

Déploiement et Administration d'un Réseau
d'Entreprise Virtualisé avec Renforcement de la
Sécurité



Table des matières

Déploiement et Administration d'un Réseau d'Entreprise Virtualisé avec Renforcement de la Sécurité	1
I. Contexte professionnel détaillé	4
II. Étude des besoins, objectifs et contraintes	5
1. Besoins identifiés	5
2. Objectifs techniques du projet	5
3. Contraintes techniques et pédagogiques	6
III. Architecture réseau cible	7
1. Réseaux virtuels définis dans Proxmox	7
2. Organisation logique du réseau interne	7
3. Cheminement des flux	7
IV. Plan d'adressage IP et rôle des machines	8
VI. Installation et configuration de pfSense	9
1. Création de la machine virtuelle pfSense	9
2. Installation du système pfSense	9
3. Configuration initiale post-installation	9
4. Accès à l'interface web	10
5. Configuration de base dans l'interface Web	10
6. Redirections NAT	10
7. Règles de pare-feu (Firewall > Rules)	10
8. Conclusion	10
VII. Installation de la VM Debian 12 (avec interface graphique)	11
Création de la VM dans Proxmox	11
Lancement de l'installation	11
Vérifications post-installation	11
VIII. Configuration réseau statique sur Debian	12

Étapes de configuration (interface graphique GNOME)	12
IX. Installation de Docker et vérification de l'environnement	12
1. Mise à jour du système et installation des dépendances	13
2. Ajout du dépôt officiel Docker	13
3. Installation de Docker Engine et des outils associés	13
4. Vérification de l'installation	13
5. Ajout de l'utilisateur au groupe Docker	13
Conclusion.....	14
X. Déploiement des services Dockerisés (ELK, Netdata, Suricata, Zabbix)	14
1. Déploiement de la stack ELK.....	15
2. Déploiement de Netdata, Suricata et Zabbix	16
XI. Installation des agents (Zabbix, Filebeat, Winlogbeat)	17
XII. Installation de Windows Server 2022	18
XIII. Promotion en contrôleur de domaine et installation du rôle DNS.....	20
XIV. Création des OU, utilisateurs, groupes et affectations.....	21
XV. Création, configuration et application des GPO	22
XVI. Intégration des postes clients au domaine	23
XVII. Déploiement de Security Onion et intégration au réseau.....	24
XVIII. Tableaux récapitulatifs (IPs, GPO, utilisateurs, services)	25
1. Plan d'adressage IP.....	25
2. GPO appliquées par OU	25
3. Services Docker déployés	25
XIX. Conclusion du projet	27

I. Contexte professionnel détaillé

L'entreprise fictive **LogiWare Solutions** est une PME spécialisée dans le développement de logiciels de gestion pour les TPE et PME du secteur logistique. Forte d'une croissance rapide, l'entreprise a récemment étendu son effectif et diversifié ses services, ce qui l'a conduite à repenser complètement son infrastructure informatique.

Jusqu'alors hébergée sur une architecture simple avec des postes en groupe de travail et un partage de fichiers de base, l'entreprise a pris conscience des risques liés à la cybersécurité, à la centralisation des données et à la maintenance inefficace de son réseau. C'est dans ce contexte que le responsable informatique a été chargé de concevoir et mettre en œuvre une **infrastructure réseau virtualisée, sécurisée, centralisée et évolutive**.

Ce projet vise donc à transformer l'environnement informatique de LogiWare Solutions en une plateforme robuste et professionnelle. Il s'appuie sur la **virtualisation avec Proxmox**, l'usage de **services conteneurisés avec Docker**, et l'intégration de plusieurs outils de supervision, sécurité, et gestion centralisée des utilisateurs.

Parmi les objectifs concrets du projet, on retrouve :

- La mise en place d'un **pare-feu centralisé (pfSense)** pour contrôler les flux réseau entrants et sortants.
- Le déploiement d'un **contrôleur de domaine Windows Server 2022** avec Active Directory pour la gestion des utilisateurs, des postes et des stratégies de sécurité.
- L'installation de **machines Linux Debian** pour accueillir les services critiques via Docker (ELK, Netdata, Zabbix, Suricata).
- L'intégration d'une solution avancée de **détection d'intrusions avec Security Onion** pour surveiller l'ensemble du réseau.

Le tout est déployé dans un environnement **virtualisé sur Proxmox**, garantissant une souplesse de gestion, une meilleure sécurité, et une isolation efficace entre les services. Cette solution permet également une meilleure évolutivité pour faire face à l'évolution des besoins de l'entreprise.

Ce projet est conçu comme un socle durable, sécurisé et évolutif sur lequel l'entreprise pourra s'appuyer pour poursuivre son développement numérique dans un cadre sécurisé et structuré.

II. Étude des besoins, objectifs et contraintes

Afin de garantir la réussite du projet, une analyse approfondie des besoins fonctionnels, des objectifs techniques et des contraintes spécifiques a été réalisée en amont. Cette étude permet de cadrer précisément la solution à mettre en œuvre et de répondre de manière cohérente aux attentes de l'entreprise LogiWare Solutions.

1. Besoins identifiés

- **Centralisation des utilisateurs et des postes** : L'entreprise souhaite abandonner le fonctionnement en groupe de travail et passer à une gestion centralisée des comptes utilisateurs, des postes clients et des permissions via Active Directory.
- **Renforcement de la sécurité réseau** : Il est impératif de sécuriser l'accès aux ressources internes, de filtrer les connexions entrantes et sortantes, et de détecter rapidement tout comportement suspect ou attaque réseau.
- **Supervision du système et collecte des logs** : Les administrateurs doivent être en mesure de surveiller en temps réel l'état des machines et des services, et de centraliser les journaux d'événements pour effectuer des analyses ou audits.
- **Infrastructure évolutive et maintenable** : Le système mis en place doit permettre de déployer de nouveaux services facilement, de restaurer rapidement en cas de panne, et de s'adapter aux évolutions futures de l'entreprise.

2. Objectifs techniques du projet

- Déployer une **infrastructure entièrement virtualisée** avec Proxmox pour permettre la gestion flexible des machines.
- Installer un **pare-feu pfSense** assurant le contrôle, la segmentation et la sécurité du réseau.
- Mettre en place une **machine Debian 12** avec Docker pour héberger plusieurs services essentiels (ELK, Netdata, Zabbix, Suricata).
- Intégrer une **machine Windows Server 2022** configurée en tant que **contrôleur de domaine Active Directory** avec gestion DNS.
- Créer une **machine client Windows 10** afin de tester la gestion des utilisateurs, GPO et connexions au domaine.
- Ajouter une **machine Security Onion** pour la **détection d'intrusions réseau avancée** et l'analyse forensique.

3. Contraintes techniques et pédagogiques

- **Ressources limitées** : Le projet doit être déployé sur une infrastructure de test limitée (machines virtuelles dans Proxmox), ce qui implique une optimisation des ressources (CPU, RAM, stockage).
- **Environnement fermé** : Aucun accès Internet direct autorisé sur les postes clients, tous les flux doivent passer par pfSense pour surveillance et filtrage.
- **Respect des bonnes pratiques** : Toutes les configurations doivent être documentées, sécurisées (mots de passe forts, pare-feu actif), et respecter les standards professionnels.
- **Contexte BTS SIO** : Ce projet s'inscrit dans le cadre d'une épreuve pratique de BTS SIO, il doit donc démontrer des compétences techniques concrètes, des capacités de diagnostic, d'automatisation et de documentation approfondie.

Cette étude permet de cadrer précisément la solution retenue, en veillant à ce qu'elle soit techniquement réalisable, pédagogique, mais surtout réaliste dans un environnement professionnel simulé.

III. Architecture réseau cible

L'architecture réseau du projet repose sur une organisation claire et cloisonnée au sein d'un environnement virtualisé sous Proxmox. Elle s'appuie sur l'utilisation de deux ponts réseau virtuels (vmbr), configurés de manière à séparer le réseau interne des VMs de l'accès à Internet. pfSense joue le rôle de pare-feu et de routeur central entre ces deux mondes.

1. Réseaux virtuels définis dans Proxmox

Nom du bridge	Rôle principal	Accès Internet	Utilisé par
vmbr0	Réseau connecté à Internet (WAN)	Oui	pfSense (interface WAN)
vmbr1	Réseau interne isolé (LAN virtuel)	Non direct	Toutes les autres VMs

Ce découpage permet de simuler un réseau d'entreprise sécurisé :

- Seul pfSense est connecté à Internet via vmbr0.
- Toutes les autres machines communiquent entre elles via vmbr1, en passant par pfSense pour sortir ou être filtrées.

2. Organisation logique du réseau interne

Le réseau LAN virtuel (via vmbr1) est configuré en 192.168.1.0/24, avec pfSense en passerelle (192.168.1.1). Chaque machine a reçu une adresse IP statique prédéfinie dans le plan d'adressage, ce qui permet un contrôle précis et un routage simple. Toutes les machines internes (serveurs comme clients) sont donc dans le même sous-réseau, ce qui facilite la communication directe entre les conteneurs, les postes clients et les services d'infrastructure (AD, logs, supervision, etc.).

3. Cheminement des flux

- Le trafic sortant des machines (vers Internet) transite par pfSense (NAT sur vmbr0).
- Le trafic entrant (accès aux services Docker comme Kibana, Zabbix, Netdata) est redirigé via des règles NAT configurées sur pfSense.
- Les flux internes entre VMs (ex : Filebeat vers Logstash, Winlogbeat vers ELK, agent Zabbix vers serveur) passent uniquement par le réseau LAN (vmbr1).

Cette architecture offre un maximum de sécurité, d'isolement et de contrôle des flux, tout en permettant l'intégration fluide des services nécessaires au bon fonctionnement de l'infrastructure.

IV. Plan d'adressage IP et rôle des machines

Le plan d'adressage IP a été conçu pour assurer une cohérence, une lisibilité et une évolutivité dans l'architecture réseau. Chaque machine virtuelle a reçu une adresse IP statique appartenant au sous-réseau 192.168.1.0/24, en fonction de son rôle dans l'infrastructure. La passerelle utilisée par toutes les machines est l'adresse LAN de pfSense (192.168.1.1).

Tableau récapitulatif des adresses IP

Nom de la machine	Rôle principal	Adresse IP	Système d'exploitation
pfSense (LAN)	Pare-feu, DNS, NAT, proxy, passerelle réseau	192.168.1.1	pfSense
Debian Docker	Hôte de tous les services Dockerisés	192.168.1.12	Debian 12 (GNOME)
Windows Server 2022	Contrôleur de domaine, DNS, GPO	192.168.1.10	Windows Server 2022
Windows Client	Poste de test utilisateur intégré au domaine	192.168.1.20	Windows 10
Security Onion	Détection d'intrusions réseau (IDS/NSM complet)	192.168.1.30	Security Onion

Rôle des machines dans le projet

- **pfSense** assure le routage, le filtrage, la translation d'adresses (NAT) et la sécurisation des accès réseau. Il joue également un rôle de proxy web et de serveur DNS interne.
- **Debian Docker** est le serveur central de l'infrastructure. Il héberge plusieurs services critiques via Docker : la stack ELK (Elasticsearch, Logstash, Kibana), Netdata pour la supervision en temps réel, Zabbix pour la supervision détaillée, et Suricata en tant qu'IDS Dockerisé.
- **Windows Server 2022** joue un rôle fondamental en centralisant la gestion des comptes utilisateurs et des ordinateurs. Il permet la création et la gestion des OU, des GPO, et le déploiement d'un environnement Active Directory sécurisé.
- **Windows Client** est utilisé pour tester l'intégration au domaine, l'application des stratégies de groupe, l'accès aux ressources réseau, ainsi que l'envoi de logs via Winlogbeat ou Filebeat vers ELK.
- **Security Onion** complète Suricata en fournissant une surveillance réseau avancée, des interfaces d'analyse graphique, et des capacités de corrélation d'événements à grande échelle.

VI. Installation et configuration de pfSense

Dans cette partie, nous détaillons l'installation et la configuration de pfSense, utilisé comme pare-feu principal de l'infrastructure. Cette machine virtuelle joue également le rôle de routeur, de serveur DNS, et de proxy filtrant.

1. Création de la machine virtuelle pfSense

La VM pfSense a été créée depuis Proxmox avec les paramètres suivants :

- 1 vCPU
- 1 Go de RAM
- 8 Go de disque virtuel
- Deux interfaces réseau :
 - **vmbr0** : pour la connexion au réseau Internet (WAN)
 - **vmbr1** : pour le réseau interne de l'entreprise (LAN)

L'ISO pfSense-CE-2.7.0.iso a été utilisé pour l'installation.

2. Installation du système pfSense

L'installation suit le processus standard :

- Démarrage depuis l'ISO dans la VM.
- Choix de l'option "Install pfSense".
- Acceptation de la licence.
- Partitionnement automatique du disque.
- Sélection du clavier français (FR ISO).
- Installation du système de base.
- Définition du mot de passe administrateur.
- Redémarrage de la VM une fois l'installation terminée.

3. Configuration initiale post-installation

À l'issue du redémarrage, pfSense détecte les interfaces réseau :

- **em0** → WAN (vmbr0)
- **em1** → LAN (vmbr1)

L'adressage réseau a été configuré comme suit :

- WAN : IP obtenue via DHCP (attribuée par Proxmox en 10.4.0.253)
- LAN : IP statique 192.168.1.1/24 (passerelle du réseau interne)

4. Accès à l'interface web

Depuis une machine du réseau interne (par exemple, Debian ou Windows Client), l'interface d'administration de pfSense est accessible via :

<https://192.168.1.1>

Le mot de passe défini à l'installation est utilisé pour se connecter au tableau de bord web.

5. Configuration de base dans l'interface Web

Une fois connecté :

- Changement du mot de passe administrateur.
- Désactivation du DHCP sur le LAN (IP statiques uniquement dans notre projet).
- Vérification de la connectivité WAN (accès Internet).
- Activation du **DNS Resolver** pour la résolution interne.

6. Redirections NAT

Pour rendre les services Docker accessibles depuis d'autres machines, des redirections NAT ont été créées dans **Firewall > NAT > Port Forward** :

- Port 5601 → VM Debian (Kibana)
- Port 8080 → VM Debian (Zabbix frontend)
- Port 19999 → VM Debian (Netdata)

Chaque règle a été suivie d'une règle firewall automatique autorisant le trafic entrant sur ces ports.

7. Règles de pare-feu (Firewall > Rules)

- **LAN vers WAN** : tout trafic autorisé (accès Internet).
- **WAN vers LAN** : tout est bloqué, sauf les ports redirigés.
- **Règles internes personnalisées** : ajout possible pour bloquer ou surveiller certains flux spécifiques.

8. Conclusion

À ce stade du projet, pfSense remplit pleinement ses fonctions dans l'architecture. Il agit comme routeur sécurisé entre le réseau interne et l'extérieur, assurant une translation d'adresses efficace via le NAT. Son pare-feu permet de filtrer finement les flux entrants et sortants selon des règles personnalisées adaptées à notre infrastructure. Il permet également d'implémenter des redirections de ports précises pour rendre accessibles les services hébergés dans Docker tout en contrôlant leur exposition. Enfin, il garantit un accès conditionné à Internet depuis le réseau interne, contribuant ainsi activement à la sécurité globale du système.

VII. Installation de la VM Debian 12 (avec interface graphique)

La machine Debian 12 servira de socle à l'ensemble des services Dockerisés déployés dans ce projet. Il est donc essentiel qu'elle soit stable, proprement configurée, et accessible sur le réseau interne.

Création de la VM dans Proxmox

Depuis l'interface web de Proxmox, une nouvelle machine virtuelle a été créée avec les paramètres suivants :

- Nom : debian-docker
- ISO utilisé : debian-12.5.0-amd64-netinst.iso
- 2 vCPU
- 4 Go de RAM
- 20 Go de disque virtuel (stockage local)
- Interface réseau : vmbr1 (réseau interne, LAN)

Lancement de l'installation

Au premier démarrage, la VM a été lancée sur l'image ISO, et le programme d'installation a été suivi étape par étape :

1. **Choix de la langue** : Français
2. **Disposition du clavier** : Français (AZERTY)
3. **Configuration du nom de la machine** : debian-docker
4. **Nom de domaine** : laissé vide car pas encore intégré au domaine AD
5. **Définition d'un mot de passe root fort**
6. **Création d'un utilisateur standard** : admin-docker
7. **Partitionnement** : Utilisation de tout le disque avec LVM
8. **Sélection des dépôts de paquets** : Miroir français, sans proxy
9. **Installation de l'environnement de bureau** : GNOME
10. **Installation du chargeur GRUB sur /dev/sda**

Une fois le système installé, la machine a été redémarrée.

Vérifications post-installation

Après le premier démarrage :

- Connexion à l'environnement GNOME avec l'utilisateur admin-docker
- Mise à jour du système avec les commandes suivantes :

`sudo apt update && sudo apt upgrade -y`

- Vérification de la reconnaissance de l'interface réseau connectée à vmbr1


La VM Debian est maintenant prête pour l'étape suivante : la configuration d'une adresse IP statique, nécessaire à la stabilité de tous les services à venir.

VIII. Configuration réseau statique sur Debian

Pour garantir un fonctionnement stable des services déployés en Docker, la machine Debian doit utiliser une adresse IP fixe. Cette configuration est nécessaire pour permettre les redirections de ports via pfSense, l'accès aux services Docker depuis les postes du réseau, et l'intégration fluide avec d'autres outils comme Zabbix ou ELK.

La configuration IP statique a été réalisée via l'interface graphique GNOME, en modifiant les paramètres réseau de la connexion filaire associée à l'interface connectée à vmbr1.

Étapes de configuration (interface graphique GNOME)

1. Cliquer sur l'icône réseau en haut à droite de l'écran.
2. Accéder aux **Paramètres Réseau**.
3. Cliquer sur l'icône  à côté de la connexion Ethernet détectée.
4. Aller dans l'onglet **IPv4**.
5. Changer le mode de configuration de **Automatique (DHCP)** à **Manuel**.
6. Entrer les paramètres suivants :
 - **Adresse IP** : 192.168.1.12
 - **Masque** : 255.255.255.0 (ou préfixe /24)
 - **Passerelle** : 192.168.1.1 (pfSense)
 - **DNS** : 1.1.1.1 et 192.168.1.1
7. Enregistrer et redémarrer la connexion (ou la machine).

Vérifications

Après redémarrage de l'interface réseau :

- Exécution de la commande `ip a` pour vérifier que l'interface `enp0s8` (ou équivalent) est bien configurée en 192.168.1.12.
- Test de connectivité avec la commande `ping 192.168.1.1` (vérifie l'accès à pfSense).
- Vérification de la résolution DNS avec `ping google.com`.

Une fois ces étapes réalisées, la machine Debian est prête à accueillir l'installation de Docker et à communiquer avec le reste du réseau de manière fiable.

IX. Installation de Docker et vérification de l'environnement

L'installation de Docker sur la machine Debian est une étape essentielle du projet, car elle permet de déployer les différents services sous forme de conteneurs. Docker apporte une

portabilité, une isolation et une simplicité de gestion très appréciées en environnement de production comme dans ce projet pédagogique.

1. Mise à jour du système et installation des dépendances

Avant d'installer Docker, il est important de mettre à jour entièrement le système et d'installer les paquets nécessaires à la gestion sécurisée des dépôts :

```
sudo apt update && sudo apt upgrade -y
```

```
sudo apt install -y ca-certificates curl gnupg lsb-release
```

2. Ajout du dépôt officiel Docker

On ajoute la clé GPG du dépôt Docker, puis on configure le dépôt stable pour Debian 12 (Bookworm) :

```
- sudo install -m 0755 -d /etc/apt/keyrings
```

```
- curl -fsSL https://download.docker.com/linux/debian/gpg | sudo gpg --dearmor  
-o /etc/apt/keyrings/docker.gpg
```

```
echo \
```

```
"deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg]  
https://download.docker.com/linux/debian \
```

```
$(. /etc/os-release && echo "$VERSION_CODENAME") stable" | \
```

```
sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

3. Installation de Docker Engine et des outils associés

```
sudo apt update
```

```
sudo apt install -y docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-  
compose-plugin
```

Une fois terminé, Docker est installé et activé automatiquement.

4. Vérification de l'installation

Pour vérifier que Docker fonctionne correctement, on peut lancer le conteneur de test officiel :

```
docker run hello-world
```

Si le message de bienvenue apparaît, cela signifie que Docker est installé et opérationnel.

5. Ajout de l'utilisateur au groupe Docker

Pour pouvoir exécuter Docker sans sudo, on peut ajouter l'utilisateur courant au groupe Docker :

```
sudo usermod -aG docker $USER
```

Un redémarrage de la session est nécessaire pour appliquer ce changement.

Conclusion

À l'issue de cette étape, la machine Debian est désormais prête à héberger les services conteneurisés prévus dans le projet (ELK, Netdata, Suricata, Zabbix). Docker offre un environnement isolé, modulaire et facilement reproductible, facilitant la gestion et la maintenance des différents outils de supervision et de sécurité.

X. Déploiement des services Dockerisés (ELK, Netdata, Suricata, Zabbix)

Le déploiement des services Dockerisés s'est fait de manière progressive, en commençant par les composants de la stack ELK. Cette stack (Elasticsearch, Logstash, Kibana) permet de centraliser et visualiser les logs provenant des différentes machines du réseau. Elle est essentielle dans notre infrastructure pour la supervision et l'analyse des événements systèmes et de sécurité.

1. Déploiement de la stack ELK

a) Présentation des composants

- **Elasticsearch** : moteur d'indexation et de recherche, il stocke les logs reçus.
- **Logstash** : pipeline de traitement des données, il reçoit les logs (depuis Filebeat ou Winlogbeat), les filtre, les structure, puis les transmet à Elasticsearch.
- **Kibana** : interface web de visualisation des données contenues dans Elasticsearch.

b) Création du fichier docker-compose.yml

Un fichier `docker-compose.yml` a été créé pour définir et lancer les trois services :

```
version: '3.7'
services:
  elasticsearch:
    image: docker.elastic.co/elasticsearch/elasticsearch:7.17.9
    container_name: elasticsearch
    environment:
      - discovery.type=single-node
      - xpack.security.enabled=false
    ports:
      - "9200:9200"
    volumes:
      - esdata:/usr/share/elasticsearch/data

  logstash:
    image: docker.elastic.co/logstash/logstash:7.17.9
    container_name: logstash
    volumes:
      - ./logstash.conf:/usr/share/logstash/pipeline/logstash.conf
    ports:
      - "5044:5044"

  kibana:
    image: docker.elastic.co/kibana/kibana:7.17.9
    container_name: kibana
    ports:
      - "5601:5601"
    depends_on:
      - elasticsearch

volumes:
  esdata:
```

Ce fichier permet à Docker de lancer automatiquement l'ensemble des services, en les reliant entre eux. Le port 5044 est utilisé par Logstash pour recevoir les logs envoyés par les agents Filebeat ou Winlogbeat. Le port 5601 donne accès à l'interface de Kibana, et le port 9200 permet d'interagir avec l'API Elasticsearch.

c) Lancement de la stack

Depuis le dossier contenant ce fichier :

```
docker compose up -d
```

Cette commande démarre les trois conteneurs en arrière-plan. Les logs peuvent être suivis avec :

```
docker compose logs -f
```

d) Résultat attendu

Kibana devient accessible à l'adresse suivante :

<http://192.168.1.12:5601>

L'interface se lance, bien que vide dans un premier temps, en attendant la réception des premiers logs.

La suite du déploiement concerne les autres services Dockerisés : Netdata, Suricata et Zabbix, qui seront détaillés dans les sous-sections suivantes.

2. Déploiement de Netdata, Suricata et Zabbix

a) Déploiement de Netdata

Netdata est un outil de surveillance en temps réel permettant d'afficher, via une interface web intuitive, les performances système détaillées (CPU, RAM, disques, trafic réseau, connexions actives, etc.) de la machine hôte. Il est particulièrement utile pour identifier en un coup d'œil les anomalies ou les pics de charge.

La commande suivante a été utilisée pour installer Netdata dans un conteneur Docker :

```
docker run -d \
  --name=netdata \
  -p 19999:19999 \
  -v /etc/passwd:/host/etc/passwd:ro \
  -v /etc/group:/host/etc/group:ro \
  -v /proc:/host/proc:ro \
  -v /sys:/host/sys:ro \
  -v /etc/os-release:/host/etc/os-release:ro \
  --cap-add SYS_PTRACE \
  --security-opt apparmor=unconfined \
  netdata/netdata
```

Cette commande monte certains fichiers système en lecture seule afin de permettre à Netdata de collecter les métriques réelles du système hôte. Le port 19999 est exposé pour accéder à l'interface web.

L'interface de Netdata est ensuite disponible via :

<http://192.168.1.12:19999>

Grâce à Netdata, les administrateurs peuvent surveiller visuellement l'état du serveur et réagir rapidement à toute surcharge, saturation disque ou activité réseau anormale.

b) Déploiement de Suricata (IDS)

Suricata est un système de détection d'intrusion (IDS) open-source qui permet d'analyser le trafic réseau en temps réel afin de repérer d'éventuelles attaques, scans ou comportements anormaux.

Voici la commande utilisée pour lancer Suricata en mode Docker :

```
docker run -d --name suricata \
  --net=host \
  --cap-add=NET_ADMIN \
  --cap-add=NET_RAW \
  -v /var/log/suricata:/var/log/suricata \
  jasonish/suricata:latest -i enp0s8
```


L'option `--net=host` permet à Suricata d'écouter le trafic de l'hôte directement. L'interface `enp0s8` correspond à celle connectée à `vmbr1`. Les journaux sont stockés dans `/var/log/suricata`, notamment le fichier `eve.json` utilisé pour l'analyse des alertes. Suricata est intégré avec ELK pour que ses alertes soient visibles dans Kibana.

c) Déploiement de Zabbix (supervision)

Zabbix est un outil de supervision très complet permettant de surveiller l'état des serveurs, des services et d'émettre des alertes en cas d'anomalies. Il est déployé avec Docker à l'aide d'un fichier `docker-compose.yml` contenant les services suivants :

- `mysql` : base de données
- `zabbix-server` : moteur de supervision
- `zabbix-web` : interface web

Après avoir configuré la base MySQL (utilisateur, base de données, privilèges), les conteneurs Zabbix sont lancés avec :

```
docker compose -f docker-compose-zabbix.yml up -d
```

L'interface Zabbix est accessible via :

<http://192.168.1.12:8080>

Zabbix permet de surveiller aussi bien la machine Debian hôte que les postes Windows grâce à l'installation d'agents spécifiques sur les clients.

Grâce à Netdata, les administrateurs peuvent surveiller visuellement l'état du serveur et réagir rapidement à toute surcharge, saturation disque ou activité réseau anormale. Il complète efficacement les outils de supervision avancée comme Zabbix en apportant une visibilité instantanée des ressources.

XI. Installation des agents (Zabbix, Filebeat, Winlogbeat)

Pour permettre la supervision complète du réseau et la collecte des journaux d'événements, plusieurs agents ont été installés sur les machines concernées.

- Sur la machine **Debian Docker**, l'agent **Zabbix** a été installé afin de remonter les informations système (utilisation CPU, mémoire, disque, services actifs) directement au serveur Zabbix hébergé sur la même machine.
- Sur le **poste Windows Client**, l'agent **Filebeat** a été installé pour envoyer les fichiers de logs système vers **Logstash**, qui se charge de les filtrer et de les structurer avant de les transmettre dans Elasticsearch pour être visualisés dans Kibana.
- Le rôle de **Winlogbeat** a également été présenté. Cet agent permet de centraliser les journaux d'événements Windows (système, sécurité, application). Dans notre projet, il a été mentionné comme solution alternative, mais nous avons choisi d'utiliser Filebeat pour simplifier la configuration sur le poste de test.

Ces agents assurent la centralisation et la supervision des événements réseau et systèmes dans un environnement homogène, structuré, et supervisé à travers les outils mis en place (Zabbix, ELK).

XII. Installation de Windows Server 2022

L'installation de Windows Server 2022 constitue une étape clé du projet car cette machine sera promue en tant que contrôleur de domaine Active Directory. Elle jouera

un rôle central dans la gestion des utilisateurs, des groupes, des politiques de sécurité (GPO) et des ressources du réseau.

Création de la VM sous Proxmox

La machine virtuelle a été créée via l'interface Proxmox avec les paramètres suivants :

- Nom : windows-server
- ISO utilisé : fr_windows_server_2022_x64_dvd.iso
- 2 vCPU
- 4 Go de RAM
- 40 Go de disque
- Carte réseau : connectée à vmbr1 (réseau interne)

Installation du système

1. Démarrage sur l'ISO de Windows Server 2022.
2. Sélection de la langue, du clavier et du format horaire : français (France).
3. Choix de l'édition : Windows Server 2022 Standard (avec interface graphique).
4. Installation personnalisée sur le disque vierge alloué.
5. Définition d'un mot de passe administrateur fort.
6. Redémarrage automatique une fois l'installation terminée.

Configuration réseau initiale

Après le premier démarrage, une IP statique a été attribuée à la carte réseau de la VM :

- Adresse IP : 192.168.1.10
- Masque : 255.255.255.0
- Passerelle : 192.168.1.1 (pfSense)
- DNS : 127.0.0.1 (auto-référence une fois AD installé)

Ces paramètres sont appliqués via le Centre Réseau et Partage > Modifier les paramètres de l'adaptateur > Propriétés IPv4.

Préparation du serveur

Avant d'installer les rôles Active Directory, DNS et DHCP, les opérations suivantes ont été effectuées :

- Renommage de la machine en SRV-AD.
- Application des mises à jour via Windows Update.
- Désactivation du pare-feu pour faciliter les tests initiaux (réactivé plus tard).

La machine est maintenant prête pour la promotion en tant que contrôleur de domaine.

XIII. Promotion en contrôleur de domaine et installation du rôle DNS

Une fois le serveur prêt, nous avons procédé à la promotion de la machine SRV-AD en tant que contrôleur de domaine Active Directory. Cette étape permet de centraliser la gestion des utilisateurs, des postes, des permissions, et d'implémenter le rôle DNS.

Ajout des rôles Active Directory et DNS

1. Ouverture du Gestionnaire de serveur.
2. Clic sur Ajouter des rôles et fonctionnalités.
3. Sélection de l'installation basée sur un rôle ou une fonctionnalité.
4. Sélection du serveur local SRV-AD.
5. Choix des rôles : Services de domaine Active Directory (AD DS) et Serveur DNS.
6. Validation des dépendances proposées.
7. Démarrage de l'installation.

Une fois les rôles installés, une notification s'affiche pour proposer la promotion du serveur en tant que contrôleur de domaine.

Promotion du serveur en contrôleur de domaine

1. Clic sur Promouvoir ce serveur en contrôleur de domaine.
2. Sélection de Ajouter une nouvelle forêt.
3. Nom de domaine racine : logiware.local
4. Définition d'un mot de passe pour le mode de restauration (DSRM).
5. Configuration automatique du service DNS.
6. Chemins par défaut pour les bases de données, les journaux et le SYSVOL.
7. Vérification de la configuration puis lancement de la promotion.

Le serveur redémarre automatiquement à la fin de l'opération.

Résultat attendu

Après le redémarrage, la machine SRV-AD est désormais un contrôleur de domaine Active Directory. Le rôle DNS est fonctionnel et utilisé par toutes les machines du réseau. Le domaine logiware.local est actif et prêt à recevoir les premières unités organisationnelles (OU), les comptes utilisateurs et les stratégies de groupe (GPO).

XIV. Création des OU, utilisateurs, groupes et affectations

Afin d'organiser correctement les utilisateurs et d'appliquer les GPO de manière structurée, plusieurs unités organisationnelles (OU) ont été créées dans l'Active Directory du domaine logiware.local. Ces OU permettent de regrouper les utilisateurs selon leur fonction ou service dans l'entreprise fictive.

Structure des OU créées

Nom de l'OU	Description
Admins	Contient les comptes d'administration
Direction	Comptes des utilisateurs de la direction
IT	Utilisateurs du service informatique
Comptabilite	Personnel du service comptabilité
Stagiaires	Comptes temporaires pour les stagiaires

Liste des utilisateurs créés

Nom complet	Identifiant	Mot de passe	OU associée	Groupe (si applicable)
Jean Dupont	jdupont	Mdp@1234	Direction	
Alice Martin	amartin	Alice@123	Comptabilite	
Kevin Leroy	kleroy	Kevin2024	IT	
Claire Dubois	cdubois	Claire2024	IT	
Thomas Bernard	tbernard	Thomas@2024	Admins	Administrateurs
Hugo Lefevre	hlefevre	Hugo!stag	Stagiaires	
Pauline Girard	pgirard	Pauline@2024	Comptabilite	
Admin Principal	admin	Admin@logiware	Admins	Administrateurs

Création des groupes

Des groupes de sécurité ont été créés pour faciliter l'attribution des droits par GPO :

- **G_IT_AccesServeurs** : accès à certains partages ou ressources internes pour le service IT
- **G_Compta_PDF** : accès aux imprimantes et outils spécifiques à la comptabilité
- **G_Stagiaires_Limites** : restrictions d'accès spécifiques aux comptes stagiaires

XV. Création, configuration et application des GPO

Les stratégies de groupe (GPO) permettent de contrôler de manière centralisée le comportement des postes utilisateurs et des comptes dans le domaine. Dans notre infrastructure, plusieurs GPO ont été créées pour renforcer la sécurité, gérer les périphériques, les connexions à distance ou encore le mappage d'imprimantes.

GPO mises en place

Nom de la GPO	Objectif principal	OU concernée(s)
GPO-Mot-de-Passe-Expiration	Forcer le changement de mot de passe tous les 90 jours	Toutes les OU
GPO-Firewall	Activer et configurer le pare-feu Windows	Toutes les OU
GPO-Limitation-RDP	Restreindre l'accès au bureau à distance aux administrateurs	Toutes sauf Admins
GPO-Sécurité-Compte	Renforcement des politiques de sécurité (verrouillage de compte)	Toutes les OU
GPO-Désactivation-USB	Désactiver l'utilisation des périphériques de stockage USB	Stagiaires
GPO-Mappage-Imprimante	Affecter une imprimante réseau aux utilisateurs comptables	Comptabilité

Application des GPO

Les GPO ont été liées aux OU correspondantes via la console GPMC (Group Policy Management Console). Chaque GPO a été soigneusement testée en environnement contrôlé pour s'assurer de son bon fonctionnement.

Des filtres de sécurité ont été appliqués pour exclure les administrateurs lorsque nécessaire (ex : GPO-Limitation-RDP), et l'héritage a été désactivé dans certaines OU sensibles (Admins) afin de conserver un contrôle total sur les stratégies appliquées.

XVI. Intégration des postes clients au domaine

Une fois les OU, les utilisateurs, les groupes et les GPO créés et testés, il a été nécessaire d'intégrer les postes clients Windows au domaine pour appliquer les stratégies configurées et centraliser la gestion.

Configuration IP du poste client Windows

Avant l'intégration au domaine, le poste client Windows 10 a été configuré avec une adresse IP statique afin de garantir une communication stable avec le contrôleur de domaine et les autres services réseau.

Les paramètres réseau appliqués sont les suivants :

- Adresse IP : 192.168.1.20
- Masque de sous-réseau : 255.255.255.0
- Passerelle par défaut : 192.168.1.1 (pfSense)
- Serveur DNS préféré : 192.168.1.10 (contrôleur de domaine SRV-AD)

Ces réglages sont essentiels pour que le client puisse résoudre le nom de domaine logiware.local et communiquer avec le serveur DNS interne.

Étapes de l'intégration d'un poste Windows au domaine

1. Sur le poste client (Windows 10), ouvrir les Paramètres système.
2. Cliquer sur Nom du PC > Renommer ce PC (avancé).
3. Cliquer sur Modifier et sélectionner l'option Domaine.
4. Saisir le nom du domaine : logiware.local.
5. Entrer les identifiants d'un utilisateur ayant les droits d'intégration (ex : admin).
6. Un message de bienvenue dans le domaine confirme la réussite.
7. Redémarrer le poste pour finaliser l'intégration.

Résultat attendu

Le poste client est désormais intégré au domaine logiware.local, il est automatiquement placé dans l'OU par défaut ou déplacé manuellement dans la bonne OU. Les stratégies de groupe définies (pare-feu, mot de passe, désactivation USB, etc.) sont appliquées selon l'OU et le groupe de l'utilisateur connecté.

XVII. Déploiement de Security Onion et intégration au réseau

Security Onion est une distribution Linux spécialisée dans la détection et l'analyse des intrusions réseau. Elle regroupe plusieurs outils puissants comme Suricata, Zeek, Wazuh, Elasticsearch et Kibana dans une plateforme unifiée. Dans notre projet, elle est utilisée pour renforcer la surveillance du réseau et compléter la supervision déjà assurée par Suricata et ELK sur Debian.

Installation de la VM Security Onion

La machine virtuelle Security Onion a été déployée via Proxmox en important l'image ISO officielle. Les paramètres de la VM sont :

- Nom : security-onion
- 2 vCPU
- 4 Go de RAM
- 50 Go de disque
- Carte réseau connectée à vmbr1

Configuration initiale

Après démarrage sur l'ISO :

1. Sélection du mode "Evaluation".
2. Interface réseau détectée automatiquement (enp0s8).
3. Configuration d'une IP statique : 192.168.1.30/24, passerelle 192.168.1.1, DNS 192.168.1.10.
4. Attribution d'un mot de passe administrateur.
5. Sélection des services à activer : Suricata, Zeek, Elasticsearch, Kibana.
6. Lancement du script de configuration automatique.

Résultat attendu

Une fois la configuration terminée, Security Onion est opérationnelle et commence à analyser le trafic transitant sur le réseau local. Grâce à ses modules intégrés (comme Suricata pour l'analyse des paquets, Zeek pour l'inspection du comportement réseau, et Wazuh pour l'analyse des logs système), elle fournit une vue d'ensemble précise sur l'état de sécurité de l'infrastructure. Les alertes générées sont centralisées et accessibles via l'interface web de Kibana, également intégrée dans la solution.

Security Onion détecte automatiquement les flux suspects, les tentatives de scan, les anomalies comportementales ou encore les connexions inhabituelles, ce qui permet à l'administrateur réseau d'agir de manière proactive. Cette solution complète vient renforcer le niveau de sécurité global du projet en offrant une surveillance en profondeur et en temps réel du réseau.

XVIII. Tableaux récapitulatifs (IPs, GPO, utilisateurs, services)

1. Plan d'adressage IP

Nom de la machine	Adresse IP	Rôle / Fonction
pfSense (LAN)	192.168.1.1	Passerelle, pare-feu, NAT, DNS
Windows Server (AD)	192.168.1.10	Contrôleur de domaine, DNS
Debian Docker	192.168.1.12	Hôte des services Docker : ELK, Zabbix, Netdata
Windows Client	192.168.1.20	Poste utilisateur client intégré au domaine
Security Onion	192.168.1.30	IDS/NSM avec Suricata, Zeek, Kibana

2. GPO appliquées par OU

Nom de la GPO	Objectif	OU ciblée(s)
GPO-Mot-de-Passe-Expiration	Changement obligatoire tous les 90 jours	Toutes les OU
GPO-Firewall	Activation et configuration du pare-feu Windows	Toutes les OU
GPO-Limitation-RDP	RDP réservé aux admins	Toutes sauf Admins
GPO-Sécurité-Compte	Verrouillage après tentatives échouées	Toutes les OU
GPO-Désactivation-USB	Blocage des périphériques USB	Stagiaires
GPO-Mappage-Imprimante	Attribution automatique d'une imprimante réseau	Comptabilite

3. Services Docker déployés

Nom du service	Port d'accès	Adresse interne (Debian Docker)	Description
Kibana	5601	192.168.1.12	Interface de visualisation ELK
Elasticsearch	9200	192.168.1.12	Moteur de stockage de logs

Logstash	5044	192.168.1.12	Réception et traitement des logs
Netdata	19999	192.168.1.12	Supervision temps réel
Suricata (Docker)	(via eve.json)	192.168.1.12	Détection d'intrusions réseau (IDS)
Zabbix Frontend	8080	192.168.1.12	Interface de supervision
Zabbix Server	10051	192.168.1.12	Serveur de collecte
MySQL (Zabbix)	3306	192.168.1.12	Base de données Zabbix

XIX. Conclusion du projet

Ce projet a permis de mettre en œuvre une infrastructure réseau complète et sécurisée en environnement virtualisé. Grâce à Proxmox, plusieurs machines virtuelles ont été déployées et configurées pour reproduire les besoins réels d'une entreprise :

- La mise en place de pfSense a permis de contrôler le trafic réseau et d'assurer une passerelle sécurisée entre les VMs et Internet.
- Un serveur Windows Server 2022 a été configuré comme contrôleur de domaine pour centraliser la gestion des utilisateurs, des stratégies de sécurité (GPO), et des ressources.
- Un poste client Windows a été intégré au domaine pour tester les politiques et la connexion aux services.
- Une VM Debian a été utilisée pour héberger plusieurs services critiques via Docker, notamment la supervision (Zabbix, Netdata), la collecte et visualisation de logs (ELK), et un IDS réseau (Suricata).
- Enfin, la VM Security Onion a renforcé la détection des menaces par une analyse avancée du trafic.

Ce projet a mis en œuvre des compétences multiples : déploiement d'infrastructure virtuelle, administration réseau, sécurité informatique, supervision, gestion centralisée des utilisateurs et politiques, tout en assurant la documentation claire de chaque étape. Il constitue une base solide pour toute entreprise souhaitant sécuriser, surveiller et centraliser son système d'information.