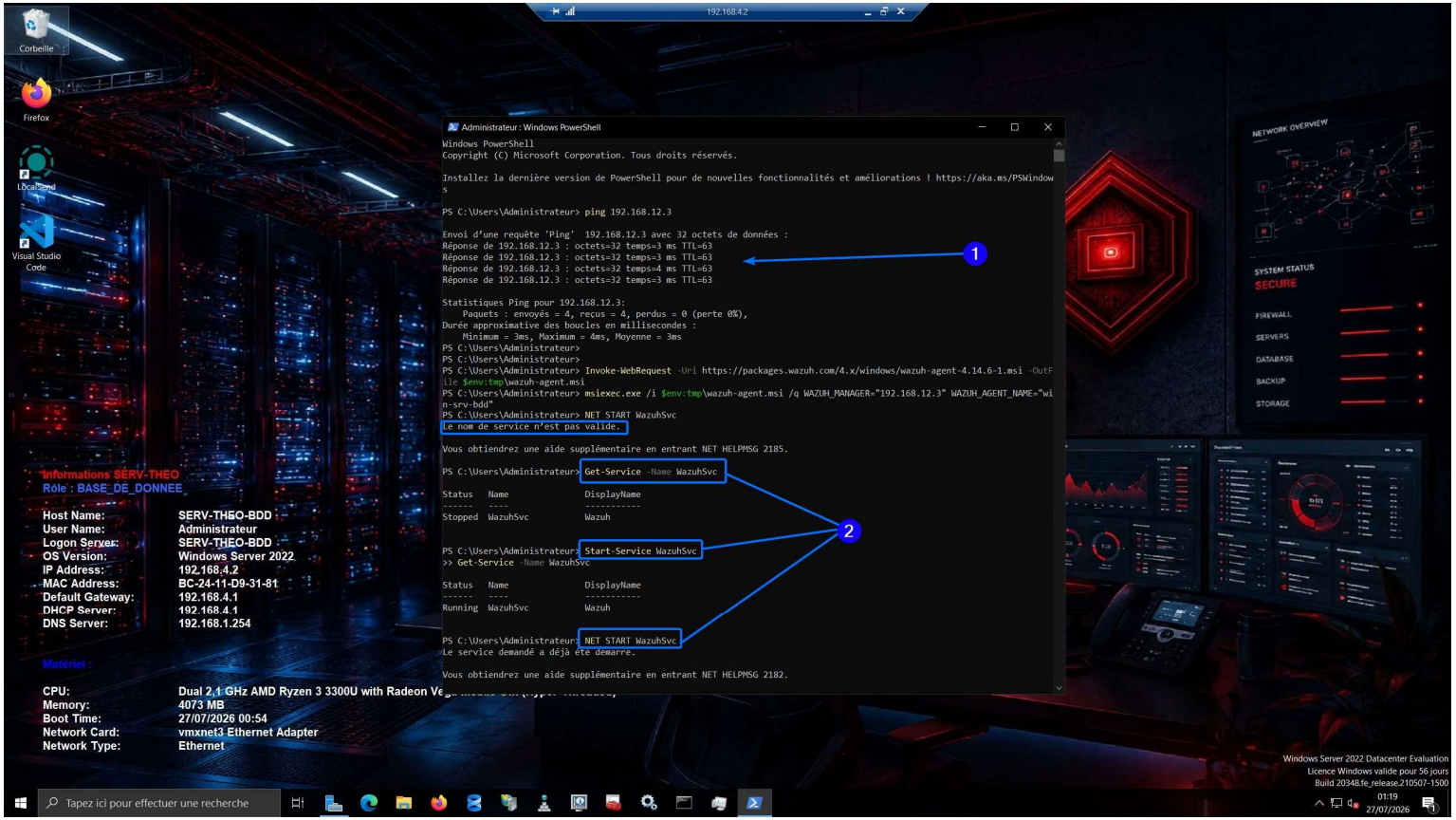


On peut voir que le serveur Web et maintenant superviser par l'agent :



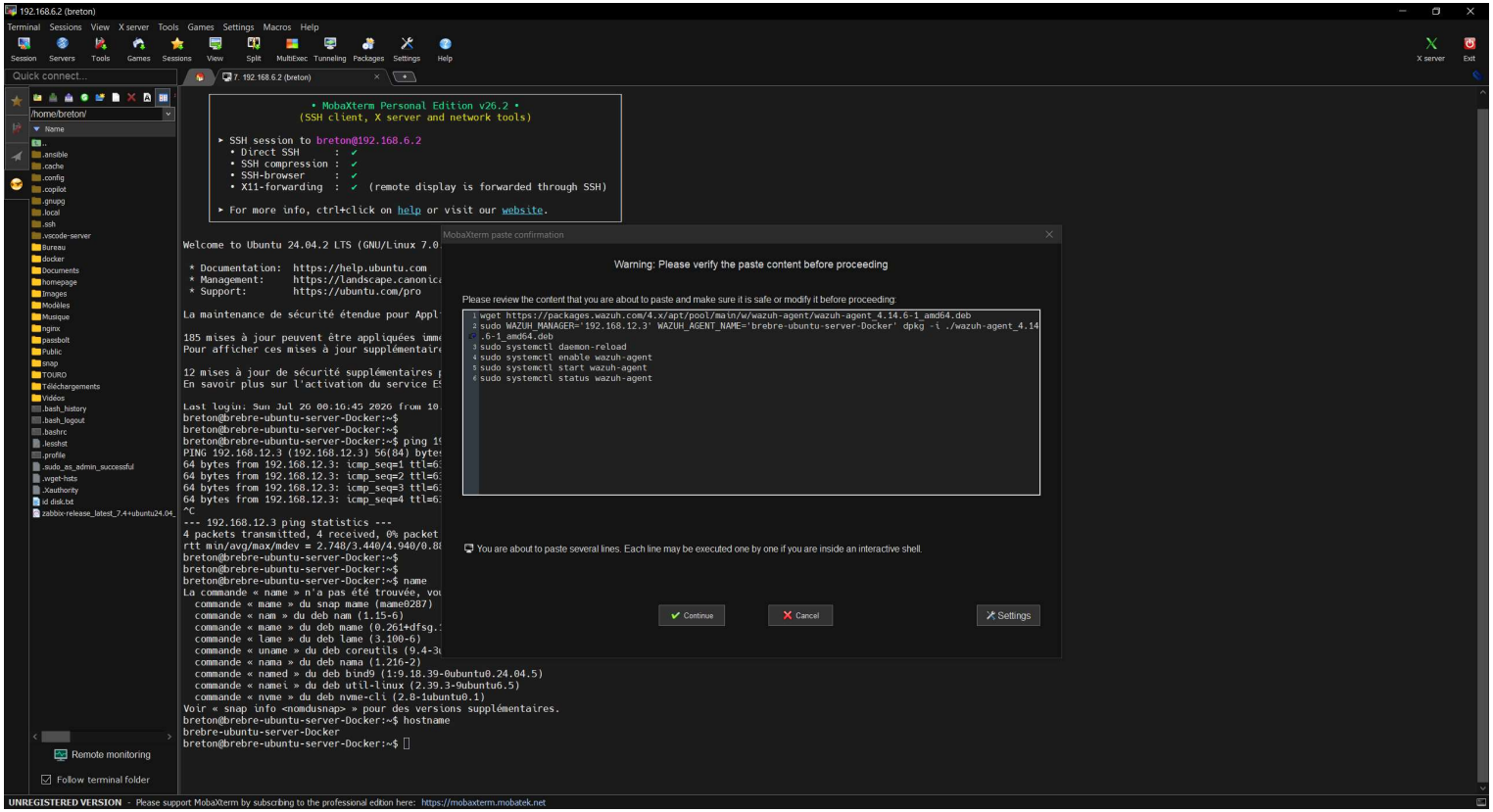


SRV-BDD :

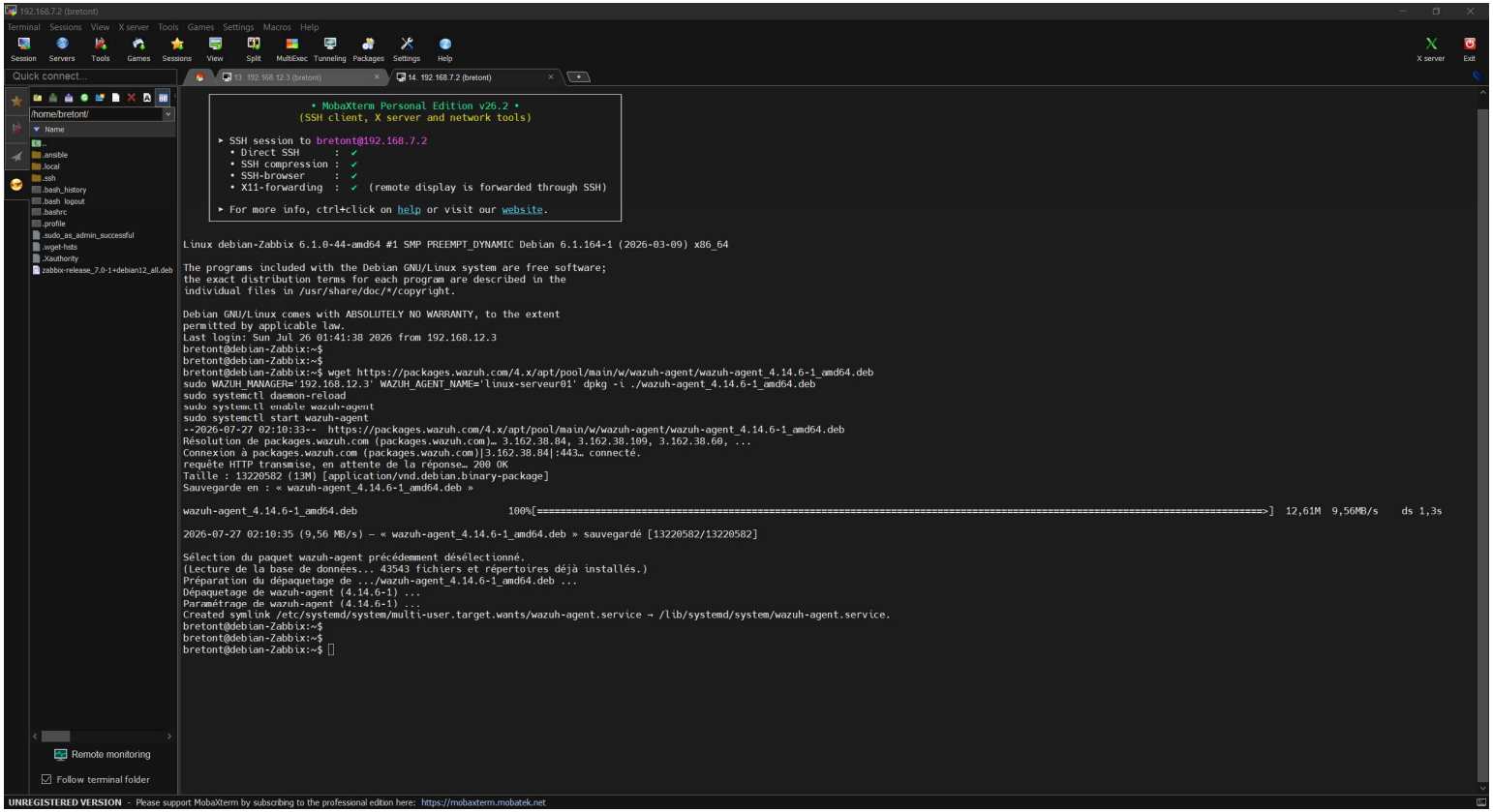

Corbeille
Firefox
Visual Studio Code
Informations SERV-THEO
Rôle : BASE_DE_DONNEE
Host Name: SERV-THEO-BDD
User Name: Administrateur
Logon Server: SERV-THEO-BDD
OS Version: Windows Server 2022
IP Address: 192.168.4.2
MAC Address: BC-24-11-D9-31-81
Default Gateway: 192.168.4.1
DHCP Server: 192.168.4.1
DNS Server: 192.168.1.254
Matériel :
CPU: Dual 2.1 GHz AMD Ryzen 3 3300U with Radeon Ve
Memory: 4073 MB
Boot Time: 27/07/2026 00:54
Network Card: vmxnet3 Ethernet Adapter
Network Type: Ethernet
Administrateur : Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. Tous droits réservés.
Installez la dernière version de PowerShell pour de nouvelles fonctionnalités et améliorations ! https://aka.ms/PSWindows
PS C:\Users\Administrateur> ping 192.168.12.3
Envoi d'une requête 'Ping' 192.168.12.3 avec 32 octets de données :
Réponse de 192.168.12.3 : octets=32 temps=3 ms TTL=63
Réponse de 192.168.12.3 : octets=32 temps=3 ms TTL=63
Réponse de 192.168.12.3 : octets=32 temps=4 ms TTL=63
Réponse de 192.168.12.3 : octets=32 temps=3 ms TTL=63
Statistiques Ping pour 192.168.12.3:
Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
Durée approximative des boucles en millisecondes :
Minimum = 3ms, Maximum = 4ms, Moyenne = 3ms
PS C:\Users\Administrateur>
PS C:\Users\Administrateur>
PS C:\Users\Administrateur> Invoke-WebRequest -Uri https://packages.wazuh.com/4.x/windows/wazuh-agent-4.14.6-1.msi -OutFile $env:tmp\wazuh-agent.msi
PS C:\Users\Administrateur> msiexec.exe /i $env:tmp\wazuh-agent.msi /q WAZUH_MANAGER="192.168.12.3" WAZUH_AGENT_NAME="win-srv-bdd"
PS C:\Users\Administrateur> NET START WazuhSvc
Le nom de service n'est pas valide.
Vous obtiendrez une aide supplémentaire en entrant NET HELPMSG 2185.
PS C:\Users\Administrateur> Get-Service -Name WazuhSvc
Status Name DisplayName
Stopped WazuhSvc Wazuh
PS C:\Users\Administrateur> Start-Service WazuhSvc
>> Get-Service -Name WazuhSvc
Status Name DisplayName
Running WazuhSvc Wazuh
PS C:\Users\Administrateur> NET START WazuhSvc
Le service demandé a déjà été démarré.
Vous obtiendrez une aide supplémentaire en entrant NET HELPMSG 2182.
1
2
NETWORK OVERVIEW
SYSTEM STATUS
SECURE
FIREWALL
SERVERS
DATABASE
BACKUP
STORAGE
Tapez ici pour effectuer une recherche
Windows Server 2022 Datacenter Evaluation
Licence Windows valide pour 56 jours
Build 20348.fe_release.210507-1500
01:19
27/07/2026

SERVEURS LINUX :

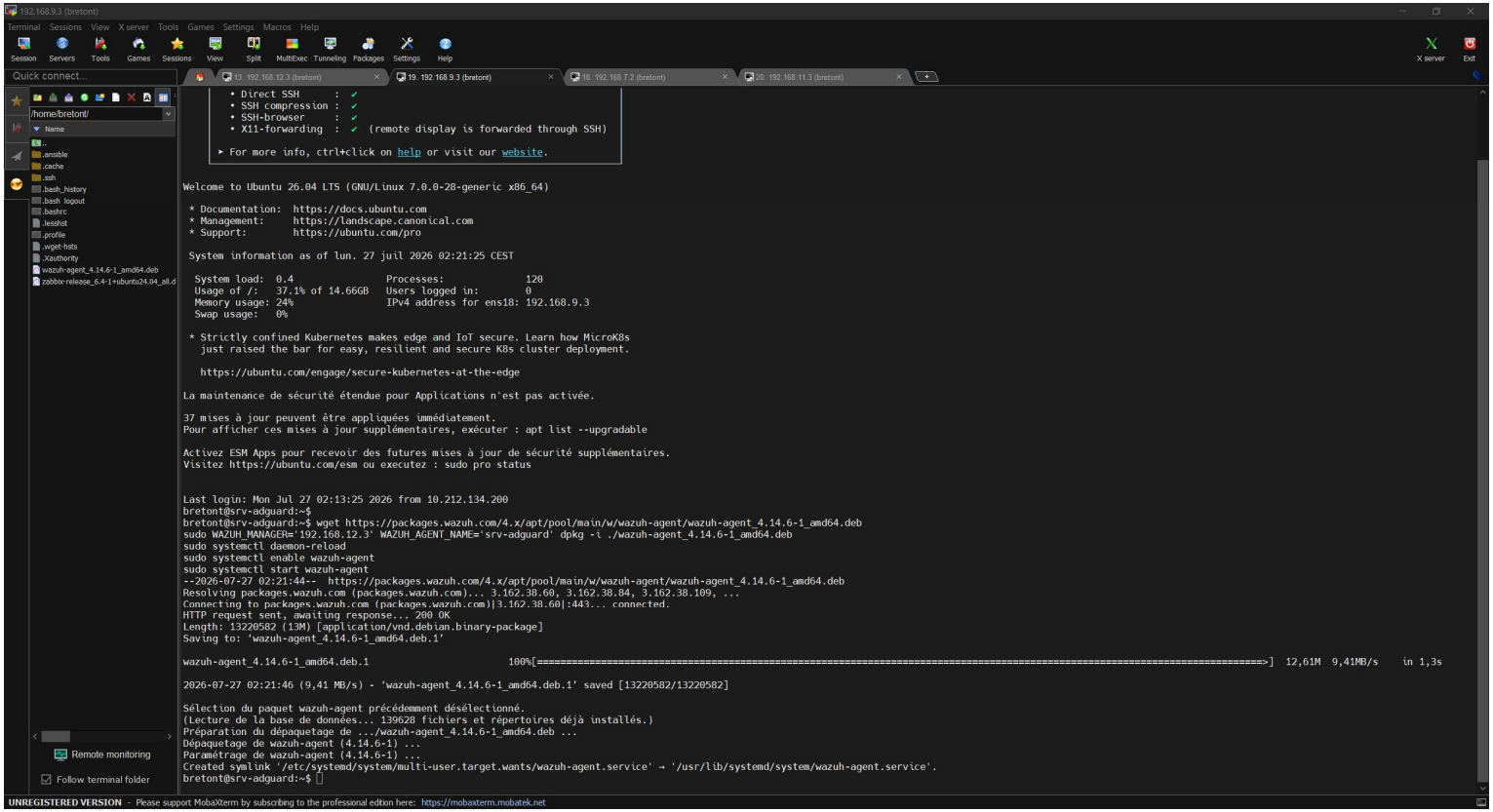
SRV-DOCKER :

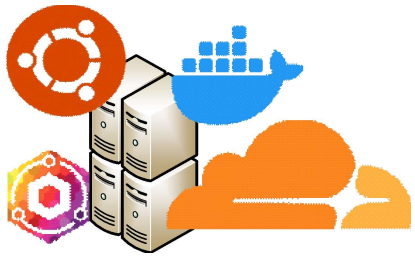


SRV-ZABBIX :

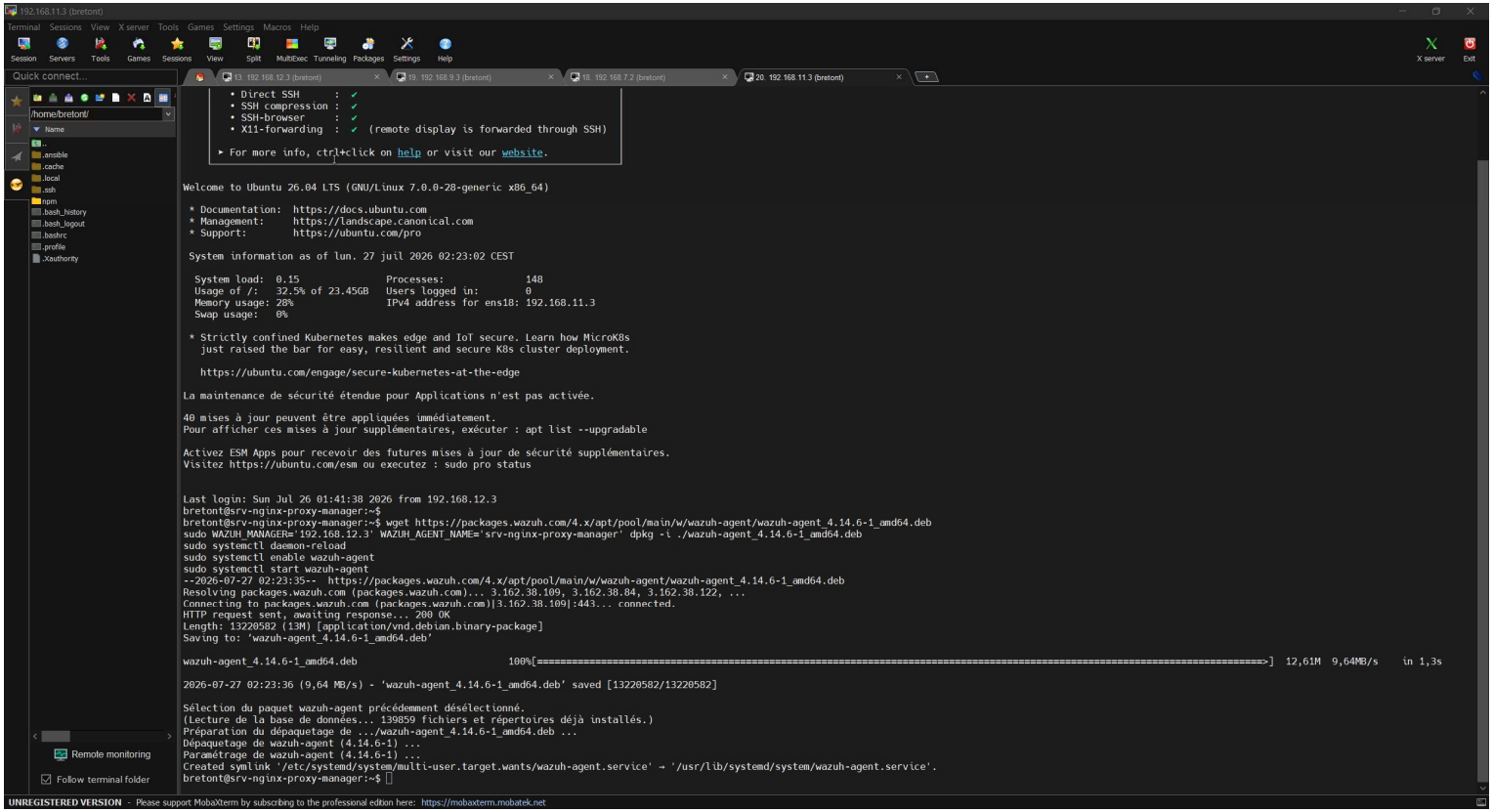

192.168.7.2 (bretont)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
X server Exit
Quick connect...
/home/bretont/
Name
.ansible
.local
.ssh
.bash_history
.bash_logout
.bashrc
.profile
.sudo_as_admin_successful
.wget-hsts
.Xauthority
zabbix-release_7.0-1+debian12_all.deb
Remote monitoring
Follow terminal folder
• MobaXterm Personal Edition v26.2 •
(SSH client, X server and network tools)
▸ SSH session to bretont@192.168.7.2
• Direct SSH : ✓
• SSH compression : ✓
• SSH-browser : ✓
• X11-forwarding : ✓ (remote display is forwarded through SSH)
▸ For more info, ctrl+click on help or visit our website.
Linux debian-Zabbix 6.1.0-44-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.164-1 (2026-03-09) x86_64
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sun Jul 26 01:41:38 2026 from 192.168.12.3
bretont@debian-Zabbix:~$
bretont@debian-Zabbix:~$
bretont@debian-Zabbix:~$ wget https://packages.wazuh.com/4.x/apt/pool/main/w/wazuh-agent/wazuh-agent_4.14.6-1_amd64.deb
sudo WAZUH_MANAGER='192.168.12.3' WAZUH_AGENT_NAME='linux-serveur01' dpkg -i ./wazuh-agent_4.14.6-1_amd64.deb
sudo systemctl daemon-reload
sudo systemctl enable wazuh-agent
sudo systemctl start wazuh-agent
--2026-07-27 02:10:33-- https://packages.wazuh.com/4.x/apt/pool/main/w/wazuh-agent/wazuh-agent_4.14.6-1_amd64.deb
Résolution de packages.wazuh.com (packages.wazuh.com)… 3.162.38.84, 3.162.38.109, 3.162.38.60, ...
Connexion à packages.wazuh.com (packages.wazuh.com)|3.162.38.84|:443… connecté.
requête HTTP transmise, en attente de la réponse… 200 OK
Taille : 13220582 (13M) [application/vnd.debian.binary-package]
Sauvegarde en : « wazuh-agent_4.14.6-1_amd64.deb »
wazuh-agent_4.14.6-1_amd64.deb 100%[=>] 12,61M 9,56MB/s ds 1,3s
2026-07-27 02:10:35 (9,56 MB/s) – « wazuh-agent_4.14.6-1_amd64.deb » sauvegardé [13220582/13220582]
Sélection du paquet wazuh-agent précédemment désélectionné.
(Lecture de la base de données... 43543 fichiers et répertoires déjà installés.)
Préparation du dépaquetage de .../wazuh-agent_4.14.6-1_amd64.deb ...
Dépaquetage de wazuh-agent (4.14.6-1) ...
Paramétrage de wazuh-agent (4.14.6-1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/wazuh-agent.service → /lib/systemd/system/wazuh-agent.service.
bretont@debian-Zabbix:~$
bretont@debian-Zabbix:~$
bretont@debian-Zabbix:~$
UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

SRV-ADGUARD :

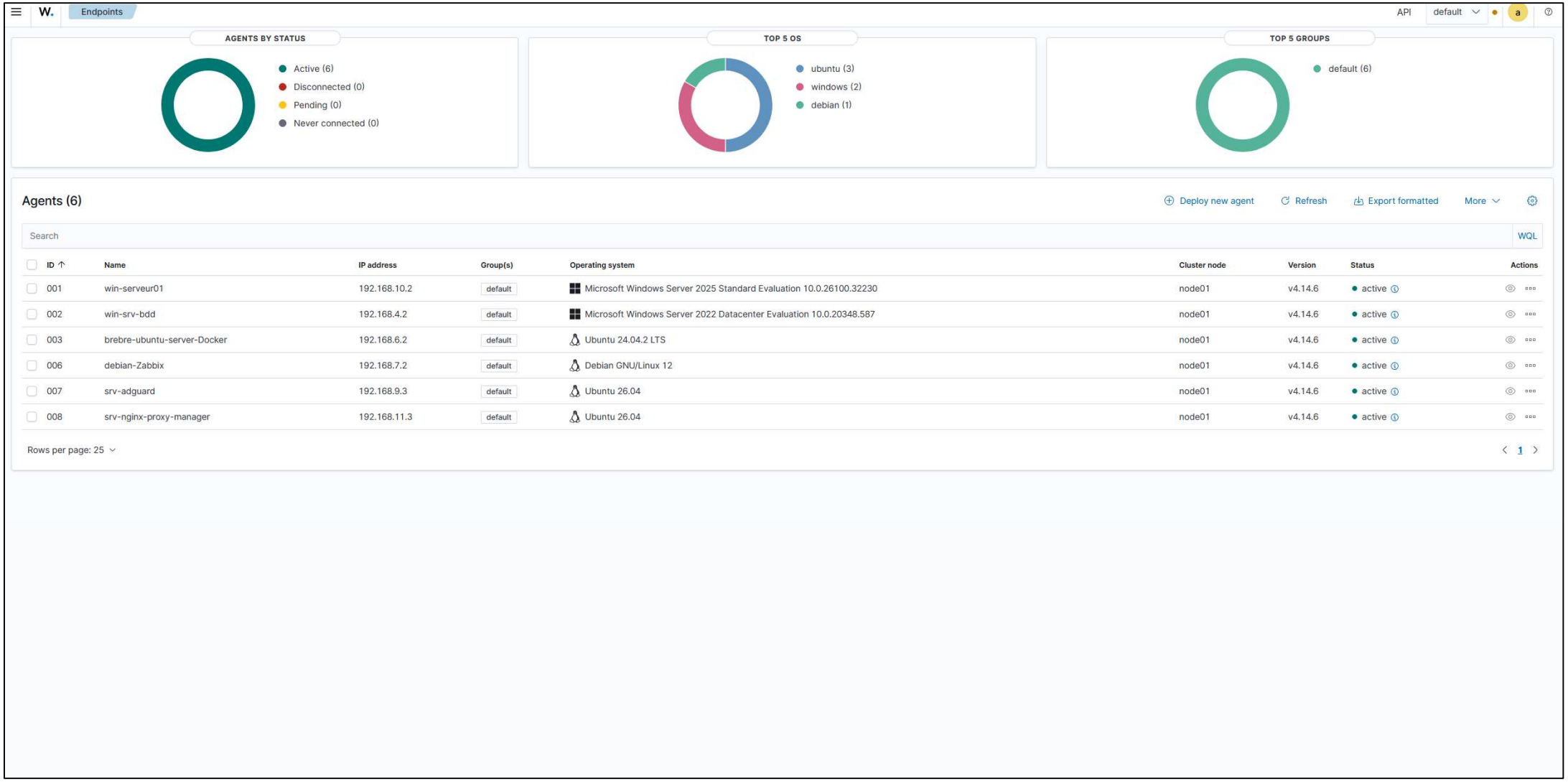

192.168.9.3 (bretont)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
X server Exit
Quick connect...
13. 192.168.12.3 (bretont)
19. 192.168.9.3 (bretont)
18. 192.168.7.2 (bretont)
20. 192.168.11.3 (bretont)
/home/bretont/
Name
.ansible
.cache
.ssh
.bash_history
.bash_logout
.bashrc
.lesshst
.profile
.wget-hsts
.Xauthority
wazuh-agent_4.14.6-1_amd64.deb
zabbix-release_6.4-1+ubuntu24.04_all.d
• Direct SSH : ✔
• SSH compression : ✔
• SSH-browser : ✔
• X11-forwarding : ✔ (remote display is forwarded through SSH)
For more info, ctrl+click on help or visit our website.
Welcome to Ubuntu 26.04 LTS (GNU/Linux 7.0.0-28-generic x86_64)
* Documentation: https://docs.ubuntu.com
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/pro
System information as of lun. 27 juil 2026 02:21:25 CEST
System load: 0.4 Processes: 120
Usage of /: 37.1% of 14.66GB Users logged in: 0
Memory usage: 24% IPv4 address for ens18: 192.168.9.3
Swap usage: 0%
* Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
just raised the bar for easy, resilient and secure K8s cluster deployment.
https://ubuntu.com/engage/secure-kubernetes-at-the-edge
La maintenance de sécurité étendue pour Applications n'est pas activée.
37 mises à jour peuvent être appliquées immédiatement.
Pour afficher ces mises à jour supplémentaires, exécuter : apt list --upgradable
Activez ESM Apps pour recevoir des futures mises à jour de sécurité supplémentaires.
Visitez https://ubuntu.com/esm ou executez : sudo pro status
Last login: Mon Jul 27 02:13:25 2026 from 10.212.134.200
bretont@srv-adguard:~$
bretont@srv-adguard:~$ wget https://packages.wazuh.com/4.x/apt/pool/main/w/wazuh-agent/wazuh-agent_4.14.6-1_amd64.deb
sudo WAZUH_MANAGER='192.168.12.3' WAZUH_AGENT_NAME='srv-adguard' dpkg -i ./wazuh-agent_4.14.6-1_amd64.deb
sudo systemctl daemon-reload
sudo systemctl enable wazuh-agent
sudo systemctl start wazuh-agent
--2026-07-27 02:21:44-- https://packages.wazuh.com/4.x/apt/pool/main/w/wazuh-agent/wazuh-agent_4.14.6-1_amd64.deb
Resolving packages.wazuh.com (packages.wazuh.com)... 3.162.38.60, 3.162.38.84, 3.162.38.109, ...
Connecting to packages.wazuh.com (packages.wazuh.com)|3.162.38.60|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 13220582 (13M) [application/vnd.debian.binary-package]
Saving to: 'wazuh-agent_4.14.6-1_amd64.deb.1'
wazuh-agent_4.14.6-1_amd64.deb.1 100%[==========>] 12,61M 9,41MB/s in 1,3s
2026-07-27 02:21:46 (9,41 MB/s) - 'wazuh-agent_4.14.6-1_amd64.deb.1' saved [13220582/13220582]
Sélection du paquet wazuh-agent précédemment désélectionné.
(Lecture de la base de données... 139628 fichiers et répertoires déjà installés.)
Préparation du dépaquetage de .../wazuh-agent_4.14.6-1_amd64.deb ...
Dépaquetage de wazuh-agent (4.14.6-1) ...
Paramétrage de wazuh-agent (4.14.6-1) ...
Created symlink '/etc/systemd/system/multi-user.target.wants/wazuh-agent.service' → '/usr/lib/systemd/system/wazuh-agent.service'.
bretont@srv-adguard:~$
Remote monitoring
Follow terminal folder
UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

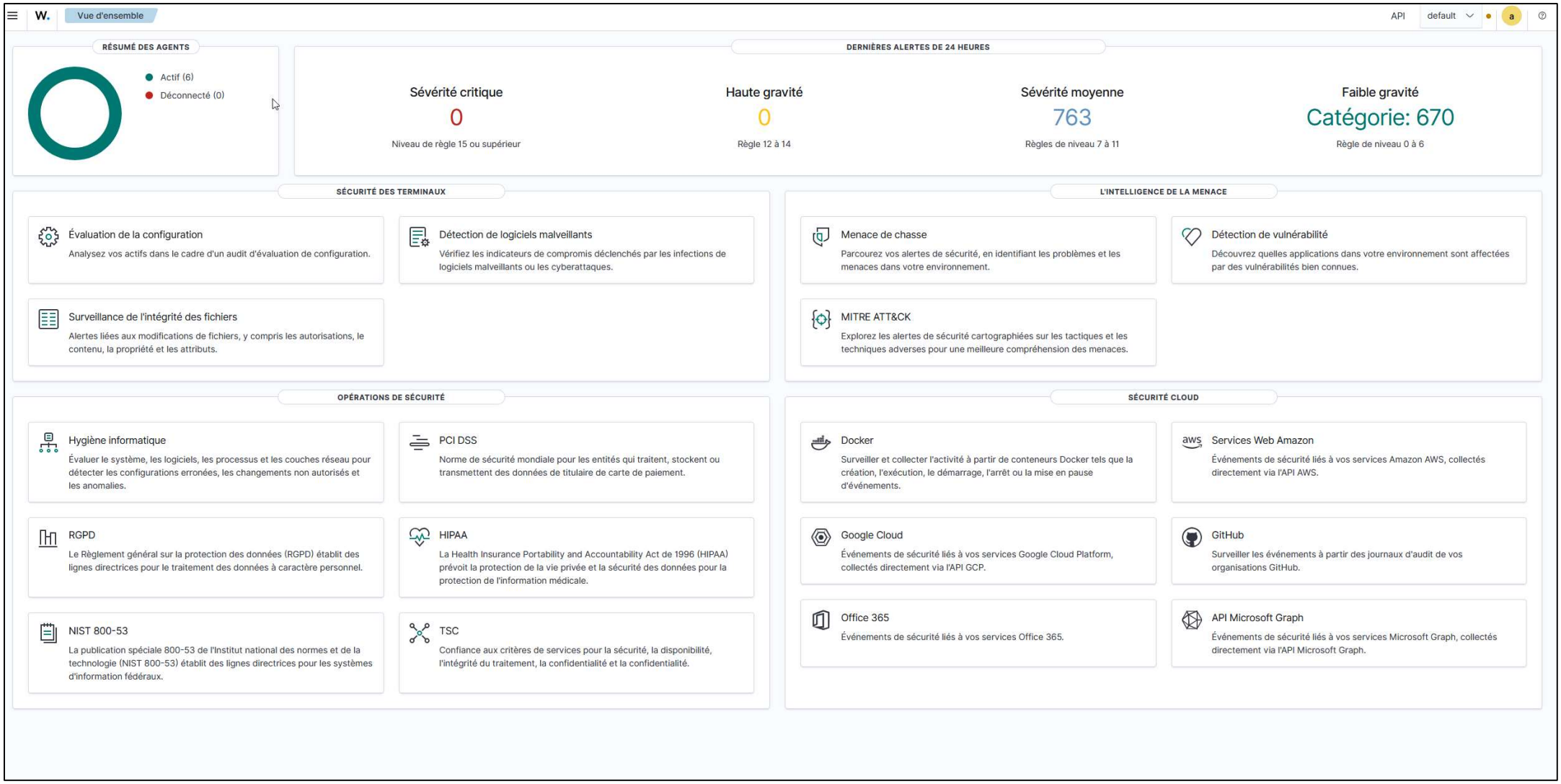


SRV-NGINX-PROXY-MANAGER :



L’ensemble des serveurs sont mis en place sur Wazuh :




W.
Vue d'ensemble
API
default
a
RÉSUMÉ DES AGENTS
Actif (6)
Déconnecté (0)
DERNIÈRES ALERTES DE 24 HEURES
Sévérité critique
0
Niveau de règle 15 ou supérieur
Haute gravité
0
Règle 12 à 14
Sévérité moyenne
763
Règles de niveau 7 à 11
Faible gravité
Catégorie: 670
Règle de niveau 0 à 6
SÉCURITÉ DES TERMINAUX
Évaluation de la configuration
Analysez vos actifs dans le cadre d'un audit d'évaluation de configuration.
Détection de logiciels malveillants
Vérifiez les indicateurs de compromis déclenchés par les infections de logiciels malveillants ou les cyberattaques.
Surveillance de l'intégrité des fichiers
Alertes liées aux modifications de fichiers, y compris les autorisations, le contenu, la propriété et les attributs.
L'INTELLIGENCE DE LA MENACE
Menace de chasse
Parcourez vos alertes de sécurité, en identifiant les problèmes et les menaces dans votre environnement.
Détection de vulnérabilité
Découvrez quelles applications dans votre environnement sont affectées par des vulnérabilités bien connues.
MITRE ATT&CK
Explorez les alertes de sécurité cartographiées sur les tactiques et les techniques adverses pour une meilleure compréhension des menaces.
OPÉRATIONS DE SÉCURITÉ
Hygiène informatique
Évaluer le système, les logiciels, les processus et les couches réseau pour détecter les configurations erronées, les changements non autorisés et les anomalies.
PCI DSS
Norme de sécurité mondiale pour les entités qui traitent, stockent ou transmettent des données de titulaire de carte de paiement.
RGPD
Le Règlement général sur la protection des données (RGPD) établit des lignes directrices pour le traitement des données à caractère personnel.
HIPAA
La Health Insurance Portability and Accountability Act de 1996 (HIPAA) prévoit la protection de la vie privée et la sécurité des données pour la protection de l'information médicale.
NIST 800-53
La publication spéciale 800-53 de l'Institut national des normes et de la technologie (NIST 800-53) établit des lignes directrices pour les systèmes d'information fédéraux.
TSC
Confiance aux critères de services pour la sécurité, la disponibilité, l'intégrité du traitement, la confidentialité et la confidentialité.
SÉCURITÉ CLOUD
Docker
Surveiller et collecter l'activité à partir de conteneurs Docker tels que la création, l'exécution, le démarrage, l'arrêt ou la mise en pause d'événements.
Services Web Amazon
Événements de sécurité liés à vos services Amazon AWS, collectés directement via l'API AWS.
Google Cloud
Événements de sécurité liés à vos services Google Cloud Platform, collectés directement via l'API GCP.
GitHub
Surveiller les événements à partir des journaux d'audit de vos organisations GitHub.
Office 365
Événements de sécurité liés à vos services Office 365.
API Microsoft Graph
Événements de sécurité liés à vos services Microsoft Graph, collectés directement via l'API Microsoft Graph.