Documentation projet E5

Contexte du projet :

Le projet consiste à intervenir sur la refonte du système d'information (SI) de la société **Health North**, un leader européen des services de diagnostic médical. L'entreprise, suite à une fusion majeure, doit moderniser et unifier les infrastructures de ses cliniques et laboratoires pour répondre aux besoins croissants en termes de disponibilité, sécurité et résilience.

Je suis donc chargé de travailler sur l'architecture réseau et système d'une **clinique basée en Guadeloupe**. L'objectif est de mettre en place des services essentiels et d'assurer la continuité des opérations.

Objectifs principaux du projet :

1. **Renforcer l'infrastructure réseau et système** :
    - Assurer la disponibilité des services vitaux (DNS, DHCP, AD).
    - Créer une architecture résiliente pour minimiser les interruptions.
2. **Configurer des services annexes pour l'entreprise :**
    - Implémenter un système de supervision pour surveiller l'état des équipements et services.
    - Automatiser la gestion des sauvegardes pour prévenir la perte de données.
3. **Documenter et structurer l'architecture réseau :**
    - Produire un schéma clair du réseau logique et physique.
    - Définir un plan d'adressage et des procédures détaillées.

Enjeux et contraintes :

- **Haute disponibilité** : Les services doivent rester accessibles, même en cas de panne d'un équipement.
- **Sécurité** : Protéger le réseau contre les cyberattaques (malwares, ransomware).
- **Interconnexion** : Permettre aux équipes distantes (Guadeloupe/métropole) de collaborer efficacement via des solutions sécurisées (ex. : VPN).

Objectifs du projet :

Le projet vise à moderniser et sécuriser l'infrastructure réseau et système de la clinique de Health North, en Guadeloupe. Les objectifs principaux sont les suivants :

1. **Assurer la disponibilité des services essentiels :**

- Active Directory (AD) : Mettre en place un système centralisé pour la gestion des utilisateurs, groupes et ressources.
- DNS : Garantir la résolution des noms de domaine internes les serveurs et postes clients.
- DHCP : Automatiser l'attribution des adresses IP pour simplifier l'administration réseau.

2. Renforcer la sécurité et la résilience du réseau :

Sécurisation des services :

- Implémenter des stratégies (GPO) pour protéger les utilisateurs et les données.
- Désactiver les protocoles inutilisés comme IPv6 si nécessaire.
- Firewall : Configurer un pare-feu pour contrôler les flux réseau et bloquer les accès non autorisés.
- Segmentation VLAN : Isoler les différents services (administration, laboratoire, direction) pour limiter les risques d'intrusion.

3. Documenter l'architecture réseau et système :

- Schéma réseau : Produire des schémas logique et physique.
- Plan d'adressage : Définir les plages IP et la configuration des VLANs.
- Procédures : Rédiger des guides pour l'installation, la configuration, et la maintenance des services.

4. Ajouter des services annexes pour enrichir l'infrastructure :

Supervision :

- Mettre en place un outil comme Zabbix pour surveiller les serveurs et équipements.
- Configurer des alertes pour anticiper les pannes.

Sauvegardes :

- Implémenter un système de sauvegarde automatisée pour les données critiques.

Hébergement web :

- Déployer un site ou une application interne pour les employés de la clinique (par exemple, un portail d'information ou un système de tickets).

5. Tester et valider l'infrastructure mise en place :

- Vérifier la communication entre les services (AD, DNS, DHCP).
- Simuler des scénarios de panne et de récupération pour garantir la résilience.
- Préparer un plan de continuité d'activité (PCA).

Avec Rufus, création d'une clef bootable Proxmox VE.

La machine où est l'environnement virtuel proxmox est un ordinateur avec 16Go de RAM et 128Go de stockage.

Lancement et configuration de base pour la création du serveur Proxmox.

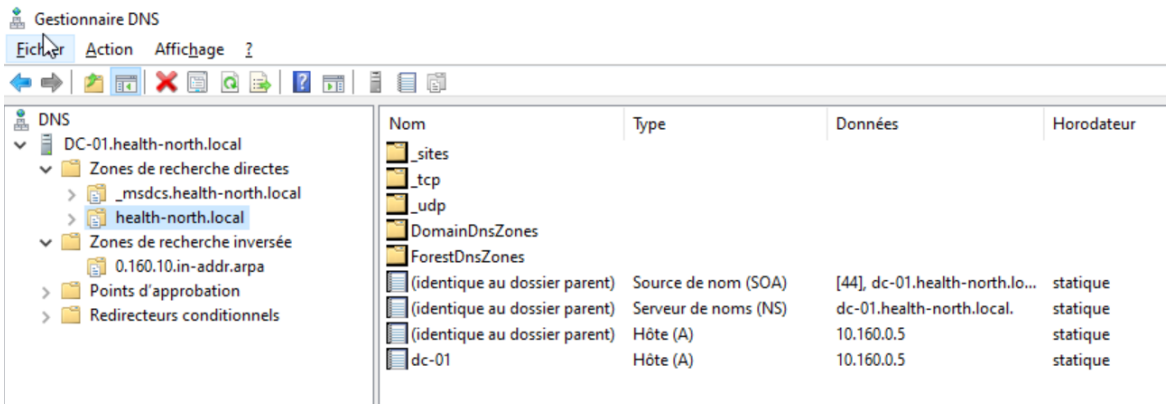
Désactivation des dépôts entreprises pour les mises à jour et ajout d'un dépôt « no subscription ».

Création de la première VM sous Windows server 2022 :

4096Mo de RAM, 50Go de stockage.

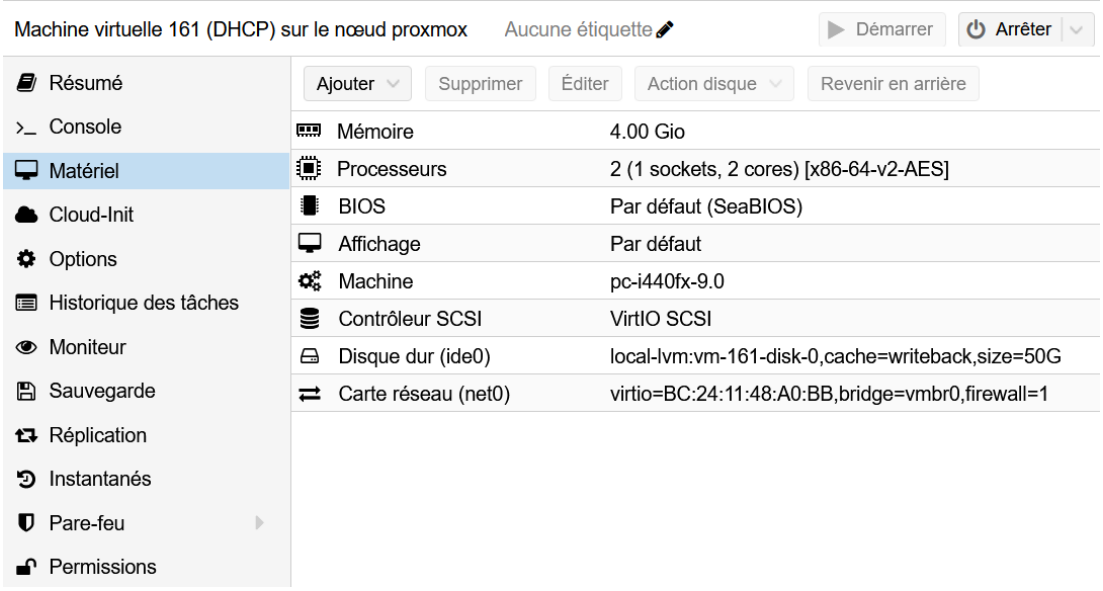
Installation des drivers et des rôles AD DS et DNS.

Configuration DNS :



Création de la deuxième VM aussi sous Windows server 2022 :

Configuration globale :



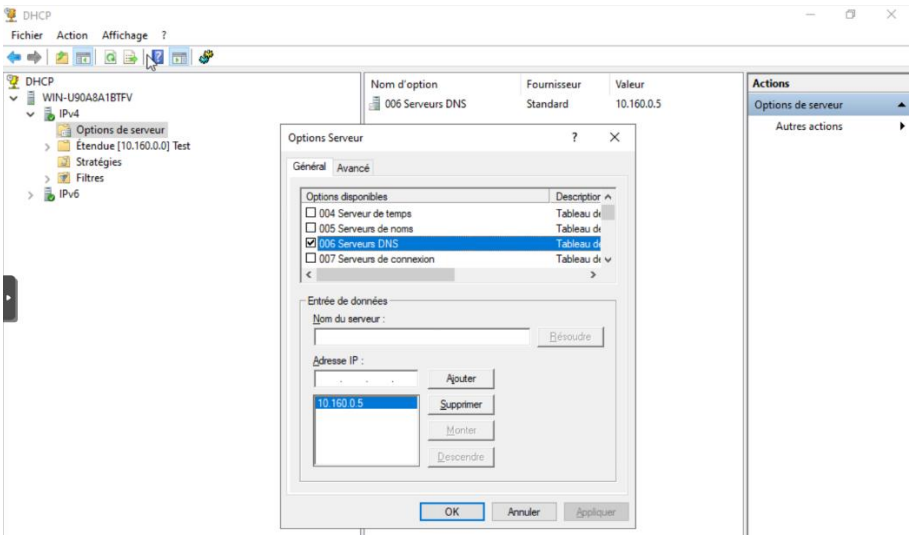
Installation des drivers et du rôle DHCP.

Problèmes rencontrés et solutions

Quand j'ai essayé de ping le serveur AD depuis le serveur DHCP cela fonctionnait, mais l'inverse échouait à cause du pare-feu Windows qui bloquait les requêtes ICMPv4. L'autorisation des pings a résolu ce problème.

Quand j'ai testé la résolution du nom de domaine et de l'IP pour m'assurer du bon fonctionnement du DNS, les serveurs AD/DNS et DHCP ne les résolvaient pas correctement à cause de l'utilisation de l'IPv6. J'ai simplement désactivé l'utilisation de l'ipv6.

Le serveur DHCP attribuait ma box internet comme DNS par défaut, empêchant la résolution du domaine interne. L'ajout de l'IP du serveur DNS (10.160.0.5) dans l'option 006 du serveur DHCP a corrigé cela :

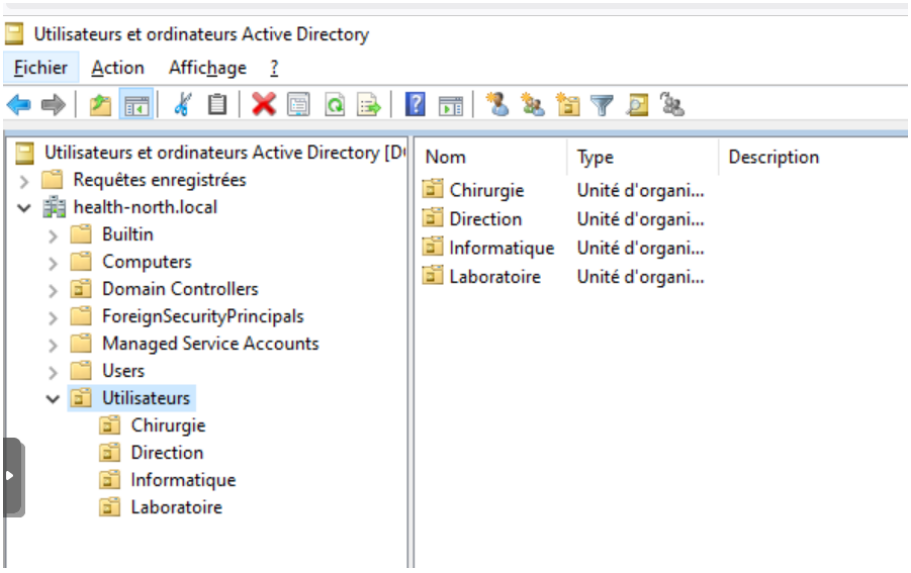


Sur le serveur AD/DNS, la commande nslookup que j'utilisais pour résoudre l'IP et le nom de domaine, me renvoyait encore ma box internet comme DNS, perturbant la résolution. La configuration manuelle pour utiliser son propre DNS et la désactivation de l'IPv6 ont réglé ce problème.

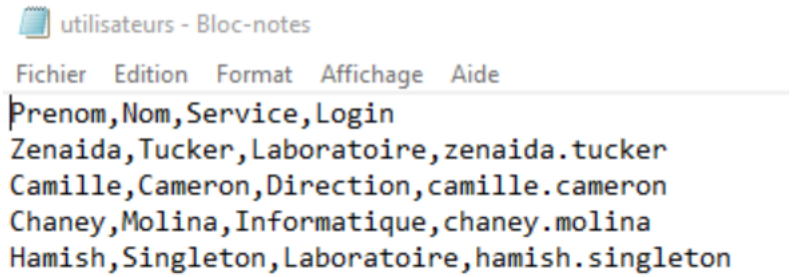
Ces ajustements ont permis de stabiliser la communication réseau et d'assurer à ce stade, la bonne communication entre les deux serveurs.

Configuration de l'AD :

Une matrice était donnée dans le sujet de l'épreuve par Studi. J'ai commencé par créer les unités d'organisations (OU) :



Puis j'ai repris chaque nom prénom, service pour commencer à créer des utilisateurs. Dans le sujet, il était demandé de créer un script PowerShell pour créer automatiquement les utilisateurs dans l'AD avec le nom, prénom, service et mail. J'ai commencé par créer un fichier csv pour y répertorier les noms prénoms, login et service de chaque utilisateur. L'objectif est que le script aille chercher ces informations pour les traitées et les créés dans l'AD. Ce fichier sert en quelque sorte de base de données, il est le point de départ du script. Chaque ligne correspond à un utilisateur unique, chaque colonne correspond à une propriété spécifique pour cet utilisateur. (Nom, prénom, service, login).



Puis j'ai réalisé le script PowerShell suivant :

*AjoutUtilisateur - Bloc-notes

Fichier Edition Format Affichage Aide

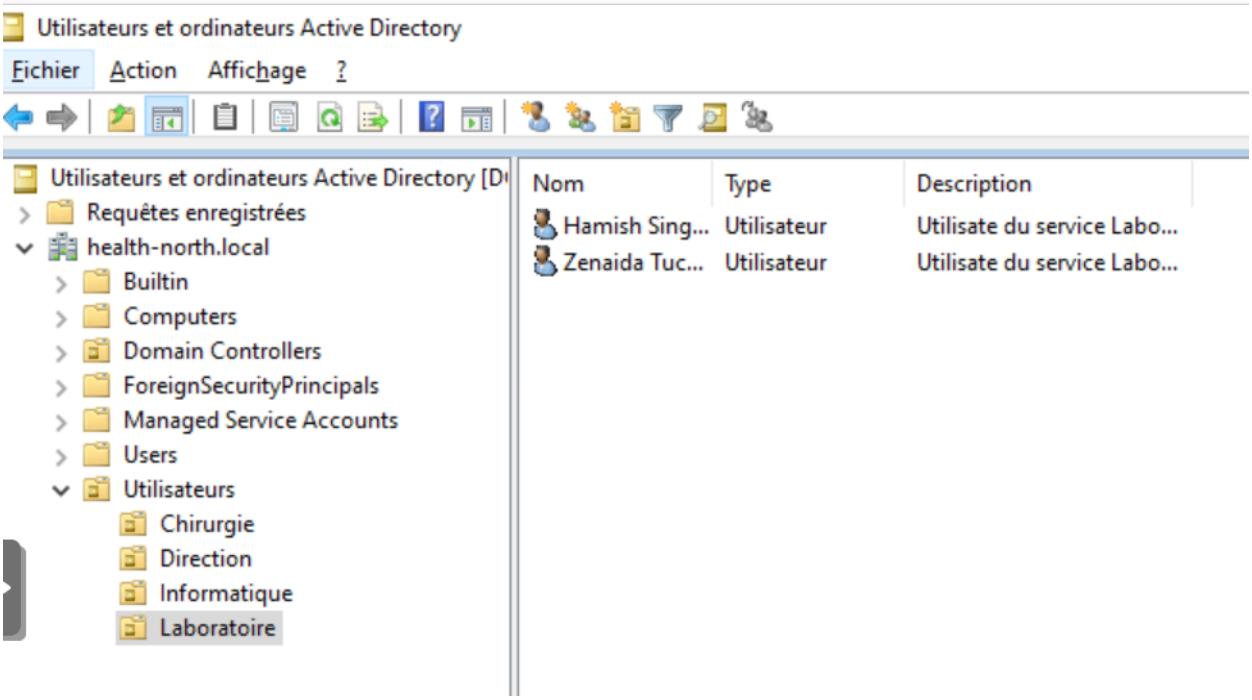
```
        #Charger le module Active Directory
Import-Module ActiveDirectory

$CsvPath = "C:\utilisateurs.csv"

Import-Csv $CsvPath | ForEach-Object {
        #Variables de l'utilisateur
        $Prenom = $_.Prenom
        $Nom = $_.Nom
        $Service = $_.Service
        $Login = $_.Login
        $OU = "OU=$Service,OU=Utilisateurs,DC=health-north,DC=local"

#Générer l'adresse mail
$Email = "$Login@health-north.fr"

#Créer l'utilisateur dans l'Active Directory
New-ADUser -Name "$Prenom $Nom" `
        -GivenName $Prenom `
        -Surname $Nom `
        -SamAccountName $Login `
        -Path $OU `
        -AccountPassword (ConvertTo-SecureString "MotDePasse123!" -AsPlainText -Force) `
        -Enabled $true `
        -Description "Utilisate du service $Service" `
        -PasswordNeverExpires $false `
        -ChangePasswordAtLogon $true

#Afficher un message de confirmation
Write-Host "Utilisateur $Prenom $Nom ajouté avec succès dans l'OU $Service"
}
```

Et cela a bien fonctionné :

Administrateur : Windows PowerShell

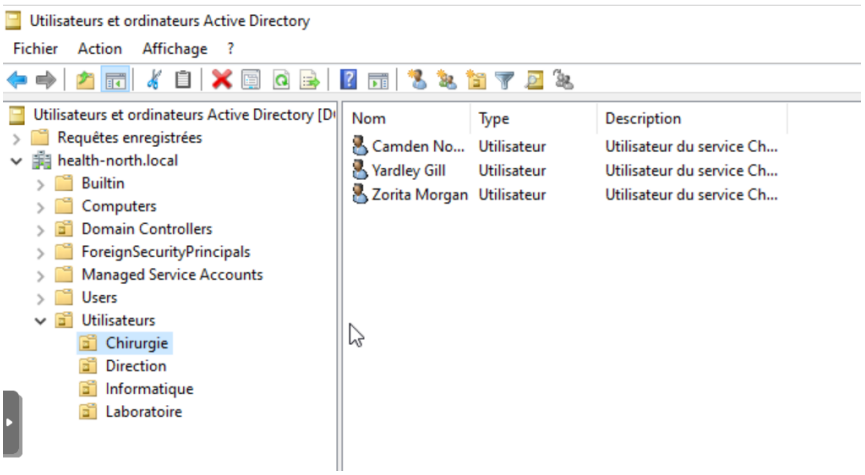
```
Windows PowerShell
Copyright (C) Microsoft Corporation. Tous droits réservés.

Installez la dernière version de PowerShell pour de nouvelles fonctionnalités et améliorations ! https://aka.ms/PSWindow
s

PS C:\Users\Administrateur> cd C:\
PS C:\> .\AjoutUtilisateur.ps1
Utilisateur Zenaida Tucker ajoutÃ© avec succÃ¨s dans l'OU Laboratoire
Utilisateur Camille Cameron ajoutÃ© avec succÃ¨s dans l'OU Direction
Utilisateur Chaney Molina ajoutÃ© avec succÃ¨s dans l'OU Informatique
Utilisateur Hamish Singleton ajoutÃ© avec succÃ¨s dans l'OU Laboratoire
PS C:\>
```

On peut voir que les utilisateurs ont effectivement été créés :



L'objectif de ce script réside aussi dans sa praticité, si je veux ajouter un utilisateur, je n'ai qu'à l'ajouter au fichier csv en indiquant son nom, prénom, service et login puis exécuter le script.

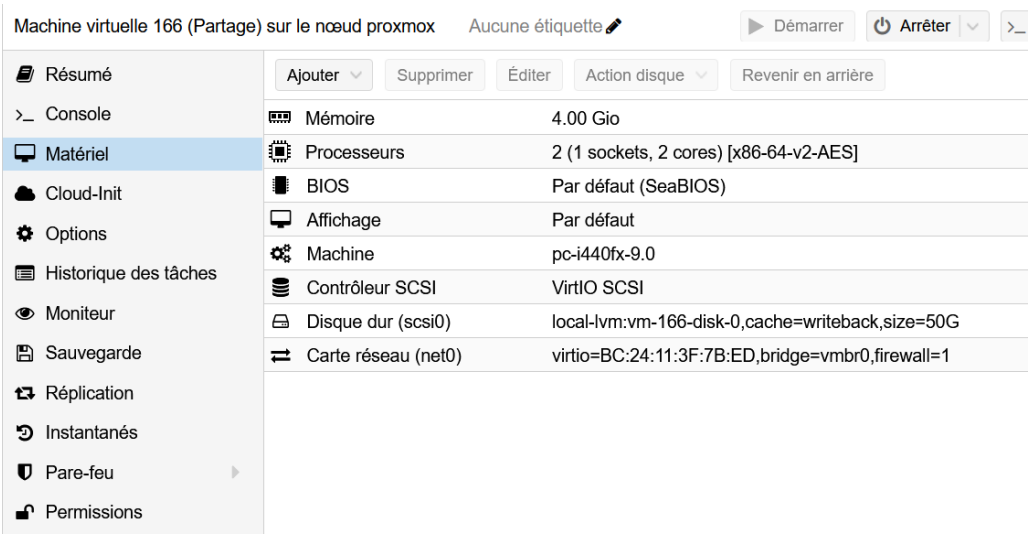
J'ai ensuite modifié le script pour que quand on modifie le fichier csv, le script ne recréé pas les utilisateurs déjà créés auparavant. Suite à cela, j'ai exécuté le script modifié et tout a fonctionné parfaitement.



Création des profils itinérants :

L'objectif principal des profils itinérants est de permettre à chaque utilisateur du domaine de retrouver son environnement (Bureau, Documents, paramètres) sur n'importe quelle machine, tout en centralisant leurs données sur le serveur de partage.

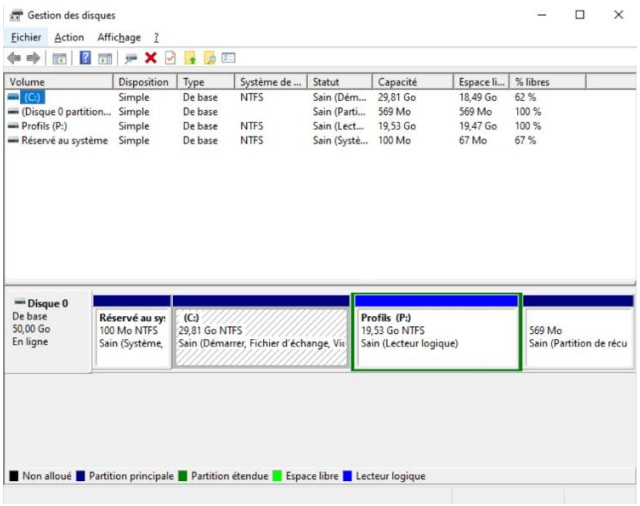
J'ai donc décidé de continuer sur les profils utilisateurs, et de créer les profils itinérants de chaque utilisateur. J'ai créé une nouvelle VM qui sera dédiée au stockage des profils itinérants et qui fera aussi office de serveur de partage de fichiers. La VM a les mêmes caractéristiques que les deux autres à savoir 4Go de RAM, 50Go de stockage le tout sous Windows server 2022.



J'ai ajouté une IP statique au serveur, je l'ai ajouté au domaine et vérifier que les communications entre les 3 serveurs passaient correctement à l'aide de la commande « ping » mais aussi dans la commande « nslookup » pour la résolution DNS et l'intégration au domaine.

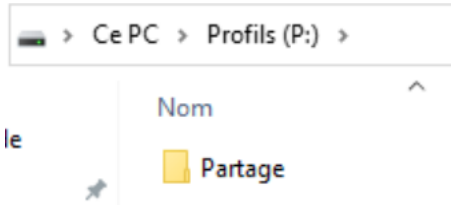
J'ai commencé par partitionner, dans le gestionnaire de disque, le disque du serveur en gardant 30Go pour le système et 20Go pour le reste à savoir le stockage des profils itinérants et le partage de fichiers.

Partitionner le disque, c'est surtout pour garder les choses bien organisées. Les profils itinérants ont leur espace à part, ce qui évite de tout mélanger avec le système ou d'autres fichiers. C'est pratique pour les sauvegardes, ça limite les risques d'erreurs, et en cas de problème, tu sais exactement où chercher. Bref, c'est plus propre et plus simple à gérer.



1. Préparation du Dossier Partage

J'ai commencé par créer un dossier nommé **Partage** sur le disque dédié (P:\Partage). Ce dossier sert de point d'accès centralisé pour les profils itinérants et d'autres partages futurs.



Ensuite, j'ai configuré les permissions réseau et NTFS pour répondre aux besoins fonctionnels et sécuritaires :

- **Partage avancé (partage réseau):**
    - **Admins du domaine** et **Utilisateurs authentifiés** ont reçu un contrôle total pour permettre l'accès et la gestion des fichiers.
- **Permissions NTFS** :
    - Les groupes suivants ont été configurés avec des rôles bien définis :
        - **Admins du domaine**, **Système**, et **l'administrateur local de la machine** : Contrôle total.
        - **Utilisateurs authentifiés** : Lecture uniquement.

Problème rencontré et ajustements

Lors des premiers tests, j'ai constaté que la synchronisation des profils échouait régulièrement, affichant des messages d'erreur indiquant une synchronisation incomplète lors de la déconnexion des utilisateurs.

Pour identifier la source du problème, j'ai temporairement modifié les permissions du dossier **Partage**, en attribuant **Contrôle total à « Tout le monde »**, tant sur le partage réseau que sur les permissions NTFS. Cette méthode m'a permis de confirmer que le problème était bien lié aux permissions, car tout fonctionnait bien une fois les permissions totales accordées à tout le monde, l'objectif était d'écartés un problème plus profond ou complexe, non lié aux permissions.

Une fois cette cause identifiée, j'ai affiné les configurations :

- J'ai remplacé « **Tout le monde »** par « **Utilisateurs authentifiés »**, ce qui limite l'accès aux seuls utilisateurs connectés au domaine.
- J'ai également vérifié que les groupes nécessaires (**Admins du domaine**, **Système**, etc.) avaient bien un contrôle total.

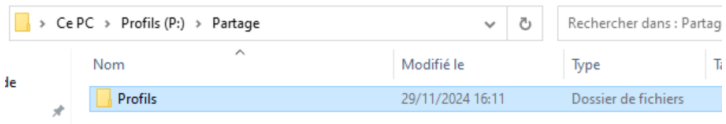
Grâce à ces ajustements, les problèmes de synchronisation ont été résolus, et la configuration est désormais fonctionnelle et sécurisée.

2. Création et Configuration du Dossier Profils

Dans le dossier **Partage**, j'ai créé un sous-dossier nommé **Profils** pour stocker les profils itinérants. Ce dossier n'a pas été partagé, car l'accès se fait via le partage du dossier parent.

Les permissions NTFS sur **Profils** ont été configurées comme suit :

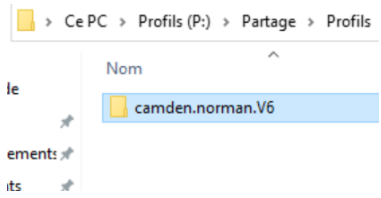
- **Héritage désactivé** pour éviter l'application de permissions non pertinentes.
- **Admins du domaine**, **Système**, et l'administrateur local de la machine ont reçu un contrôle total.
- Chaque utilisateur, comme **Camden Norman**, a été ajouté manuellement avec un contrôle total sur son propre dossier une fois créé.



3. Configuration dans Active Directory

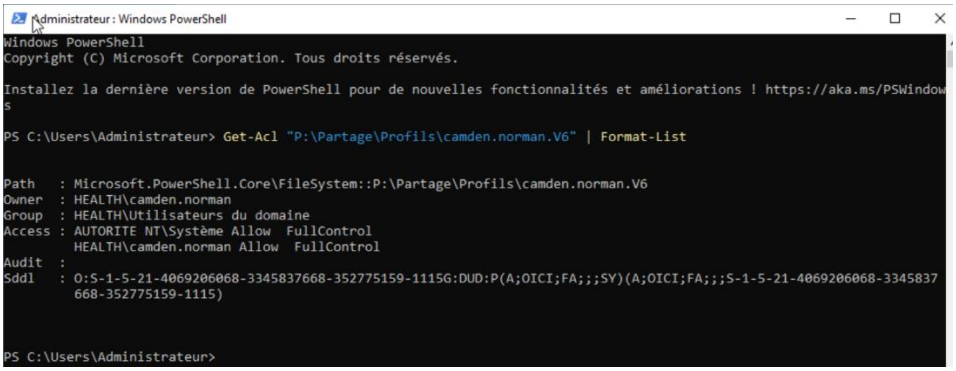
Dans les propriétés de chaque utilisateur dans l'AD, le chemin du profil itinérant a été défini comme :
\\10.160.0.10\Partage\Profils\%username%.

Cette configuration permet au système de créer automatiquement un dossier utilisateur à la première connexion. Les permissions spécifiques sont alors appliquées dynamiquement par Windows.



Problèmes rencontrés et solutions

- **Accès aux dossiers utilisateur bloqué** :
  En testant les permissions des dossiers comme **Camden.Norman.V6**, nous avons constaté que l'administrateur ne pouvait pas consulter les permissions directement. Ce comportement a été compris comme une mesure de sécurité par Windows. Si nécessaire, les permissions peuvent être vérifiées via PowerShell ou en modifiant temporairement le propriétaire. J'ai opté pour la vérification avec PowerShell et j'ai pu constater les permissions attribuées à ce dossier par Windows et ainsi m'assurer des bonnes pratiques.



5. Tests finaux

Une série de tests a confirmé le bon fonctionnement des profils itinérants :

- Connexions et déconnexions sur plusieurs postes.
- Création, modification, et suppression de fichiers dans le profil.
- Synchronisation correcte des données sans message d'erreur.

Bien que plusieurs ajustements aient été nécessaires, notamment au niveau des permissions, la configuration actuelle est robuste et conforme aux bonnes pratiques. Les prochaines étapes consistent à appliquer cette configuration pour tous les utilisateurs restants.

J'aurais pu faire ces étapes par GPO, mais je trouvais ça plus intéressant de les faire manuellement, cela améliore aussi la compréhension et les connaissances des différentes permissions que l'on peut accorder par exemple.

Implémentation des partages réseau automatique pour les services

Par la suite, j'ai mis en place une solution pour automatiser l'accès aux partages réseau des différents services via des lecteurs mappés. L'objectif principal était de garantir que chaque utilisateur puisse accéder directement aux données de son service dès sa connexion, tout en respectant les restrictions d'accès aux autres services.

Pour structurer cette configuration, j'ai créé un sous-dossier appelé « **Services »** dans le dossier « Partage » qui se trouve sur le serveur de partage. Ce dossier regroupe les sous-dossiers de chaque service (Laboratoire, Chirurgie, Informatique, Direction), chacun étant isolé pour répondre aux besoins spécifiques de chaque équipe. L'objectif de ce dossier était de regrouper tous les partages réseau, facilitant ainsi leur accès pour les utilisateurs.

Cette configuration a été réalisée avec succès grâce à une combinaison de permissions NTFS, de partages réseau, et de stratégies de groupe (GPO).

Création des partages réseau pour les services

1. Organisation des dossiers

Les dossiers de chaque service ont été créés dans la structure existante sur le serveur de partage, sous le dossier principal Services. Voici la structure finale :

- \\10.160.0.10\Partage\Services\Laboratoire
- \\10.160.0.10\Partage\Services\Chirurgie
- \\10.160.0.10\Partage\Services\Informatique
- \\10.160.0.10\Partage\Services\Direction

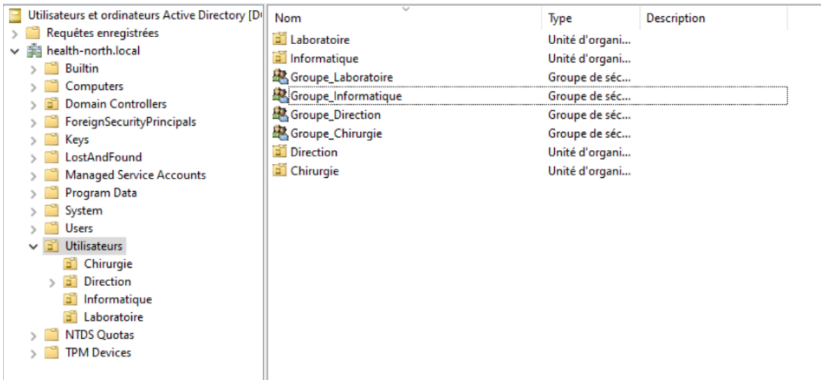
Chaque dossier est dédié à un service précis et ne peut être accédé que par les utilisateurs de ce service.

2. Configuration des permissions NTFS et de partage

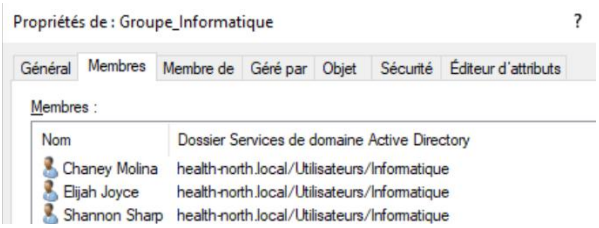
Création des groupes pour les services

Avant de configurer les permissions, nous avons créé des groupes dans Active Directory pour chaque service afin de gérer les accès de manière centralisée et efficace. Ces groupes permettent de définir les permissions pour chaque dossier sans avoir à gérer chaque utilisateur individuellement. Les groupes créés sont :

- **Groupe_Laboratoire**
- **Groupe_Chirurgie**
- **Groupe_Informatique**
- **Groupe_Direction**



Chaque utilisateur a ensuite été ajouté au groupe correspondant à son service. Cette approche facilite la gestion des droits d'accès, surtout si de nouveaux utilisateurs rejoignent les services à l'avenir.



Pour assurer une gestion optimale des accès et garantir une **séparation stricte** des services, j'ai mis en place des **permissions spécifiques** pour chaque sous-dossier dédié à un service (Laboratoire, Chirurgie, Informatique, Direction). En effet, **l'héritage des permissions** a été désactivé sur chacun de ces sous-dossiers afin de mieux contrôler les droits d'accès et éviter toute propagation non souhaitée de permissions depuis les dossiers parents ou de niveau supérieur.

L'héritage des permissions dans Windows permet à un dossier de **transmettre** ses permissions à ses sous-dossiers, ce qui peut être pratique dans certains cas, mais dans le cadre de ma configuration, cela aurait pu poser problème. En désactivant l'héritage, je me suis assuré que :

- **Chaque service** dispose d'un contrôle exclusif sur ses fichiers et dossiers, sans que des permissions indésirables provenant du dossier Services ou du niveau supérieur ne soient appliquées.
- **Les droits sont clairs et précis :** En d'autres termes, seul le groupe d'utilisateurs lié à un service peut accéder à ses propres fichiers et dossiers, réduisant ainsi les risques d'accès non autorisé.

- **Une meilleure sécurité :** J’ai veillé à ce que les utilisateurs n'aient accès qu’aux ressources nécessaires à leur travail, sans être exposés à des données appartenant à d'autres services.

Permissions NTFS :

- **Admins du domaine** : Contrôle total.
- **Groupe_[NomDuService]** (par exemple, Groupe_Chirurgie) : Modifier.
- **Système** : Contrôle total.

Ces permissions garantissent que :

- Les administrateurs peuvent gérer tous les dossiers.
- Chaque utilisateur d’un service peut uniquement accéder, modifier ou supprimer des fichiers dans son propre dossier.

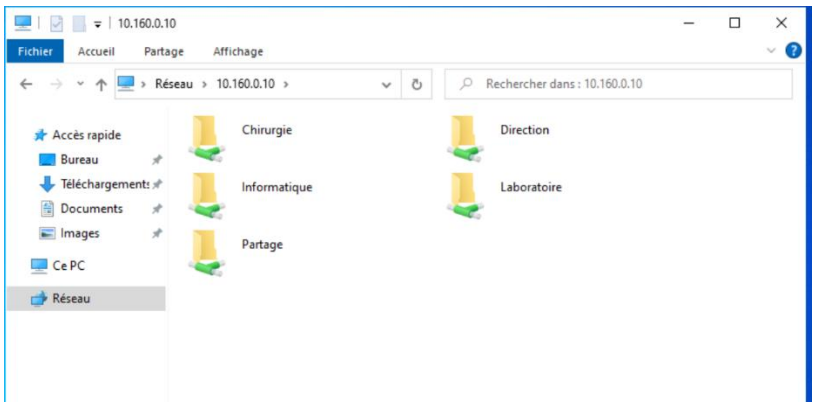
Permissions de partage :

Pour chaque dossier, les permissions de partage ont été configurées comme suit :

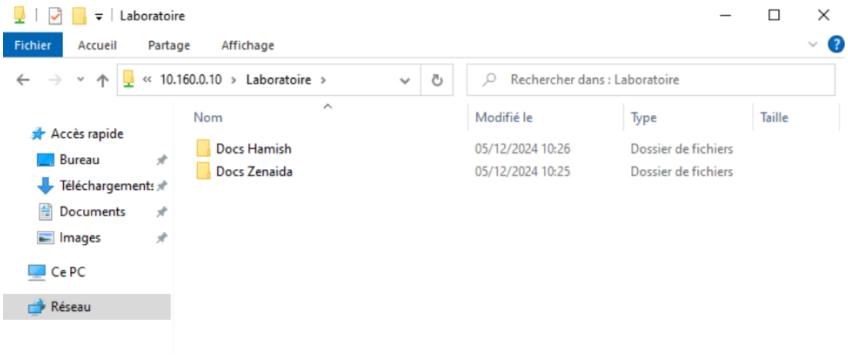
- **Admins du domaine** : Contrôle total.
- **Groupe_[NomDuService]** : Modifier ou Contrôle total selon les besoins.

Une attention particulière a été portée à la cohérence entre les permissions NTFS et de partage pour éviter tout conflit.

Quand un utilisateur accédait au partage cela donnait ceci :



J’ai fait quelques tests pour vérifier que chaque utilisateur de chaque service accédait au bon dossier :



Et que l’accès était bien refusé selon les permissions si l’utilisateur essayait d’accéder à un service qui n’était pas le sien :

Automatisation des lecteurs réseau via GPO

1. Création des GPO

Quatre stratégies de groupe ont été créées, une pour chaque service :

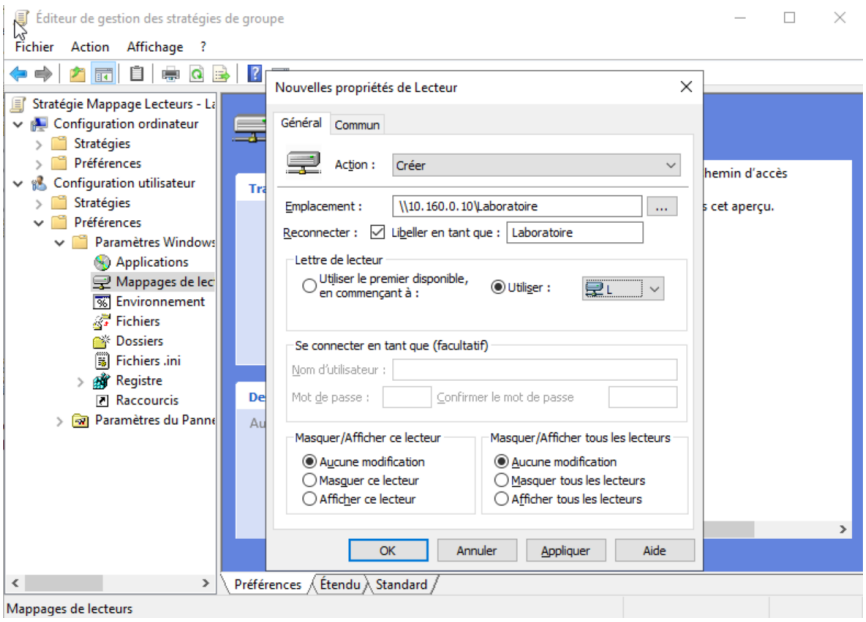
- **Mappage Lecteurs - Laboratoire**
- **Mappage Lecteurs - Chirurgie**
- **Mappage Lecteurs - Informatique**
- **Mappage Lecteurs - Direction**

Ces GPO permettent de mapper automatiquement un lecteur réseau pour les utilisateurs de chaque service.

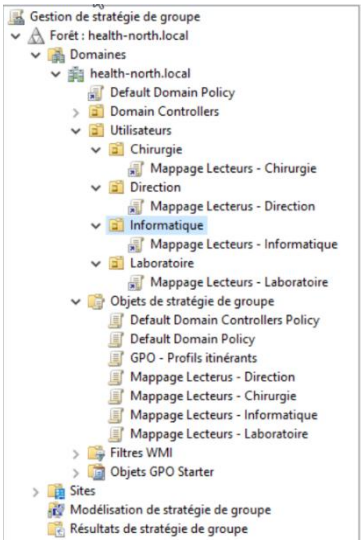
2. Configuration des lecteurs dans chaque GPO

Pour chaque GPO :

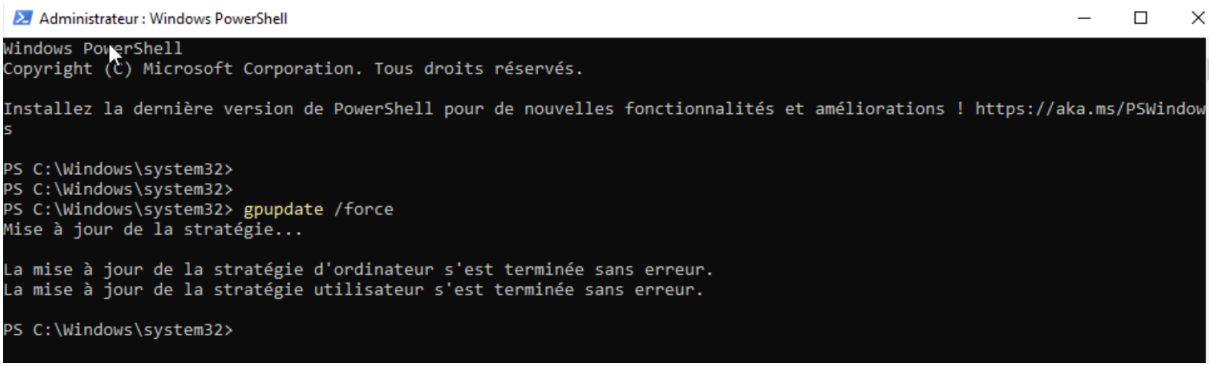
- Le chemin réseau du service a été configuré (ex : \\10.160.0.10\Laboratoire).
- Une lettre de lecteur unique a été assignée :
    - L: pour Laboratoire
    - Z: pour Chirurgie (après résolution d'un conflit avec C:)
    - I: pour Informatique
    - D: pour Direction



Il fallait ensuite lier la GPO à chaque OU de l'AD qui correspondait à un service :



J'ai ensuite réalisé un gpupdate /force sur la machine utilisé pour mettre à jour les GPO :



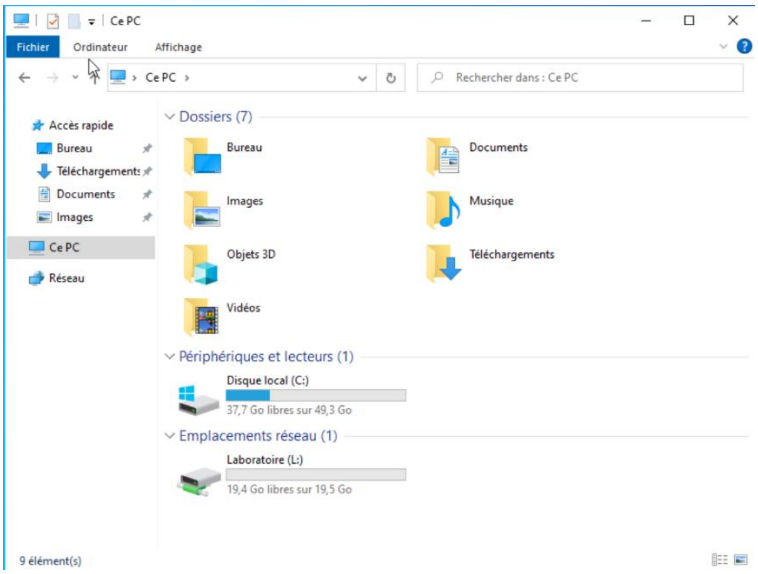
3. Résolution d'un conflit

Lors du mappage pour le service Chirurgie, un conflit est survenu en tentant d'utiliser la lettre C:, réservée au système local. Ce problème a été résolu, simplement en modifiant la lettre assignée à Z:.

Validation et tests

Les tests ont été effectués pour chaque service en utilisant plusieurs utilisateurs. Résultats :

- Chaque utilisateur a pu accéder à son partage via le lecteur réseau correspondant.
- Les permissions ont été respectées : aucun utilisateur ne pouvait accéder aux données d'un autre service.
- Les données créées par un utilisateur étaient immédiatement visibles et modifiables par d'autres membres du même service.



La configuration des partages réseau et leur automatisation via GPO ont été réalisées avec succès. Cette solution garantit :

- Une accessibilité rapide et simplifiée pour les utilisateurs.
- Une isolation stricte entre les services.
- Une gestion centralisée et évolutive grâce aux groupes Active Directory et aux GPO.

Blocage de l’invite de commande sur les postes utilisateurs

Dans le cadre de la sécurisation des postes utilisateurs, il a été décidé de **bloquer complètement l’accès à l’invite de commande Windows (cmd.exe)**, empêchant à la fois son ouverture et son utilisation. Initialement, une solution partielle avait été mise en place, permettant d’ouvrir l’invite de commande mais en bloquant son utilisation. Cependant, cette approche a été abandonnée pour garantir un contrôle total et répondre précisément aux exigences du projet.

1. GPO initiale : Blocage de l’utilisation de cmd.exe

Dans un premier temps, une GPO a été configurée pour interdire l’utilisation de l’invite de commande tout en autorisant son lancement. Cette configuration a été réalisée via le paramètre :

- **"Interdire l'accès à l'invite de commande"** (Configuration utilisateur → Modèles d'administration → Système).

Avec cette configuration :

1. Les utilisateurs pouvaient lancer cmd.exe.
2. Une fois ouverte, un message s’affichait :
   "L’administrateur a désactivé l’utilisation de l’invite de commande."
3. Lorsque l’utilisateur appuyait sur une touche, la fenêtre de l’invite de commande se fermait automatiquement.

Limites identifiées :

- Bien que l’utilisateur ne puisse rien exécuter dans l’invite de commande, la possibilité de la lancer pouvait prêter à confusion.

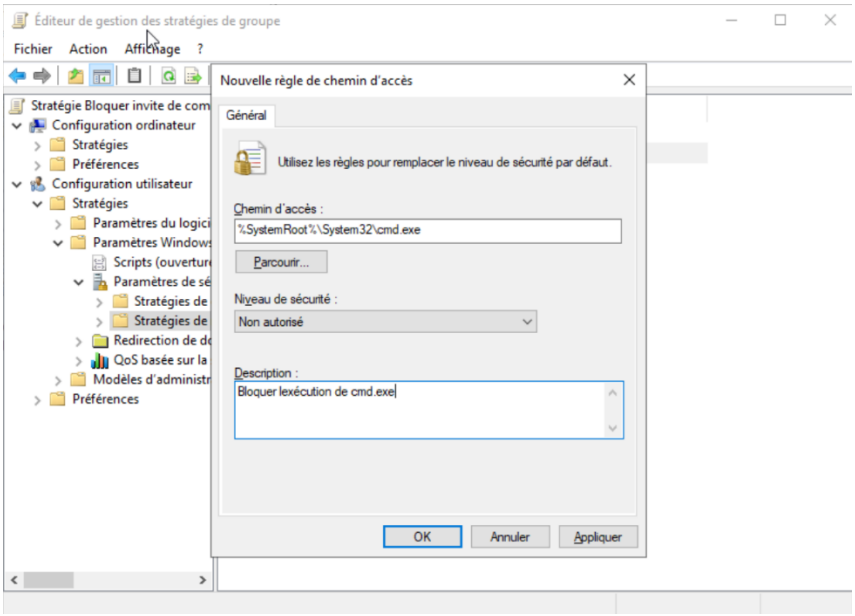
Changement de stratégie : Blocage total de cmd.exe

Pour répondre pleinement aux exigences de sécurité du projet, il a été décidé de bloquer complètement l’ouverture de l’invite de commande, empêchant ainsi tout accès, même limité.

2. GPO révisée : Stratégie de restriction logicielle (SRP)

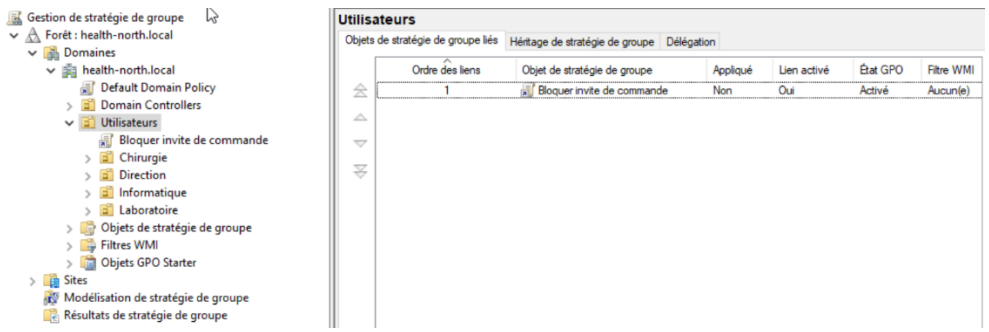
Une stratégie de restriction logicielle a été mise en place pour interdire directement l'exécution de cmd.exe. Voici comment cela a été configuré :

1. Une règle de chemin a été créée pour bloquer cmd.exe :
    - Chemin : %SystemRoot%\System32\cmd.exe.
    - Niveau de sécurité : Interdit.



2. Une règle similaire a été ajoutée pour les systèmes 64 bits :
    - Chemin : %SystemRoot%\SysWOW64\cmd.exe.
    - Niveau de sécurité : Interdit.

3. Cette GPO a été appliquée uniquement aux utilisateurs standards pour ne pas limiter l'accès des administrateurs.



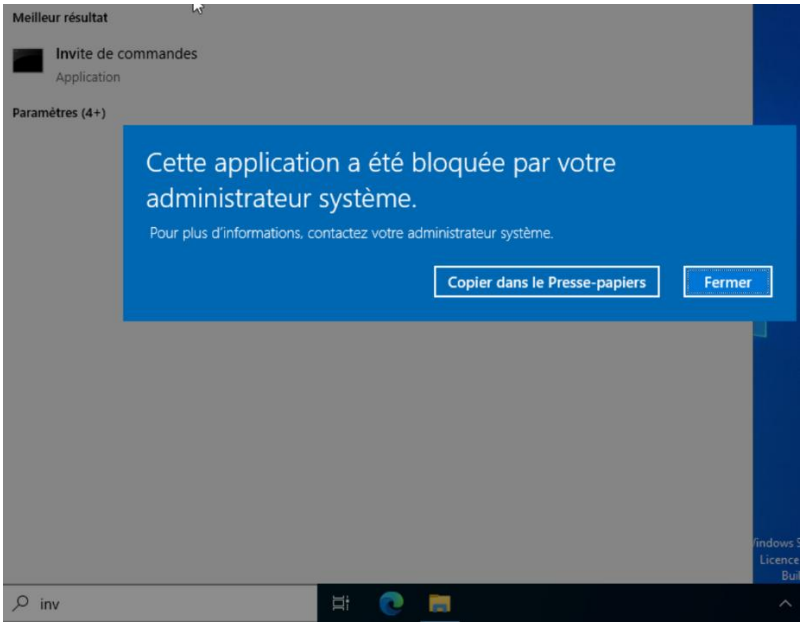
Résultats observés

Avec la première configuration (GPO initiale) :

- Lorsqu'un utilisateur lançait cmd.exe, un message indiquait que l'accès était désactivé par l'administrateur.
- L'utilisateur ne pouvait pas exécuter de commandes, et la fenêtre se fermait après avoir appuyé sur une touche.

Avec la configuration finale (Blocage total) :

1. Tentative d'accès à cmd.exe :
    - Toute tentative de lancer cmd.exe est immédiatement bloquée avec un message :



2. Blocage des scripts .bat et .cmd :
    - Les scripts .bat et .cmd sont également bloqués car leur exécution dépend de l'invite de commande.
3. Accès administrateur préservé :
    - Les administrateurs peuvent toujours accéder à cmd.exe pour effectuer des tâches d'administration.

Tests réalisés

Les tests ont été effectués avec des utilisateurs standards et administrateurs pour valider la configuration. Voici les résultats :

1. **Première configuration :**
    - L’accès à cmd.exe était limité mais possible. L’utilisateur voyait un message avant la fermeture automatique.
2. **Configuration finale :**
    - L’accès à cmd.exe est totalement bloqué pour les utilisateurs standards. Aucune fenêtre ne s’ouvre.

Après avoir testé une approche intermédiaire consistant à limiter l’utilisation de l’invite de commande tout en autorisant son lancement, un blocage total a été mis en œuvre pour répondre pleinement aux exigences du projet. Cette solution garantit une sécurité renforcée en empêchant toute tentative d’accès à l’invite de commande et à ses fonctionnalités, tout en maintenant la flexibilité nécessaire pour les administrateurs.

Politique de gestion des mots de passe conforme aux recommandations de la CNIL

Dans le cadre de la sécurisation des comptes utilisateurs du domaine, une politique de gestion des mots de passe a été mise en place pour répondre aux recommandations récentes de la CNIL. L'objectif est de garantir que les mots de passe soient suffisamment robustes pour limiter les risques de compromission tout en restant adaptés aux besoins opérationnels.

Les règles configurées permettent de renforcer la sécurité tout en préservant une expérience utilisateur acceptable.

Les préconisations de la **CNIL**, publiées en **octobre 2022**, ont été retenues comme base pour cette politique. Ces recommandations mettent l'accent sur :

1. **Longueur minimale des mots de passe : 12 caractères pour assurer une bonne entropie.**
2. **Complexité : Encourager l'utilisation de majuscules, minuscules, chiffres et caractères spéciaux.**
3. **Pas de renouvellement excessif : Éviter une expiration trop fréquente pour limiter l'adoption de mots de passe faibles.**
4. **Histoire des mots de passe : Empêcher la réutilisation des derniers mots de passe.**
5. **Blocage après échec : Verrouiller temporairement les comptes après plusieurs tentatives infructueuses.**

La politique a été mise en œuvre via la Default Domain Policy, appliquée à tous les utilisateurs du domaine. Voici les réglages configurés :

1. Longueur minimale des mots de passe

- La longueur minimale a été fixée à **12 caractères** pour garantir une robustesse suffisante.
- Paramètre configuré dans :
  Configuration ordinateur → Paramètres Windows → Paramètres de sécurité → Stratégies de compte → Stratégie de mot de passe.

2**. Complexité des mots de passe**

- La complexité a été activée pour imposer des mots de passe contenant au moins :
  - Une majuscule.
  - Une minuscule.

- Un chiffre.
- Un caractère spécial.

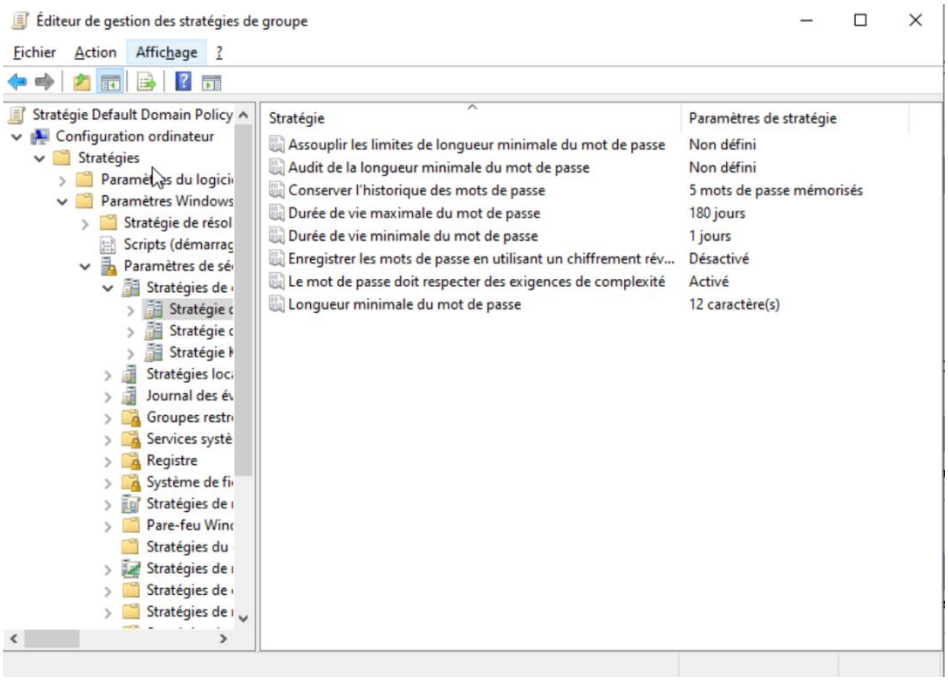
- Objectif : Empêcher l'utilisation de mots de passe trop simples ou courants.

3. Expiration des mots de passe

- Les mots de passe doivent être renouvelés tous les 180 jours (6 mois). Cette périodicité trouve un équilibre entre sécurité et praticité.
- Une durée minimale de validité de 1 jour a été configurée pour empêcher les utilisateurs de contourner les règles en changeant immédiatement leur mot de passe.

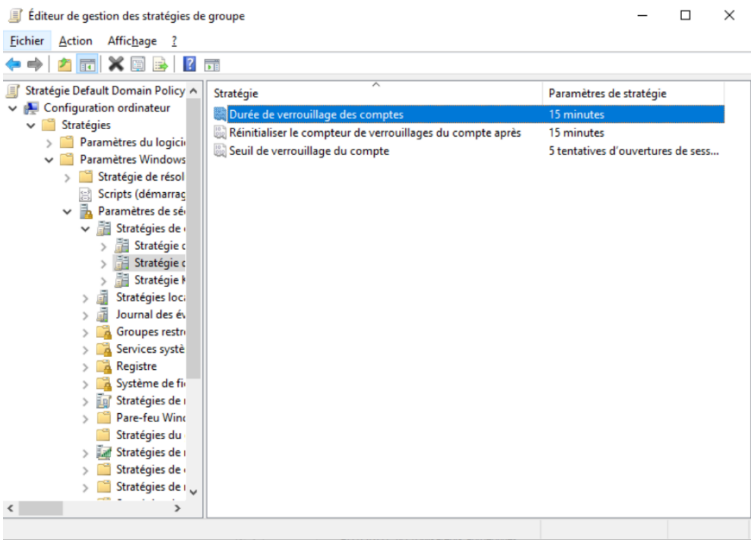
4. Histoire des mots de passe

- L'historique des mots de passe a été fixé à 5 mots de passe. Cela empêche la réutilisation des derniers mots de passe définis par l'utilisateur.



5. Verrouillage après plusieurs tentatives échouées

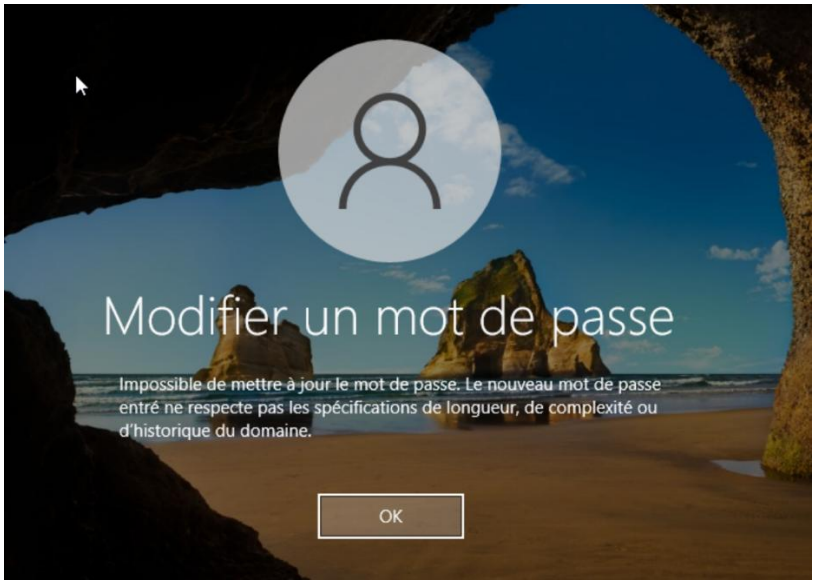
- Un verrouillage temporaire est appliqué après 5 échecs de connexion successifs.
  - Durée de verrouillage : 15 minutes.
  - Réinitialisation du compteur : 15 minutes après l'échec.
- Ce paramètre réduit les risques d'attaques par force brute tout en permettant une récupération rapide.



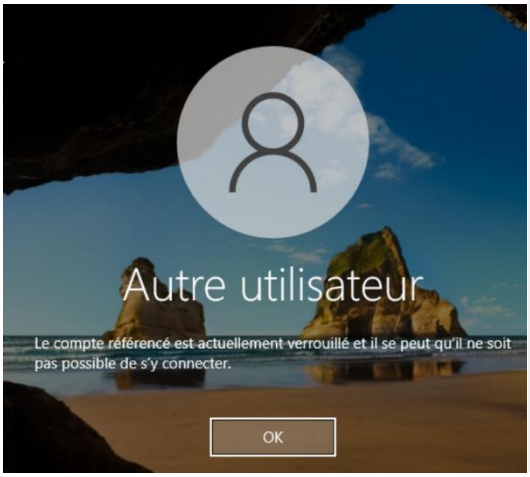
Des tests ont été réalisés pour vérifier que les paramètres configurés fonctionnaient correctement. Voici les résultats obtenus :

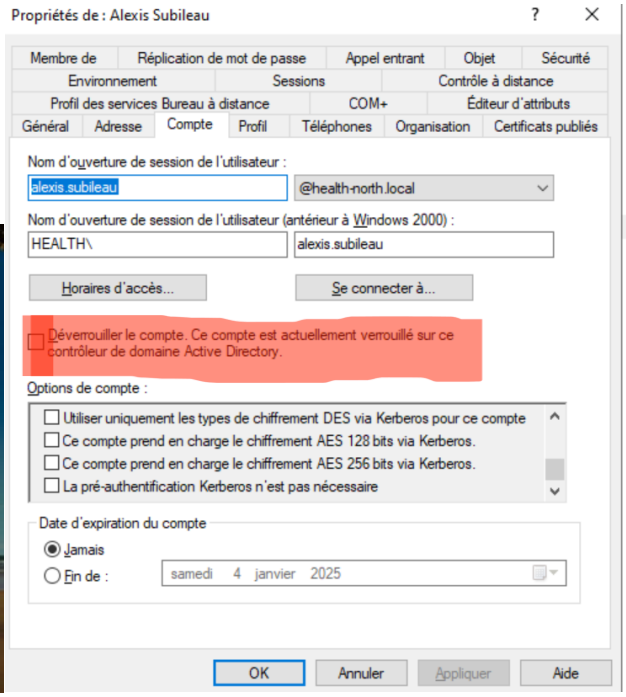
1. **Longueur minimale et complexité :**
    - Lorsqu'un utilisateur tentait de définir un mot de passe trop court ou sans complexité, le système refusait le mot de passe avec un message explicite.
2. **Expiration des mots de passe :**
    - Les mots de passe expirés imposent un renouvellement lors de la connexion, conformément aux règles définies.
3. **Historique des mots de passe :**
    - Les 5 derniers mots de passe testés ont été refusés, confirmant que l'historique était bien appliqué.

Tous ces cas avaient pour résultat ce message, ce qui prouve que la politique de mot de passe a bien été appliquée.



4. **Verrouillage après plusieurs échecs :**
    - Après 5 tentatives échouées, le compte s'est verrouillé automatiquement. Un administrateur ou le système peut débloquer le compte après 15 minutes.





La politique de gestion des mots de passe a été configurée avec succès, alignée avec les recommandations de la CNIL pour garantir une sécurité robuste des comptes utilisateurs. Ces règles permettent :

1. **Une meilleure protection contre les attaques de type brute force.**
2. **Une réduction des risques de compromission liés à des mots de passe faibles ou réutilisés.**
3. **Une gestion simplifiée grâce à l'application centralisée via la Default Domain Policy.**

Cette politique s'intègre parfaitement dans les objectifs globaux de sécurisation du domaine et répond aux besoins opérationnels tout en respectant les standards modernes de sécurité, s'alignant avec les politiques d'organismes reconnus dans le domaine de la cybersécurité comme **l'ANSSI** ou comme ici avec la **CNIL.**

Configuration et déploiement des fonds d'écran pour les utilisateurs

Dans le cadre du projet, et conformément au sujet proposé, chaque utilisateur devait recevoir un fond d’écran spécifique correspondant à son service. Cette personnalisation a été réalisée à l’aide de GPO sur Windows Server 2022.

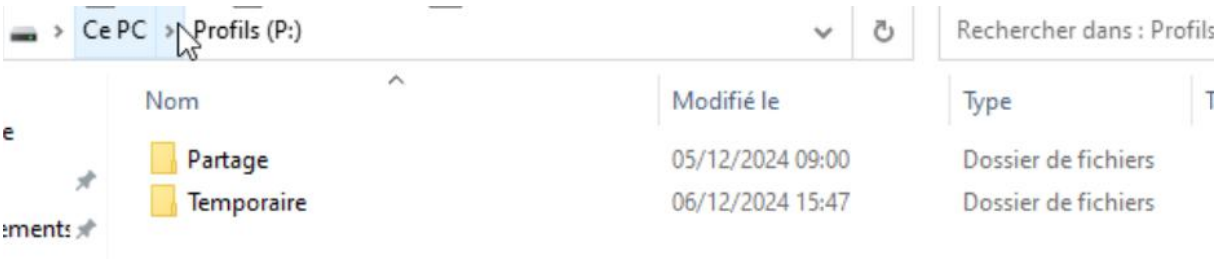
1. Préparation des fichiers de fonds d'écran

- J’ai commencé par réaliser les différents fonds d’écrans sur Canva avec un fonds d’écran par service. Les fichiers de fonds d’écran ont été créés et organisés sur mon PC personnel.
- Ils ont ensuite été convertis au format **JPG** pour assurer leur prise en charge par Windows Server.

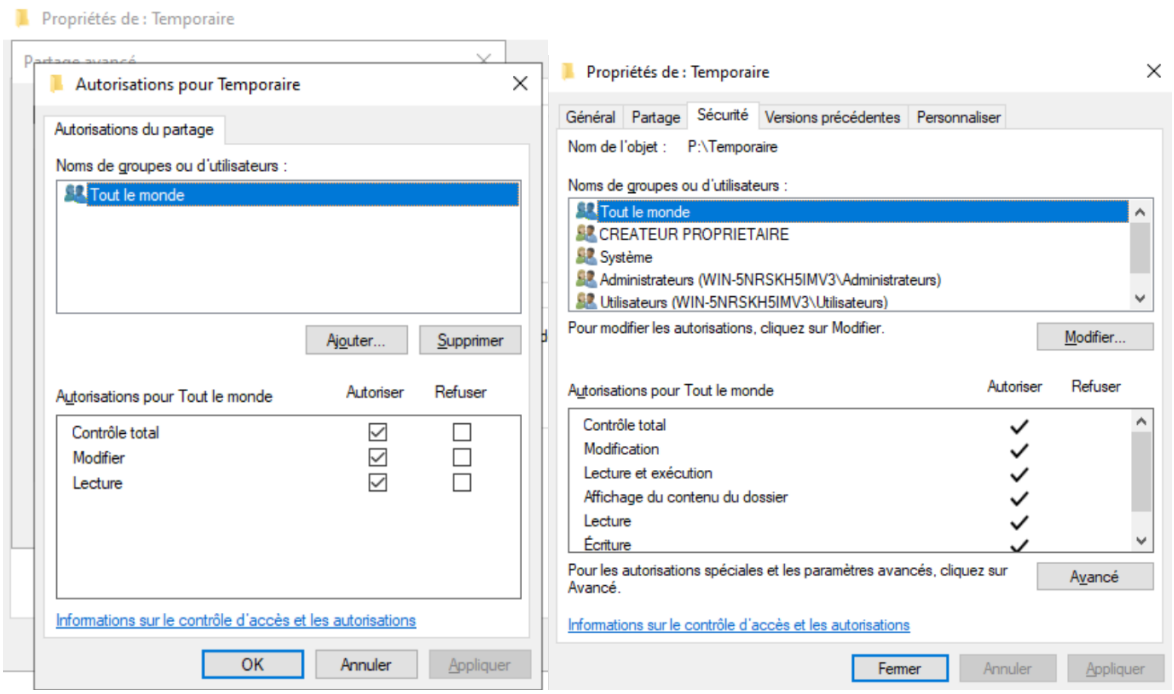
2. Transfert des fichiers vers le serveur

- J’ai configuré une adresse IP statique mon ordinateur personnel pour intégrer temporairement le réseau du projet et ainsi pouvoir joindre les fonds d’écrans sur le réseau.

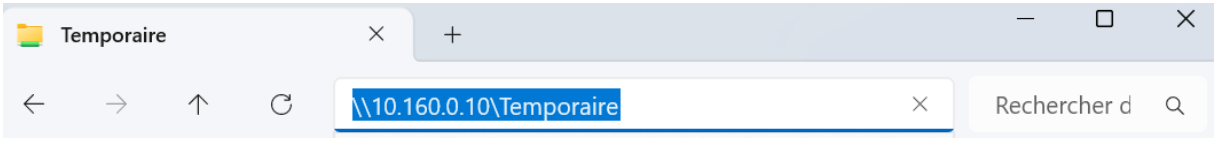
- J’ai créé un dossier nommé « Temporaire » sur le serveur de partage.



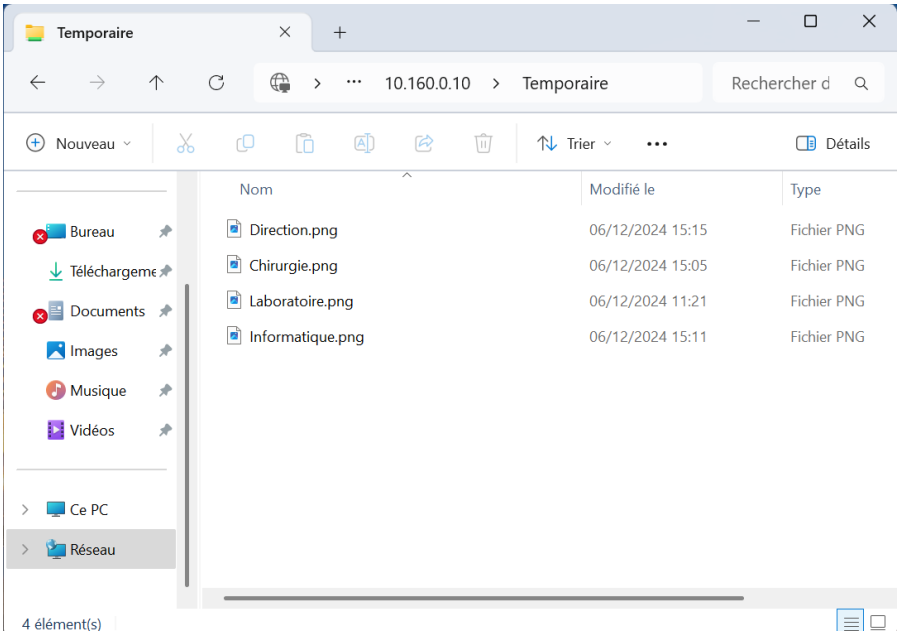
- J’ai ouvert les permissions de partage et NTFS à tout le monde en accès total pour être certain de pouvoir y accéder depuis mon PC.



- J’ai pu accéder au chemin réseau depuis mon PC.



- Les fichiers JPG ont été copiés dans le dossier partagé final : \\10.160.0.10\Partage\FondsEcran.



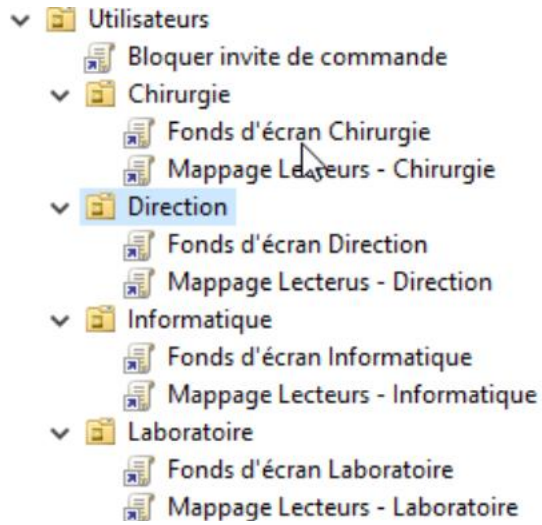
3. Configuration des permissions

- Les permissions sur le dossier **FondsEcran** ont été définies comme suit :
    - **Partage avancé** :
        - **Utilisateurs authentifiés** : Lecture seule.
        - **Admins du domaine** : Contrôle total.
    - **Sécurité NTFS** :
        - **Utilisateurs authentifiés** : Lecture seule.
        - **Admins du domaine** : Contrôle total.
        - **Système** : Contrôle total.

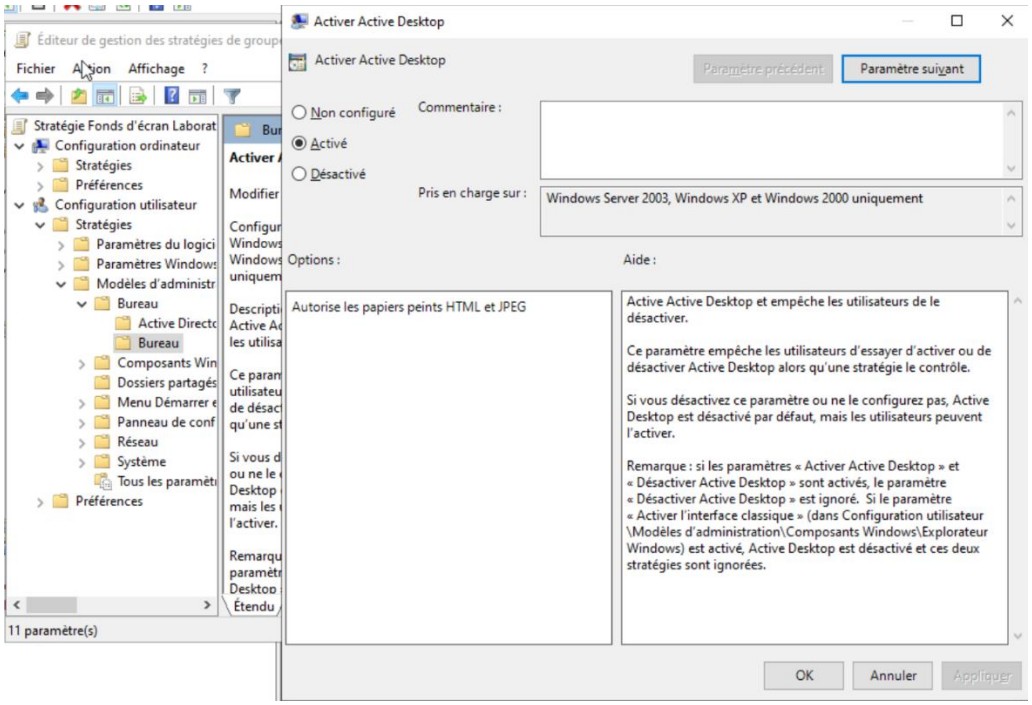
Ces configurations garantissent l'accessibilité aux fonds d'écran tout en empêchant toute modification accidentelle ou non autorisée.

4. Création des GPO pour chaque service

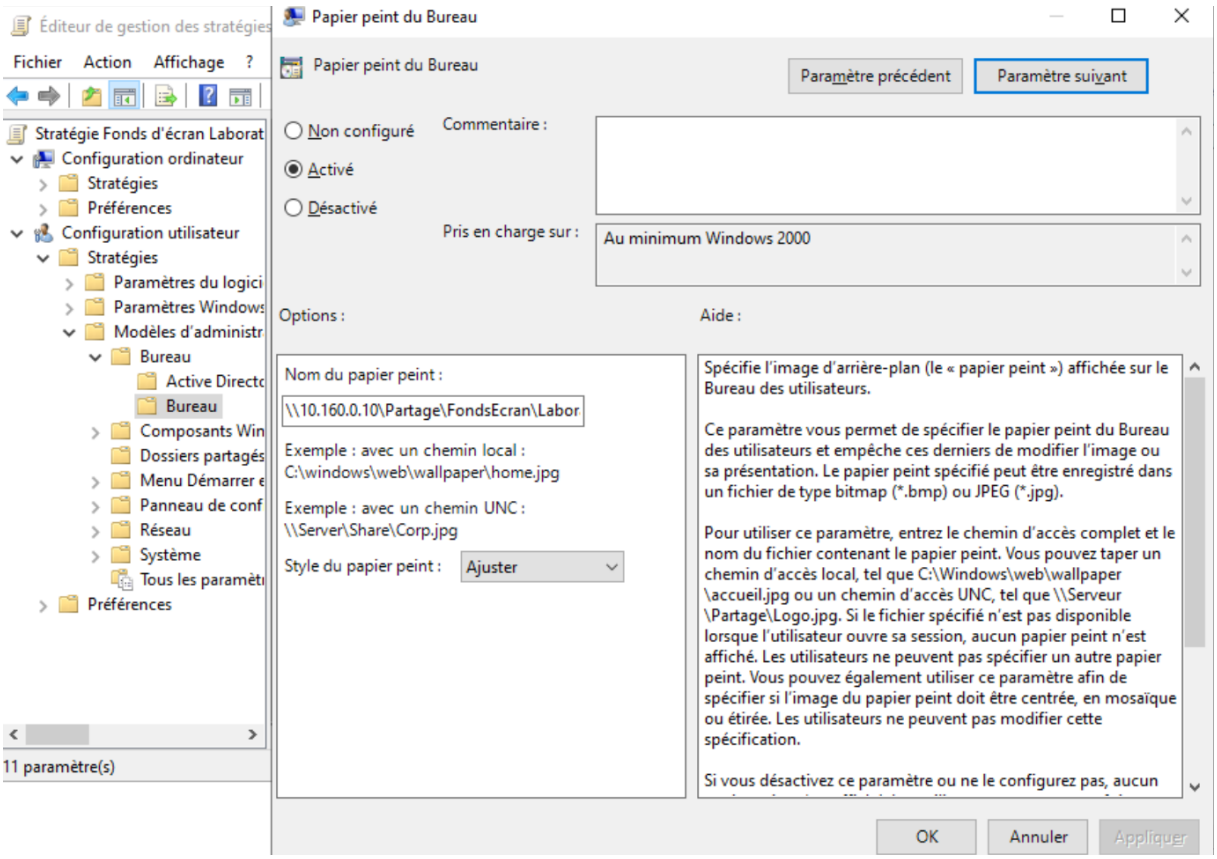
- Une GPO a été créée pour chaque service (ex. : Laboratoire, Chirurgie, Informatique, Direction).



- Il a d'abord fallu activé le paramètre « Activer Active Desktop » pour autoriser les fonds d'écrans jpg depuis une GPO.



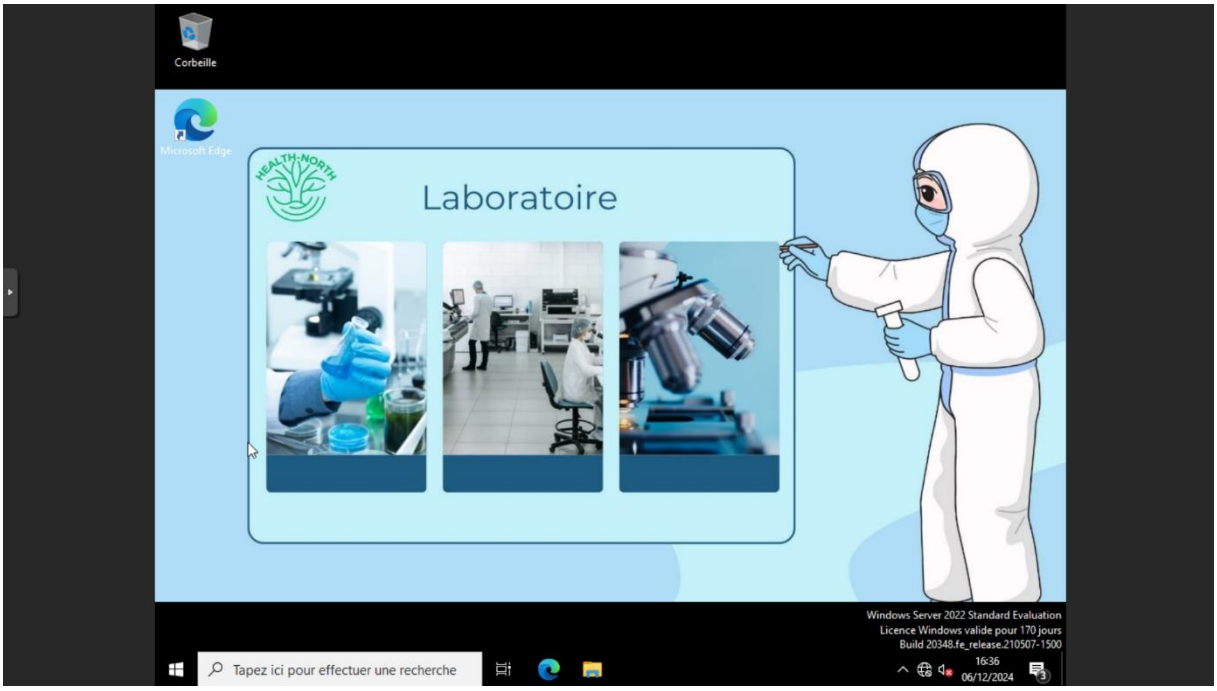
- Le paramètre à activer était donc « Papier peint du Bureau ». Dans chaque GPO, le chemin UNC vers le fichier correspondant a été configuré comme suit :
    - Exemple : \\10.160.0.10\Partage\FondsEcran\Laboratoire.jpg.



5. Tests et validation

- Les GPO ont été appliquées aux unités d’organisation (OU) respectives des services.
- Des tests ont été effectués avec différents comptes utilisateurs pour vérifier que :
    - Les fonds d'écran s’appliquent correctement.
    - Aucune erreur ou restriction d’accès n’apparaît.

Résultats des tests :



Cette configuration permet de personnaliser facilement l’environnement des utilisateurs tout en respectant les bonnes pratiques de sécurité. Le déploiement des fonds d'écran est désormais centralisé et géré via les GPO, rendant la maintenance et les mises à jour rapides et efficaces. Cela peut aussi contribuer à ce que l’environnement de travail des utilisateurs soit plus personnalisé et plus sympa.

Présentation du problème des profils itinérants et de sa résolution

Les utilisateurs utilisant des **profils itinérants** rencontraient des erreurs de synchronisation au moment de la déconnexion. Lors de celle-ci, j'avais un message qui s'affichait qui faisait référence à la synchronisation des profils. J'ai donc été vérifier dans l'observateur d'événements Windows et des erreurs Winlogon étaient visibles (codes 1509 et 1540). Ces erreurs provenaient de conflits liés à la présence de **copies locales** des profils sur les machines clientes.

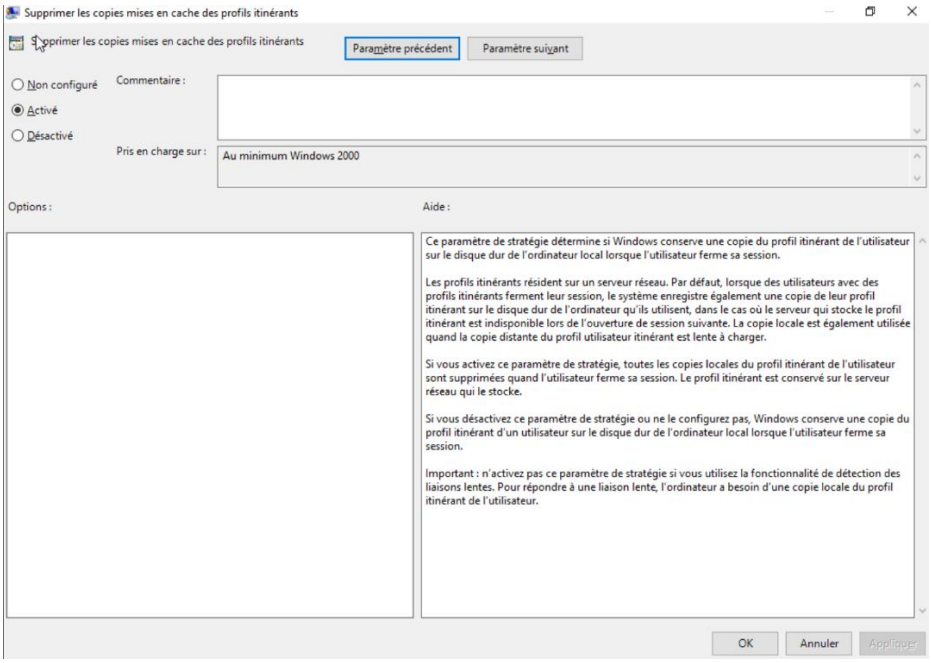
J'ai aussi fait plusieurs vérifications en créant des dossiers sur un profil utilisateur, et en me déconnectant/reconnectant je voyais bien que la synchronisation ne se faisait pas parfaitement.

Conséquences :

- Problèmes de synchronisation entre le serveur et les machines clientes.
- Erreurs répétées dans les journaux d'événements.
- Difficulté à gérer les profils pour plusieurs utilisateurs dans un environnement réseau (car une seule machine pour les utilisateurs).

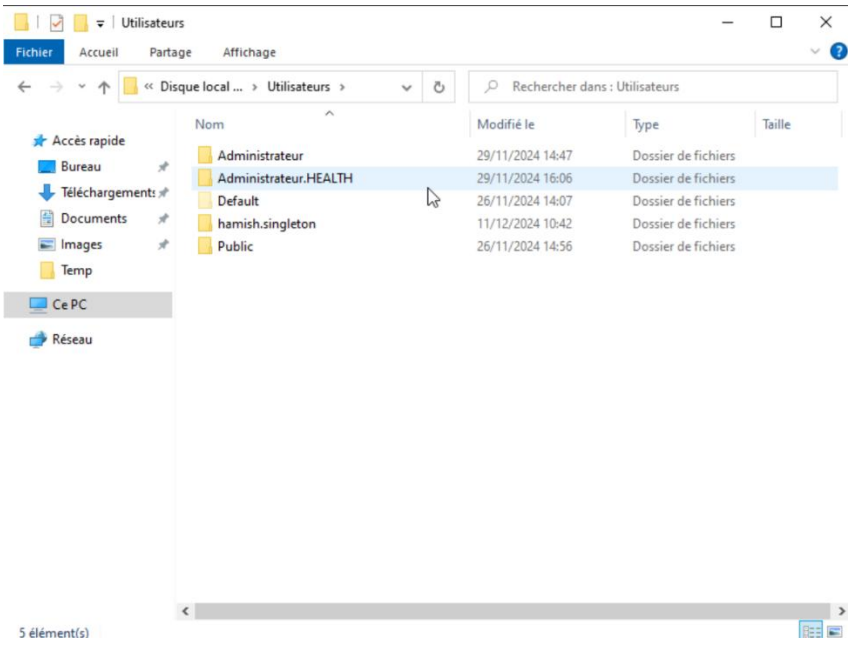
Solution mise en place

Pour résoudre ces problèmes, une stratégie de groupe (GPO) a été configurée pour supprimer automatiquement les copies locales des profils itinérants à chaque déconnexion.

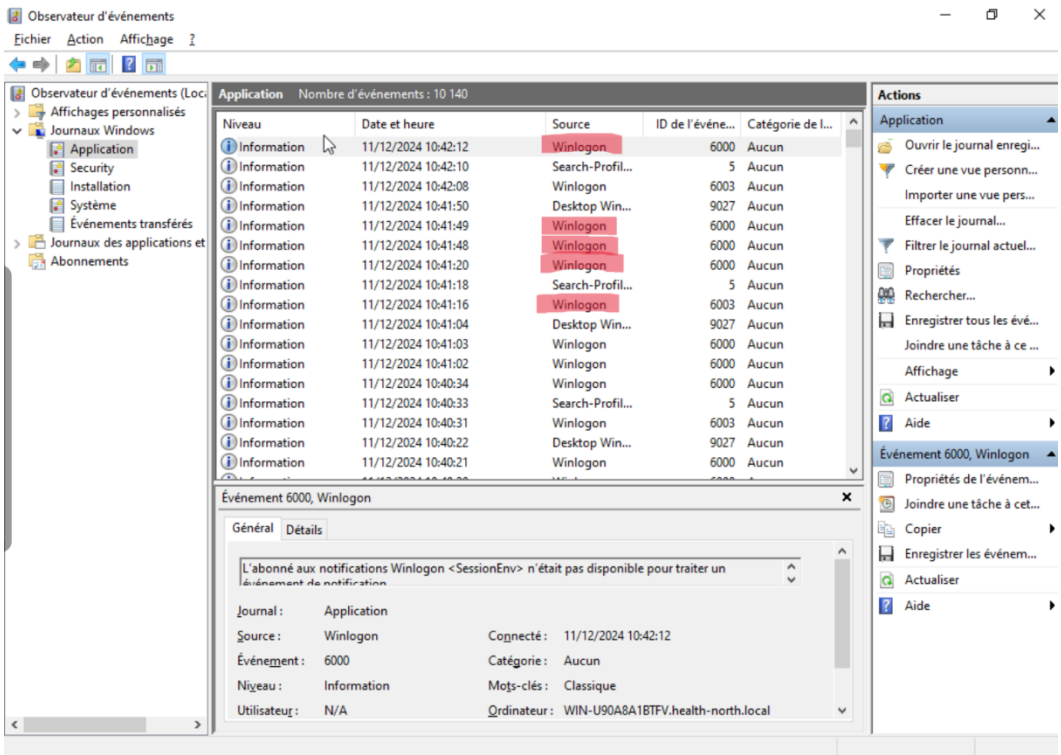


Résultat

- Les copies locales des profils sont supprimées automatiquement lors de chaque déconnexion.



- Les erreurs **Winlogon** (codes 1509 et 1540) ne sont plus présentes dans les journaux d'événements.



- Les profils itinérants fonctionnent correctement, sans conflit, pour tous les utilisateurs.

Problème rencontré : Manque de stockage

Pour mettre un peu de contexte, la création d'une nouvelle machine cliente (Windows 11) pour les tests utilisateurs était nécessaire, car les tests étaient auparavant effectués sur la même VM que le serveur DHCP.

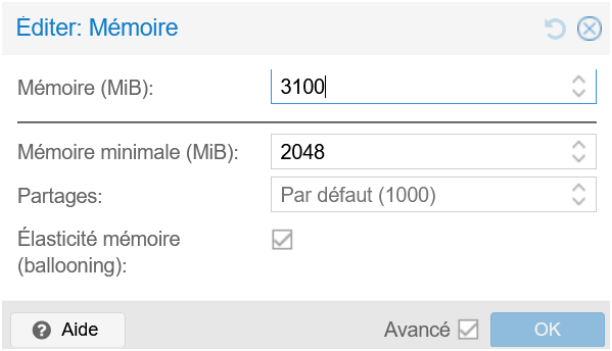
- L’installation de Windows 11 échouait systématiquement, avec des blocages et crashs pendant le processus.
- Après analyse, le problème a été identifié comme un manque de stockage disponible sur la machine physique hébergeant l’ensemble des VMs et donc du serveur proxmox.

Pour résoudre ce problème, j’ai ajouté un SSD NVMe de 256Go sur la machine physique, pour que je puisse avoir l’espace requis pour ajouter une machine cliente.

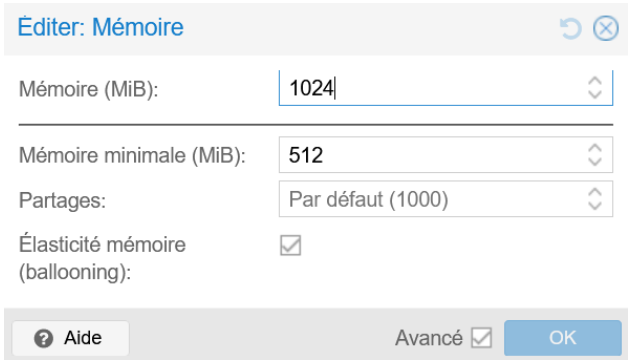
J’en ai profité pour réévaluer les besoins de mon serveur et surtout pour la RAM car c’est un élément que je ne pouvais pas augmenter à cause de problème physique de la machine que j’utilisais. Je voulais que les 4 VMs puissent tourner en même temps sans que cela ne cause trop de ralentissement sur une VM ou sur une autre et sur le serveur proxmox en général aussi.

J’ai donc réalisé ces ajustements :

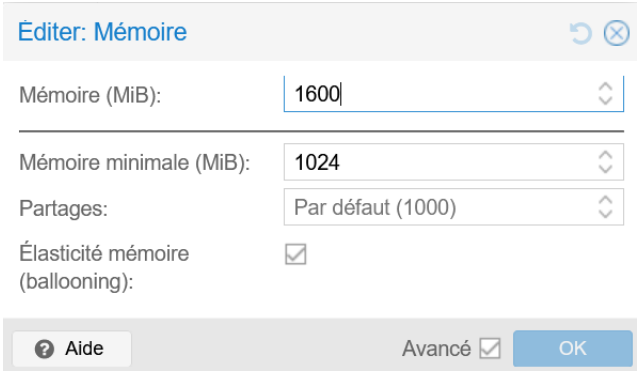
- 3,1Go pour la VM hébergeant le serveur AD/DNS, en utilisant l’allocation dynamique de la RAM, le ballooning, que j’ai fixé à 2Go minimum :



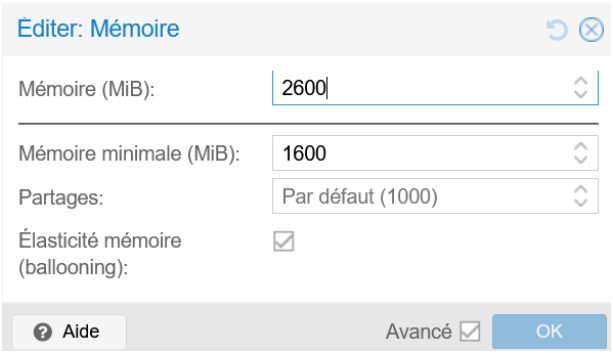
- 1Go pour la VM hébergeant le serveur DHCP, en utilisant le ballooning, que j'ai fixé à 512Mo minimum :



- 1,6Go pour la VM hébergeant le serveur de partage, en utilisant le ballooning, que j'ai fixé à 1Go minimum :

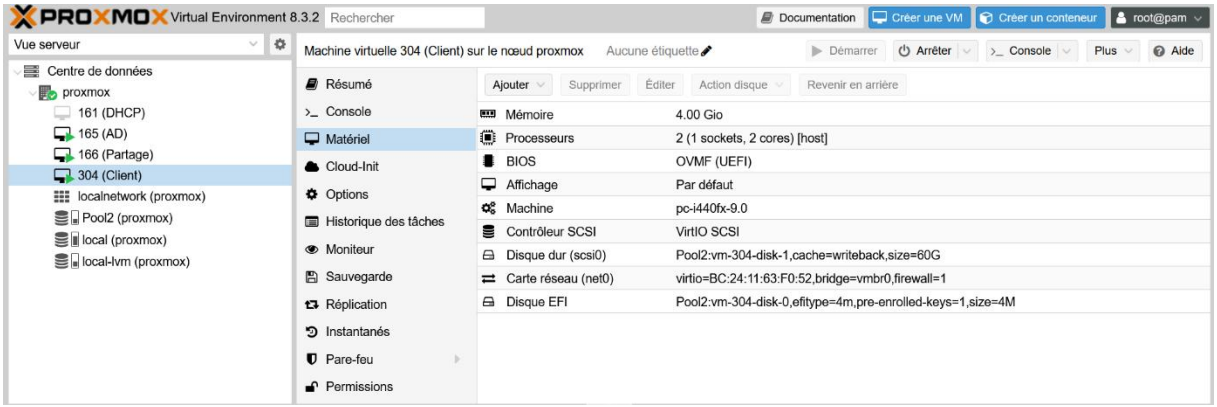


- 2,6Go pour la VM hébergeant un système Windows 11 pour les utilisateurs, en utilisant le ballooning, que j'ai fixé à 1,6Go minimum :



Création d'une nouvelle VM cliente :

J'ai créé cette VM pour qu'elle soit dédiée exclusivement aux profils utilisateurs. Elle est sous Windows 11 et a cette configuration matérielle :



Déploiement des applications

Tentative initiale avec les GPO

La première méthode envisagée pour déployer les applications était d'utiliser les GPO. L'objectif était de configurer un déploiement automatique des applications sur les machines clientes via des stratégies de groupe.

Malgré une configuration correcte des chemins réseau pour les packages d'installation, les GPO ne parvenaient pas à appliquer le déploiement des applications.

Problèmes rencontrés :

- Les GPO ne parvenaient pas à appliquer correctement le déploiement des applications, même après vérification des chemins réseau et des configurations.
- Dans l'Observateur d'événements, des erreurs étaient enregistrées, signalant des échecs liés à l'installation des packages logiciels.
- Ces erreurs incluaient notamment des problèmes comme :
- Échec de traitement des packages MSI et des exe.

Les nombreuses tentatives et les erreurs répétées ont conduit à l'abandon de cette méthode.

Tentative avec WAPT

Une solution alternative, **WAPT**, a été envisagée pour gérer le déploiement centralisé des applications, c'est un logiciel que j'avais déjà utilisé lors de mes stages donc c'était assez familier pour moi et je connaissais globalement comment cela fonctionnait.

WAPT est donc un outil utilisé pour installer, mettre à jour et gérer les logiciels sur des machines clientes depuis un serveur dédié.

Problèmes rencontrés :

Problèmes d'installation du serveur WAPT :

- Une première tentative échouait lorsque j'essayais de résoudre un problème lié au certificat auto-signé.
- Suite à plusieurs manipulations infructueuses, j'ai décidé de repartir de 0, et j'ai donc supprimé tout ce qui était lié à WAPT.
- Lors de la seconde tentative, un problème est venu au début de l'installation sur la mise en place du serveur WAPT, des erreurs liées à la configuration de **NGINX** et des certificats HTTPS ont été rencontrées.
- Le serveur retournait régulièrement des erreurs telles que **"Bad Gateway (502)"**, empêchant l'accès à la console d'administration.

Finalement, WAPT a été abandonné, de nombreuses manipulations infructueuses aussi, une seconde tentative où ça bloquait encore plus tôt dans le processus ont eu raison de moi.

Les deux premières solutions (GPO et WAPT) ont été jugées incompatibles pour répondre aux besoins spécifiques de mon projet. Ces limitations ont conduit à envisager une approche différente pour le déploiement des applications.

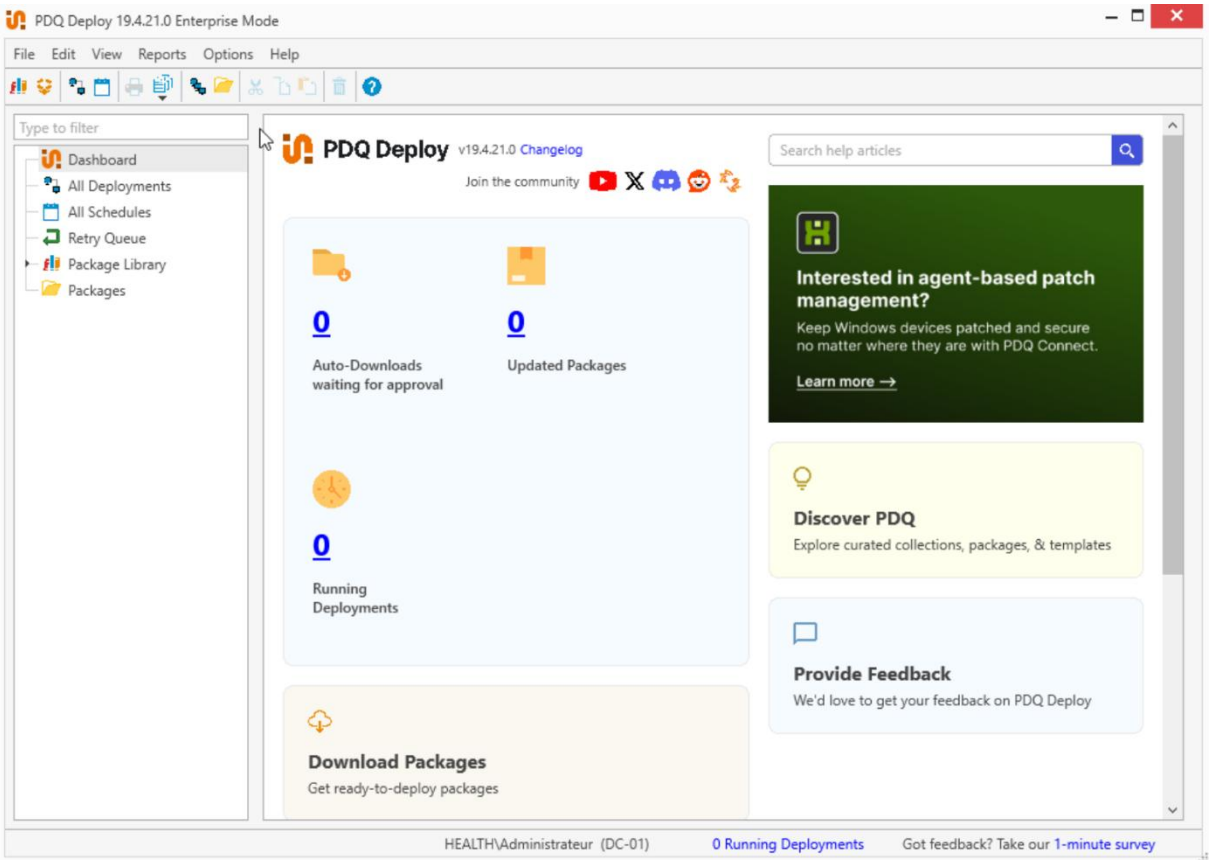
Déploiement des applications avec PDQ Deploy

Après les échecs avec les GPO et WAPT, PDQ Deploy a été choisi comme solution alternative pour gérer le déploiement des applications. Cet outil permet un déploiement centralisé et silencieux des logiciels sur les machines clientes.

Étapes de mise en place

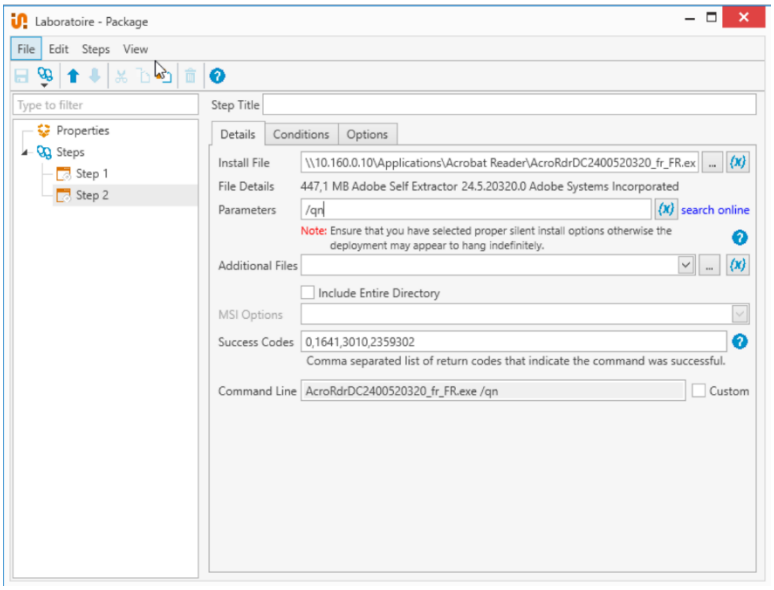
Téléchargement et installation de PDQ Deploy :

J'ai donc téléchargé le logiciel **PDQ Deploy** depuis le site officiel, en choisissant l'édition gratuite. J'ai installé l'outil en mode local sur mon serveur AD (une option serveur/centralisé était aussi proposé). La configuration initiale est très rapide et simple.



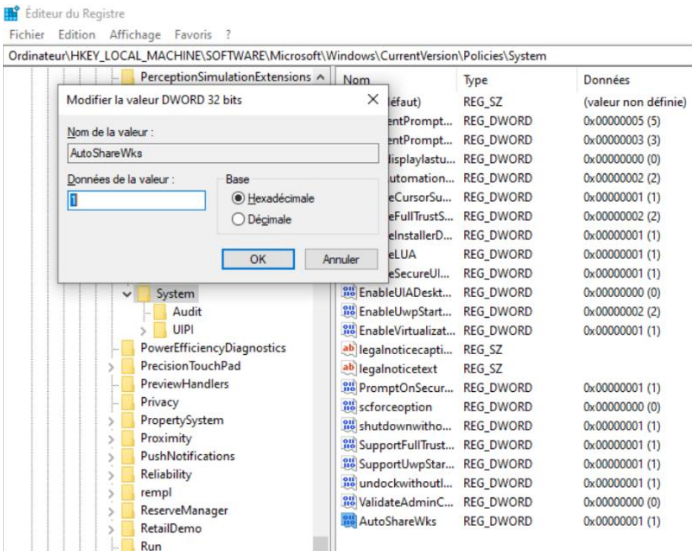
Configuration réseau et accès :

Pour commencer j'ai créé mon premier package (un ensemble de logiciel à déployer) , qui contenait VLC et Acrobat Reader.



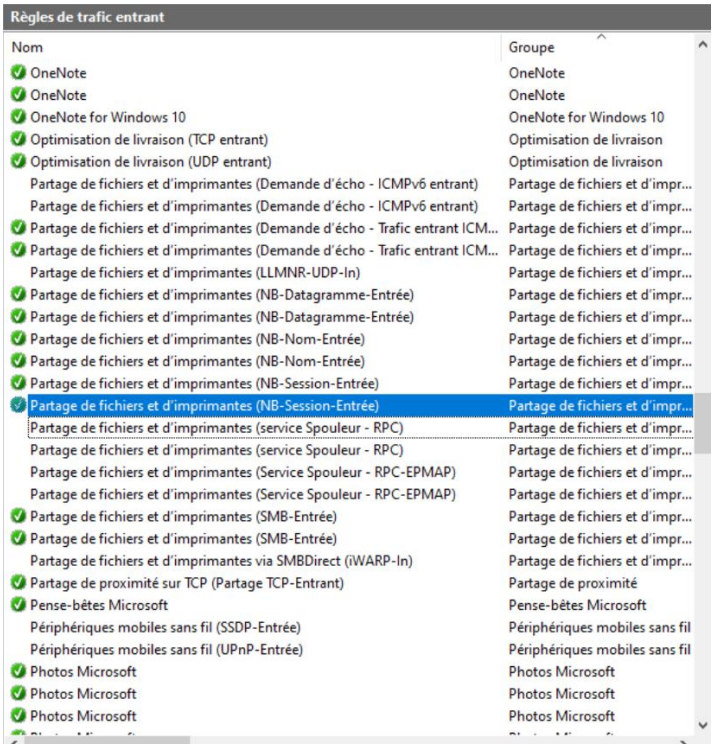
Après quelques tentatives échouées, j'ai réalisé qu'une erreur autre que la configuration du package se produisait.

J'ai trouvé que PDQ Deploy utilisait les partages administratifs pour les installations. J'ai donc vérifié et essayer d'aller dans le dossier C$ de la machine cliente et je n'y avais pas accès ! J'ai ajouté la clé de registre AutoShareWks avec la valeur 1 pour activer les partages administratifs.



Le problème persistait, j'ai donc réaliser de simple test de connectivité entre les deux machines à l'aide de la commande ping. La machine cliente arrivait à joindre la machine où se trouvait l'AD, mais pas le contraire.

J'ai alors été voir si une règle de pare-feu pouvait correspondre à mon problème. En cherchant l'info sur internet, j'ai activé plusieurs règles de pare-feu concernant le partage et les entrées ICMP et SMB.

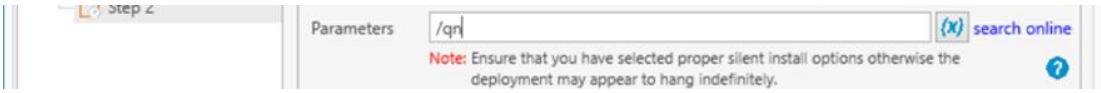


Il y avait au total 6 applications requises, et chaque service avait ses propres applications selon une matrice qui m'avait été donnée lors de l'attribution du sujet.

Service	VLC	7zip	Acrobat Reader	Firefox	Putty	Winscp
Laboratoire	X		X			
Chirurgie	X					
Informatique	X	X	X	X	X	X
Direction	X		X			

Le premier package était destiné au service laboratoire et devait contenir les applications VLC et Acrobat Reader.

Il était possible d'attribuer des paramètres à l'installation pour par exemple spécifié la langue, ou encore spécifié à ce logiciel en particulier de faire l'installation en silence, pour n'avoir aucune interaction avec l'utilisateur.



Problèmes rencontrés et solutions

VLC

Problèmes rencontrés :

- Le déploiement de VLC échouait systématiquement avec un **code erreur 2** dans PDQ Deploy.
- Malgré l'absence d'erreur réseau ou d'accès, l'installation silencieuse ne fonctionnait pas.
- Lorsque la commande /S (installation silencieuse) était désactivée, l'installation manuelle était possible.

Solutions apportées :

- Le fichier d'installation VLC a été re-téléchargé directement depuis le site officiel, car celui initialement utilisé semblait corrompu ou incompatible.
- Une nouvelle configuration a été appliquée avec la commande /S pour permettre une installation silencieuse.
- Après cette modification, VLC s'est correctement installé sur la machine cliente.

Acrobat Reader

Aucun problème majeur pendant le déploiement. L'installation fonctionnait correctement, même en mode silencieux.

7-Zip

Aucun problème majeur pendant le déploiement. L'installation fonctionnait correctement, même en mode silencieux.

Mozilla Firefox

1. **Problèmes rencontrés :**

- Initialement, l'installation silencieuse fonctionnait, mais les tests suivants échouaient :
    - L'installation prenait un temps infini.
    - PDQ Deploy renvoyait un **code erreur 1**.
- Des résidus de fichiers Firefox, laissés sur la machine cliente (dans AppData et Program Files), empêchaient une nouvelle installation.

2. **Solutions apportées :**

- Suppression manuelle de tous les fichiers et dossiers liés à Firefox, notamment dans :
    - C:\Program Files\Mozilla Firefox
    - C:\Users\[Utilisateur]\AppData\Roaming\Mozilla
- Après nettoyage complet, l'installation silencieuse (avec /S) fonctionnait à nouveau sans problème.

PuTTY

1. **Problèmes rencontrés :**

- L'installation via l'exécutable .exe échouait systématiquement.
- La commande /s ne fonctionnait pas avec la version .exe.

2. **Solutions apportées :**

- Abandon de la version .exe pour utiliser un package .msi.
- Après avoir utilisé le fichier .msi, l'installation s'est déroulée correctement, avec un déploiement silencieux réussi.

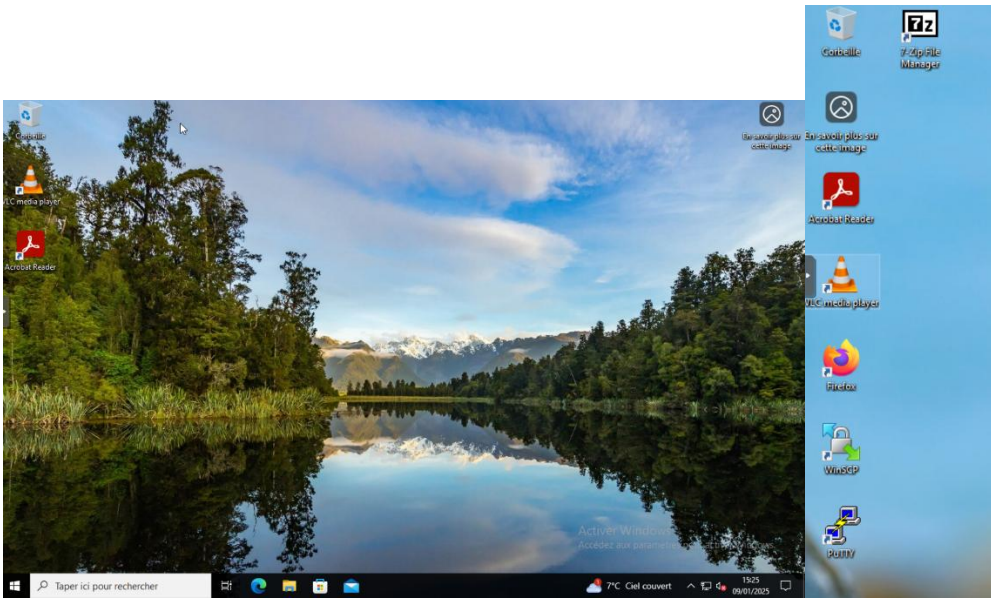
WinSCP

1. **Problèmes rencontrés :**

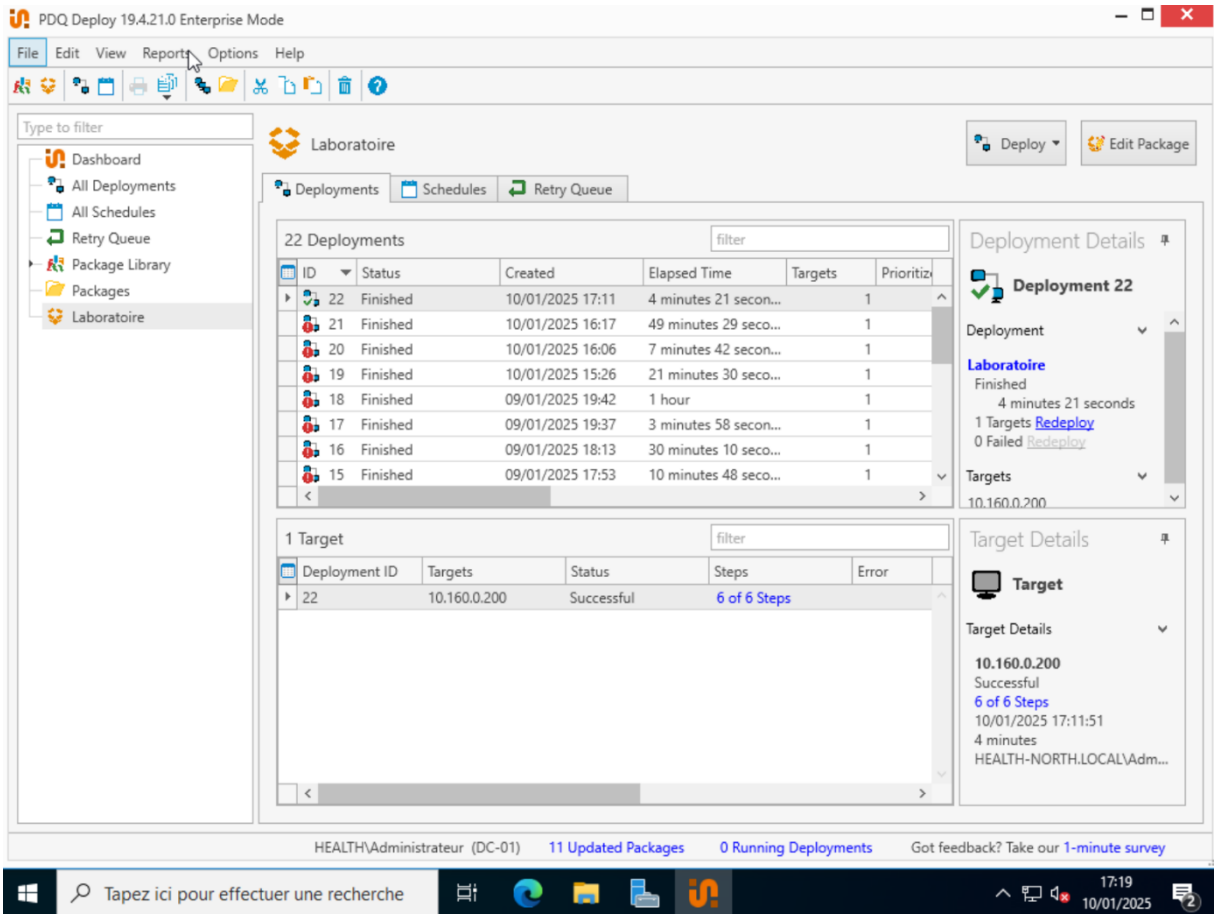
- L'installation via l'exécutable .exe échouait systématiquement.
- Les commandes /silent et /verysilent ne fonctionnaient pas avec la version .exe.

2. **Solutions apportées :**

- Abandon de la version .exe pour utiliser un package .msi.
- Après avoir utilisé le fichier .msi, l'installation s'est déroulée correctement, avec un déploiement silencieux réussi.

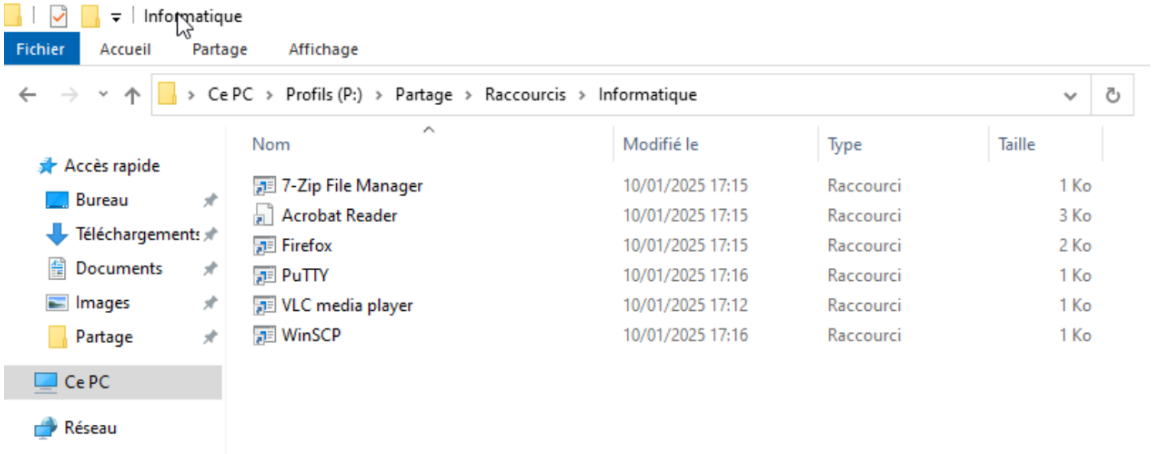


Cela a nécessité beaucoup de tentatives d'installations, mais finalement le déploiement de toutes les applications à fonctionner !



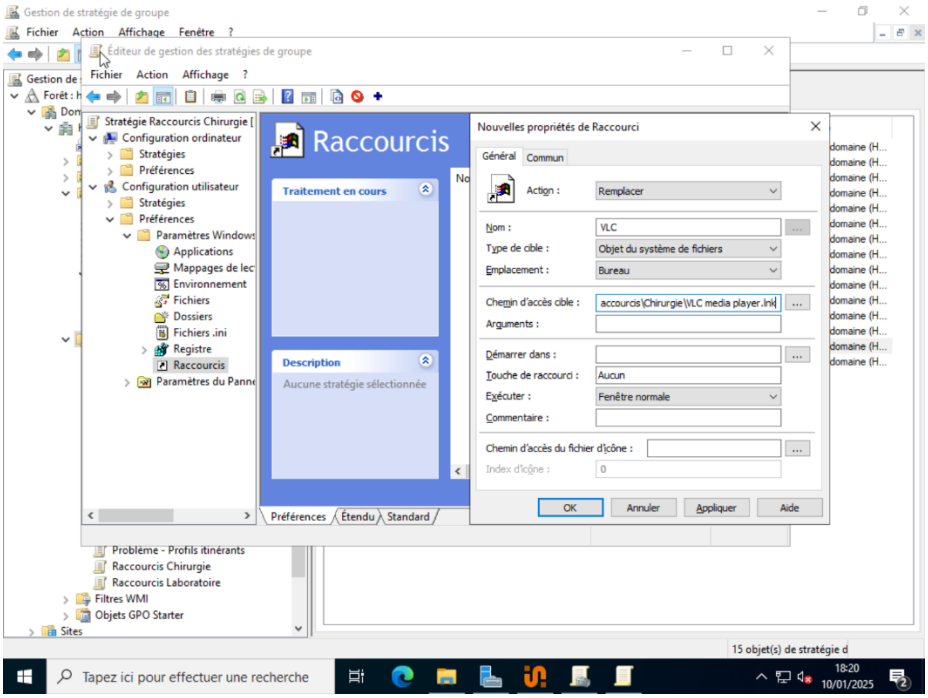
Personnalisation par utilisateur

Chaque utilisateur est attribué à un service et chaque service à accès à certaines applications bien spécifique (voir matrice plus haut). J'ai donc créé un dossier raccourci sur le serveur de partage en vue d'attribuer par GPO les raccourcis spécifiques à chaque service. Dans ce dossier « Raccourcis » se trouve donc 4 dossiers correspondant aux 4 services. Dans chacun de ses sous-dossiers, il y a les raccourcis appropriés à chaque service.

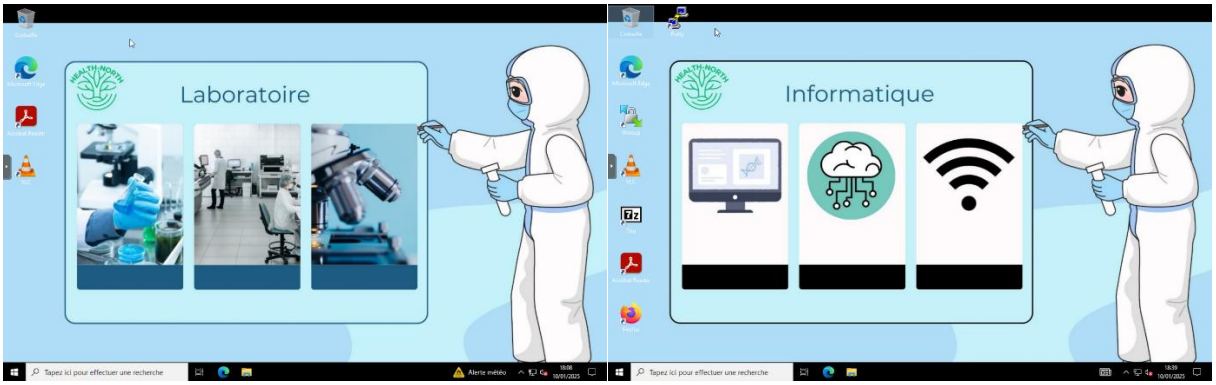


Déploiement des raccourcis via GPO :

Les GPO ont permis de copier automatiquement les raccourcis appropriés sur les bureaux des utilisateurs.



Puis j'ai vérifié chaque utilisateur de chaque service pour m'assurer que les raccourcis appropriés étaient attribués au bon utilisateur.



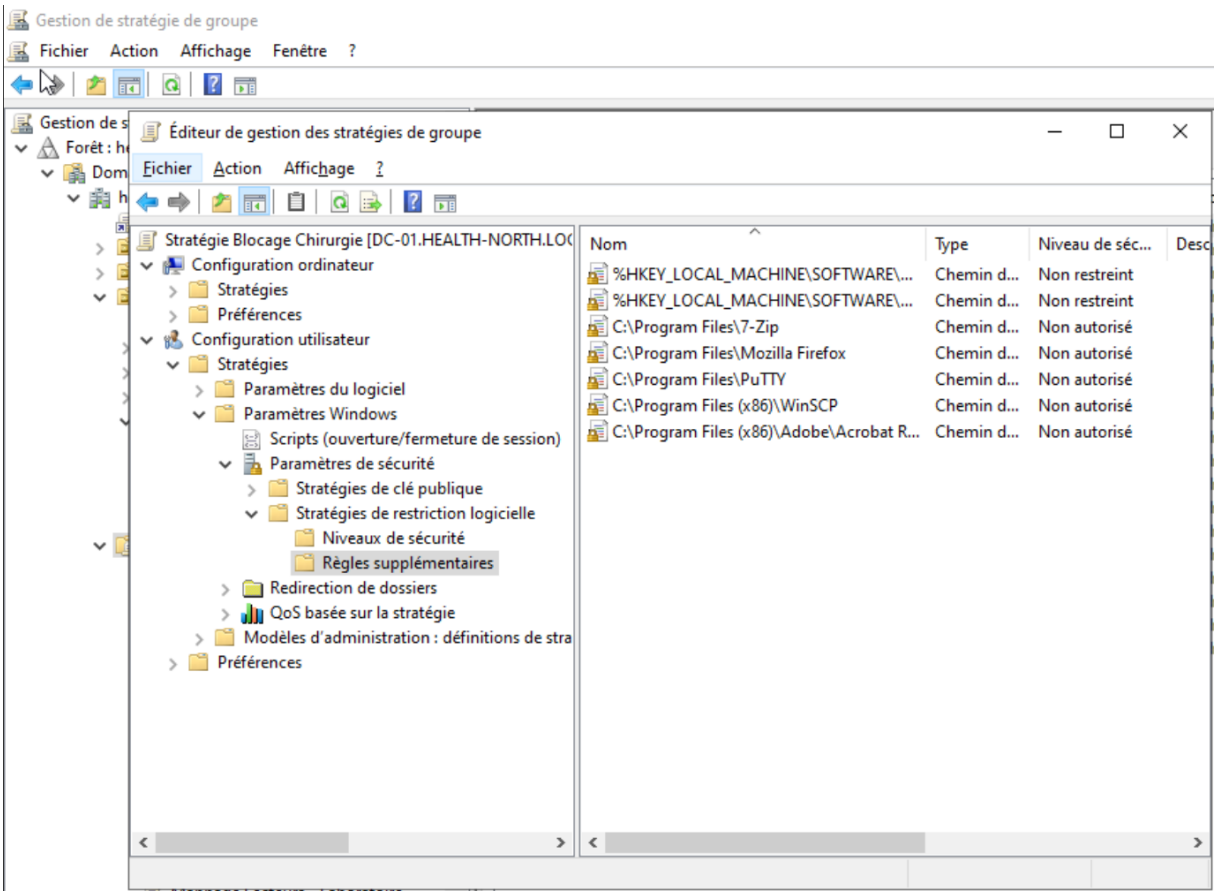
Blocage des applications par utilisateur

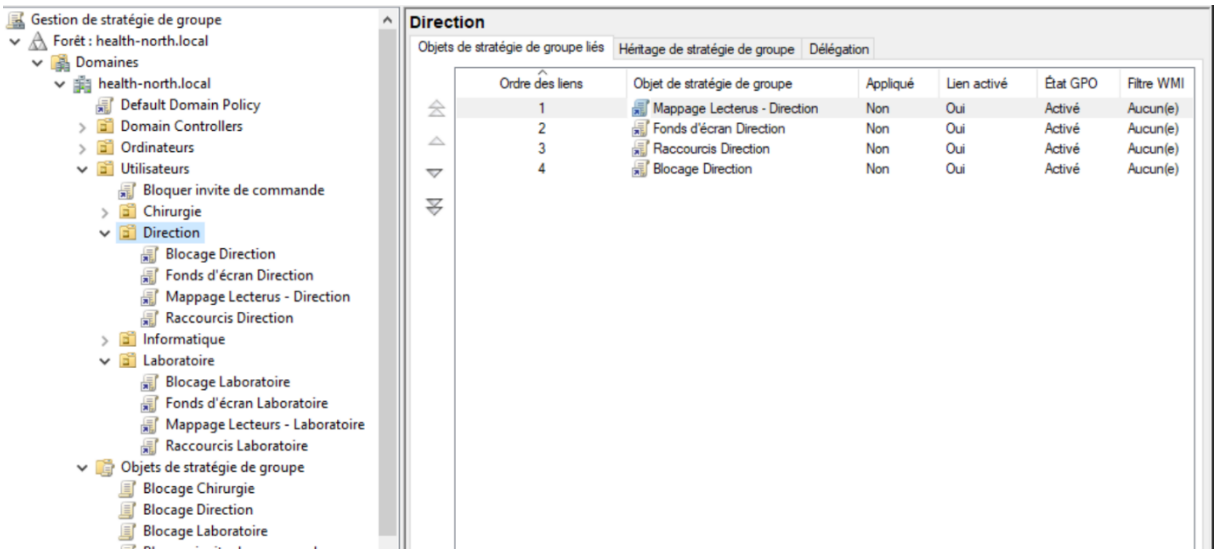
Pour aller au bout de mon idée, j'ai décidé d'associer aux raccourcis par service, le blocage des applications. Comme je n'ai qu'une VM à ma disposition qui sert de seul machine pour tout les utilisateurs, le déploiement des applications s'est fait sur cette seule machine. Malgré les raccourcis mis en place en fonction du service auquel est rattaché chaque utilisateur cela ne suffisait pas. En condition réelle, si un utilisateur cherche sur son ordinateur une application à laquelle il n'a normalement pas accès, elle s'affichera quand même puisque toutes les applications sont installées sur la machine et il peut éventuellement alors s'en service.

Pour pallier à ce problème , j'ai opté pour le blocage des applications par groupe qui correspond à un service, par GPO.

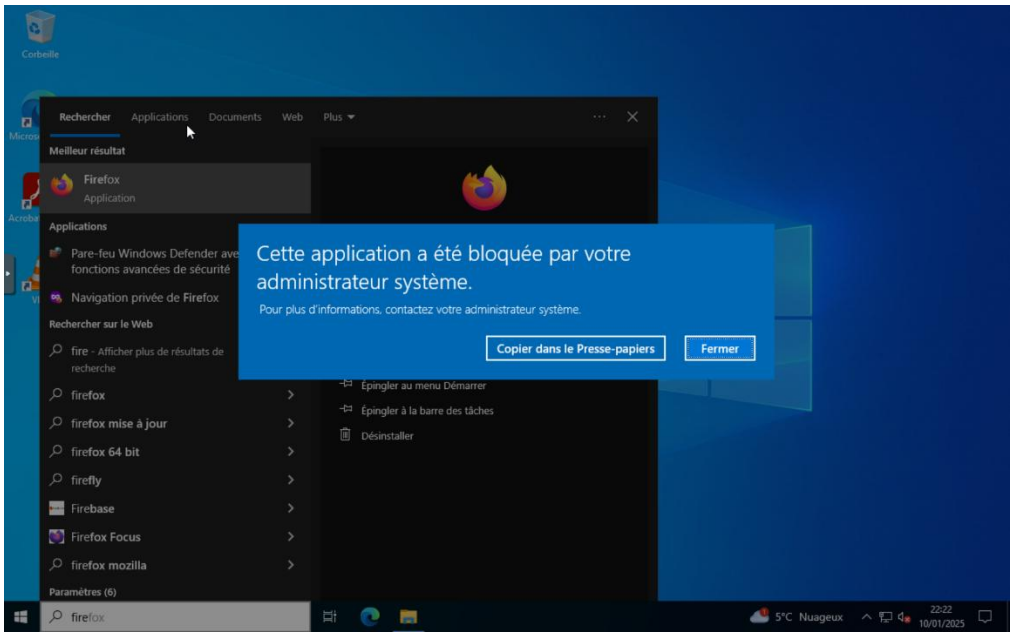
Mise en place d'une GPO :

- Les exécutables des applications non autorisées pour un service ont été bloqués via des règles de restriction logicielle. Il fallait simplement spécifier le chemin de l'application en question. J'ai alors opté de spécifier le dossier entier de l'application à chaque fois.





J’ai ensuite testé chaque application sur chaque utilisateur de chaque service pour vérifier que tout fonctionnait bien. Les utilisateurs pouvaient donc chercher, et même trouver les applications auxquelles ils n’avaient normalement pas accès, mais lorsqu’ils essayaient d’exécuter celles-ci, cela leur affichait ce message :



Résultat final

- Les applications ont été déployées avec succès pour tous les services, et les raccourcis sont personnalisés selon les besoins.
- Les utilisateurs ne peuvent accéder qu’aux applications autorisées pour leur service.
- PDQ Deploy a permis de surmonter les limitations rencontrées avec les GPO et WAPT.

Installation et mise en place de GLPI

Introduction et contexte : Dans le cadre de mon projet **BTS SIO SISR**, j'ai déployé et configuré **GLPI (Gestion Libre de Parc Informatique)**, un outil open-source permettant la **gestion des actifs IT**, le **suivi des incidents** à l'aide de tickets, ainsi que la **supervision des équipements du réseau**.

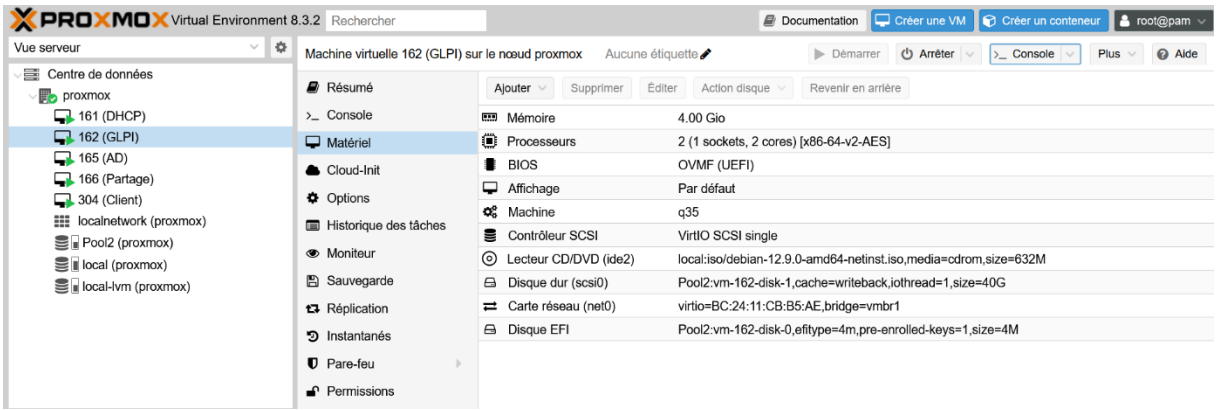
L'objectif principal de cette mise en place était de centraliser la **gestion du parc informatique** de l'infrastructure que j'ai mise en place, tout en intégrant **Active Directory** pour assurer une authentification simplifiée des utilisateurs. En parallèle, le déploiement de l'**agent GLPI** sur les machines du réseau permet d'inventorier automatiquement le matériel et les logiciels installés, facilitant ainsi la supervision et l'administration.

Ce projet s'inscrit dans une démarche visant à **optimiser la gestion des incidents**, tout en offrant une meilleure **traçabilité des équipements et des interventions**. Il constitue un élément essentiel pour assurer le **bon fonctionnement et la maintenance de l'infrastructure réseau** mise en place.

L'installation de **GLPI** s'est déroulée en plusieurs étapes, allant de la mise en place de l'environnement serveur jusqu'à l'intégration avec **Active Directory** et le déploiement des **agents GLPI** sur les postes clients. L'objectif était de rendre la plateforme pleinement fonctionnelle pour la gestion des tickets et l'inventaire du parc informatique.

1. Préparation du serveur GLPI

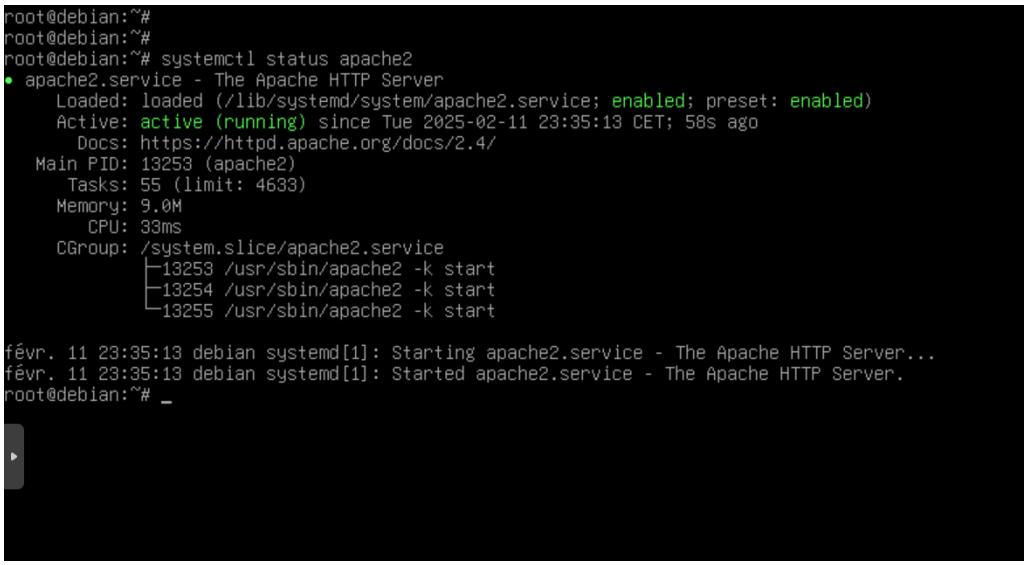
Le serveur GLPI a été installé sur une **machine virtuelle Debian 12** avec ces spécifications :



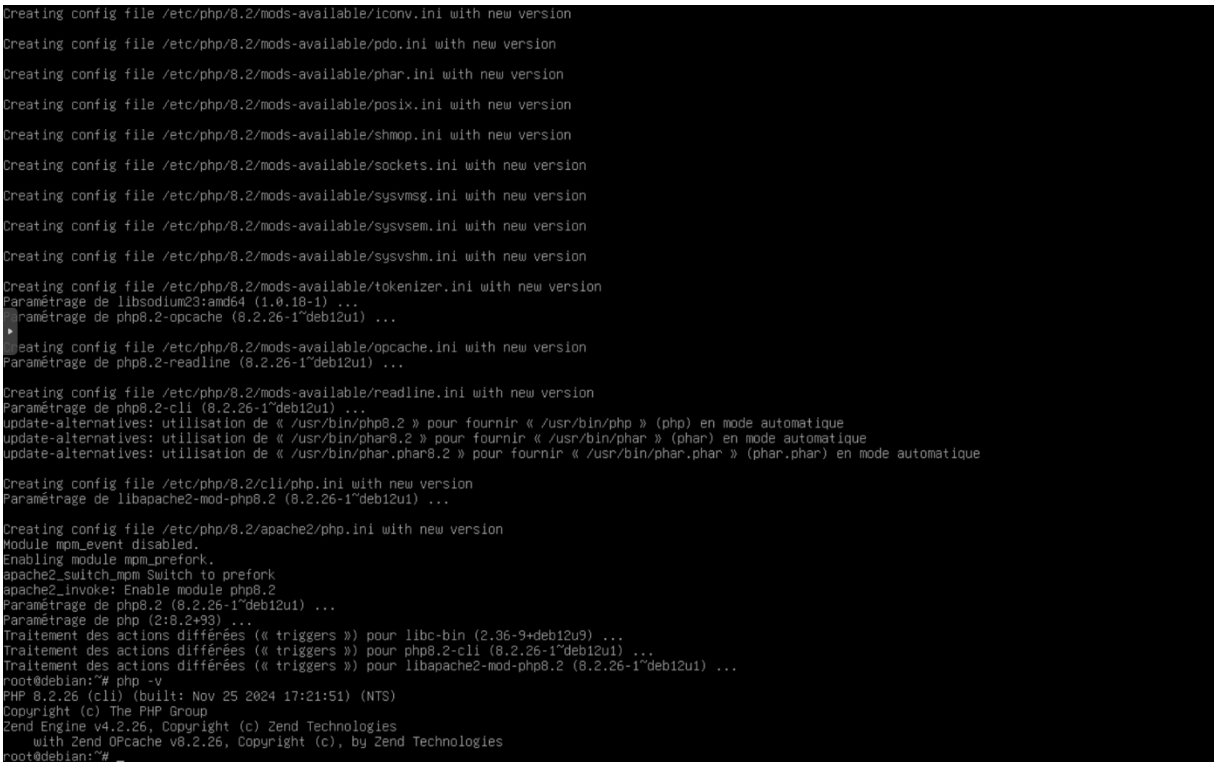
Pour permettre l'exécution de GLPI, il était nécessaire d'installer une **pile LAMP** comprenant :

- **Apache 2** (serveur web)
- **MariaDB** (base de données)
- **PHP 8.2** avec plusieurs modules requis par GLPI

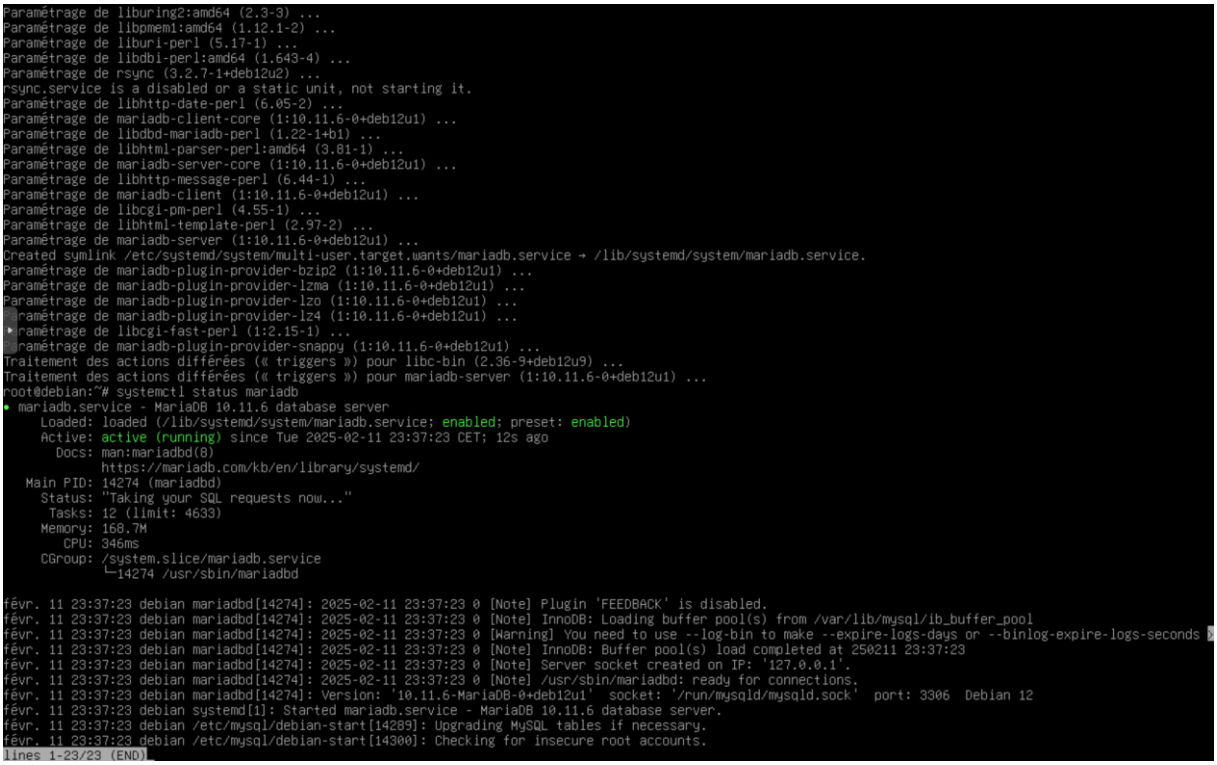
J’ai commencé par l’installation de cette pile avec en premier l’installation d’Apache2 qui aura le rôle de serveur web :



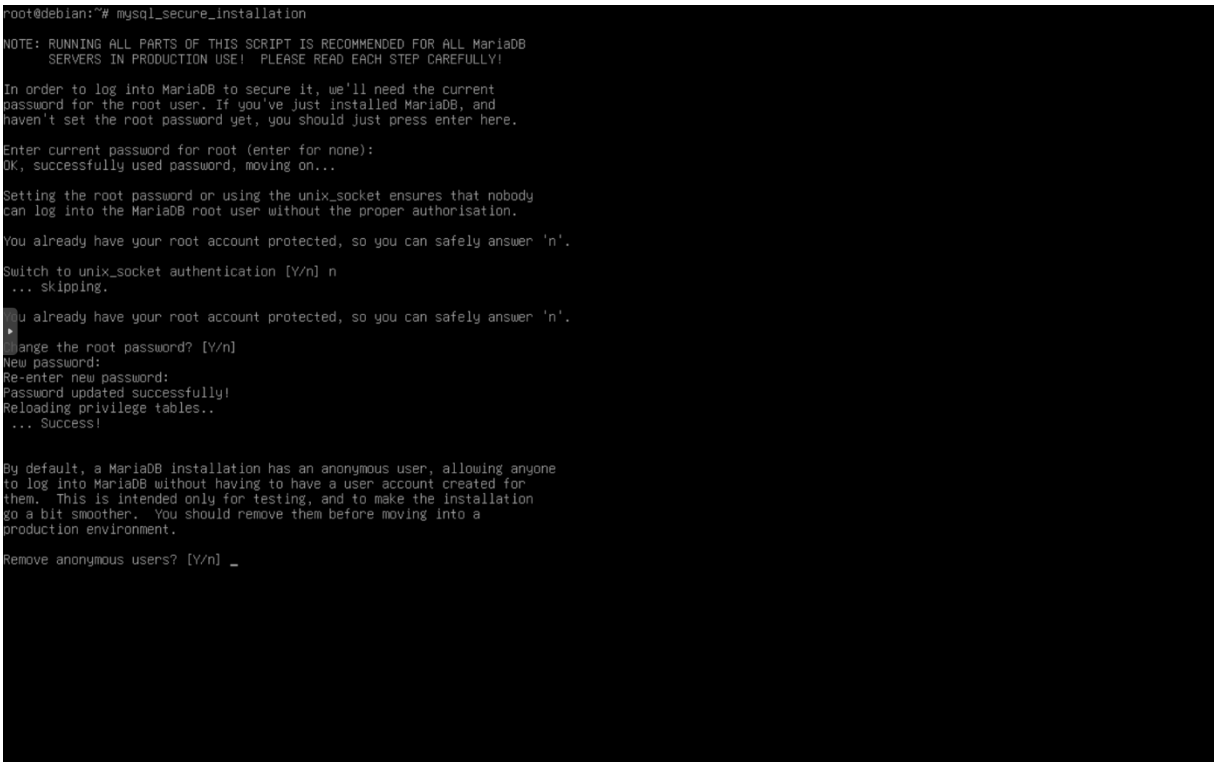
Ensuite j'ai effectué l'installation de php :



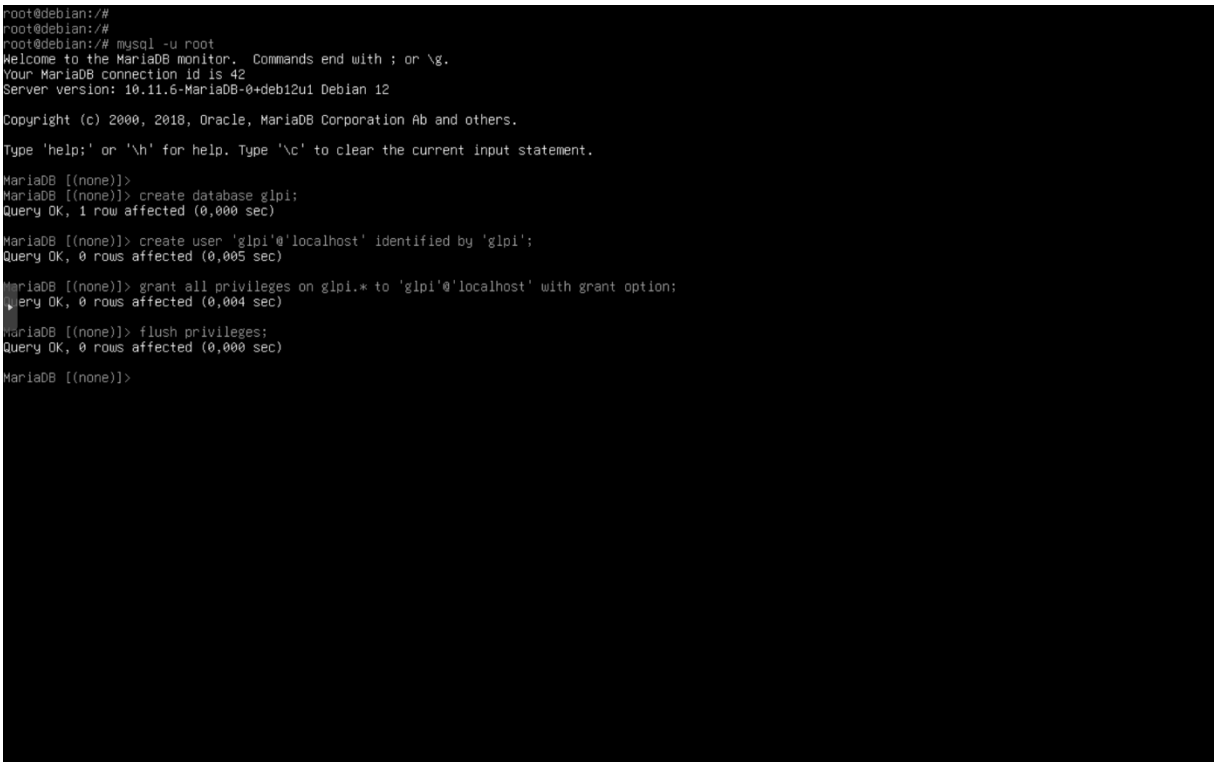
Puis celle de mariadb qui aura le rôle de base de données :



J’ai ensuite sécurisé la base de données mariadb :

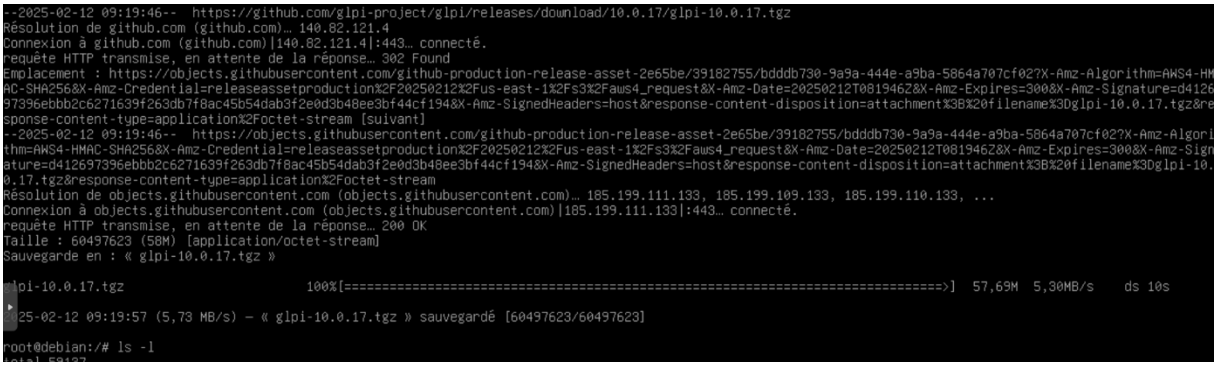


Je me suis ensuite connecté à la base de données mariadb pour créer la base de données de GLPI :



J’ai créé un utilisateur à qui j’ai augmenter les droits, il aura le rôle de faire la liaison entre GLPI et le serveur pour exécuter les requêtes, il interagira avec la base de données.

J’ai aussi télécharger GLPI depuis le dépôt officiel :

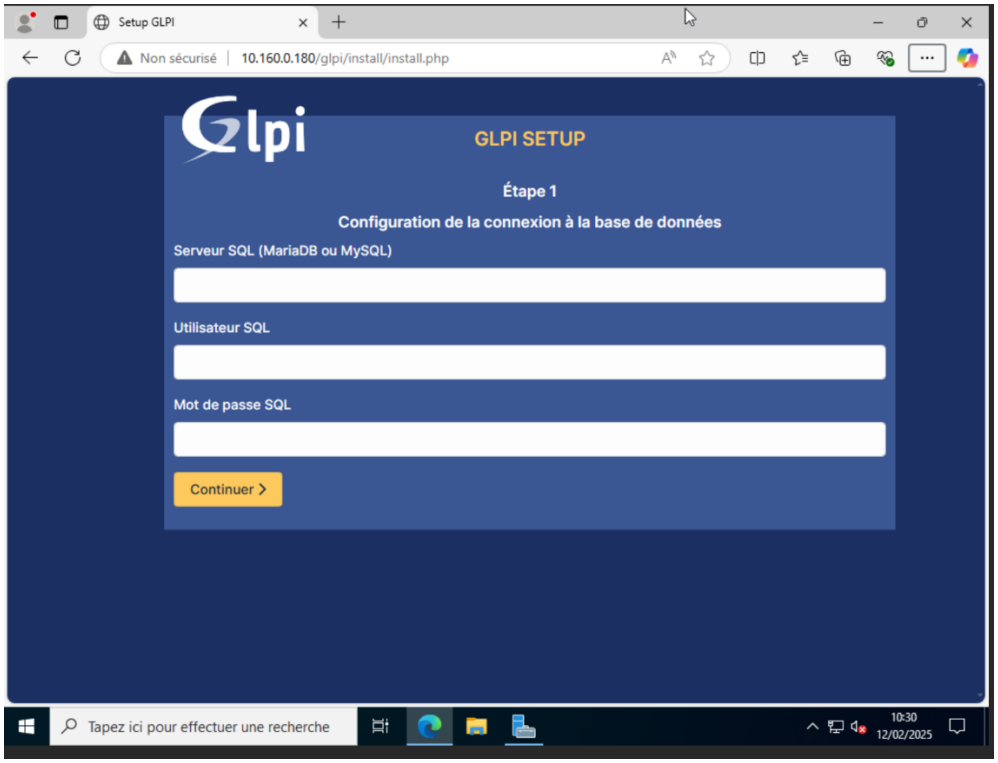


Avant le déploiement de GLPI, il a fallu décompresser l’archive dans l’arborescence d’Apache à un endroit spécifique (/var/www/html).

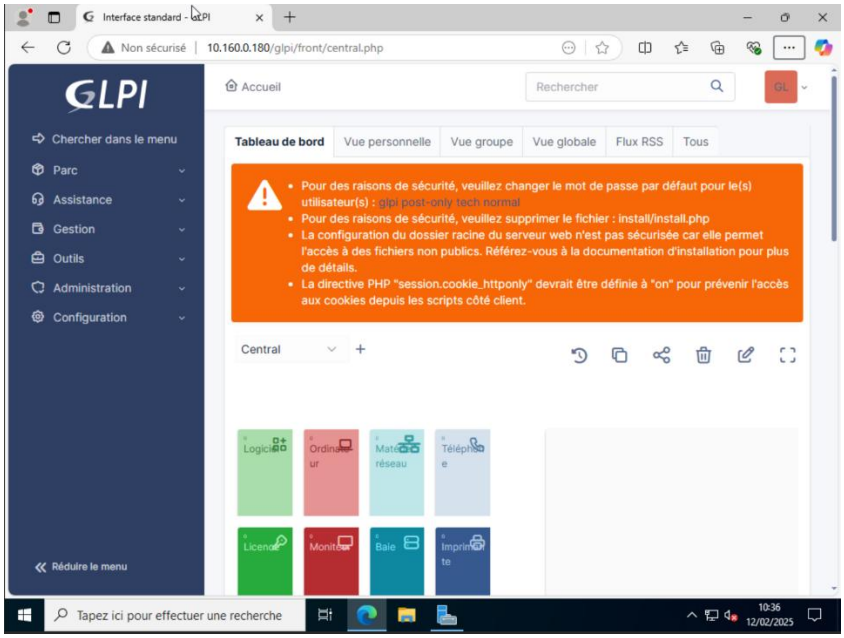
Pour finir, il faut dans les prérequis de la documentation officielle de GLPI, différents modules PHP qui vont servir à plusieurs choses comme lié le LDAP au serveur GLPI etc …

2. Déploiement et configuration de GLPI

Une fois les prérequis en place, j’ai lancé un navigateur web et je me suis rendu sur l’interface de GLPI à l’aide de l’adresse IP du serveur, puis j’ai effectué les premières configuration :



Une fois les premières configurations effectuées, je suis arrivé sur l'interface finale de GLPI :

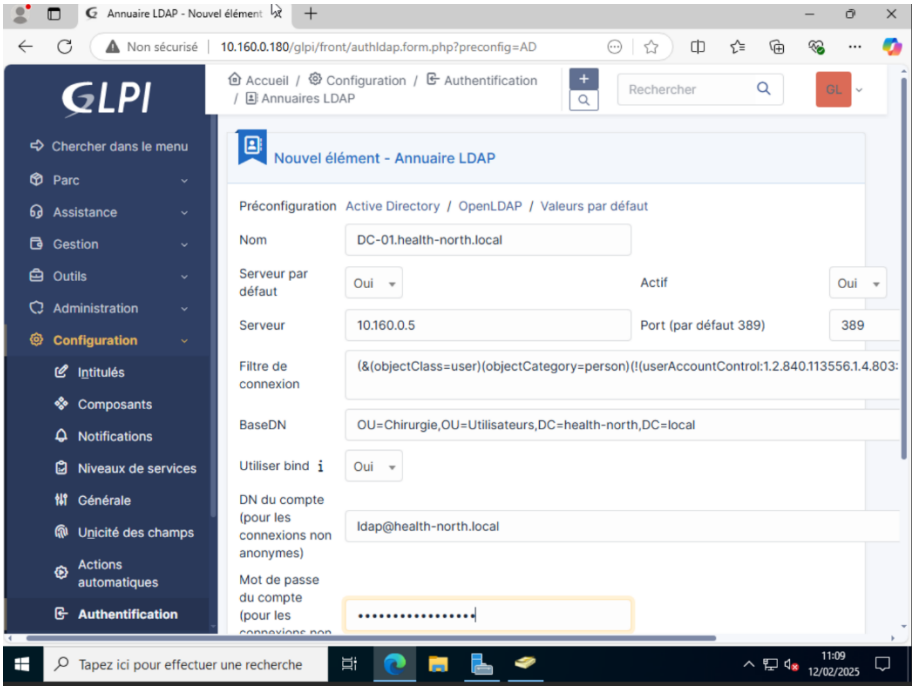


3. Intégration de GLPI avec Active Directory (LDAP)

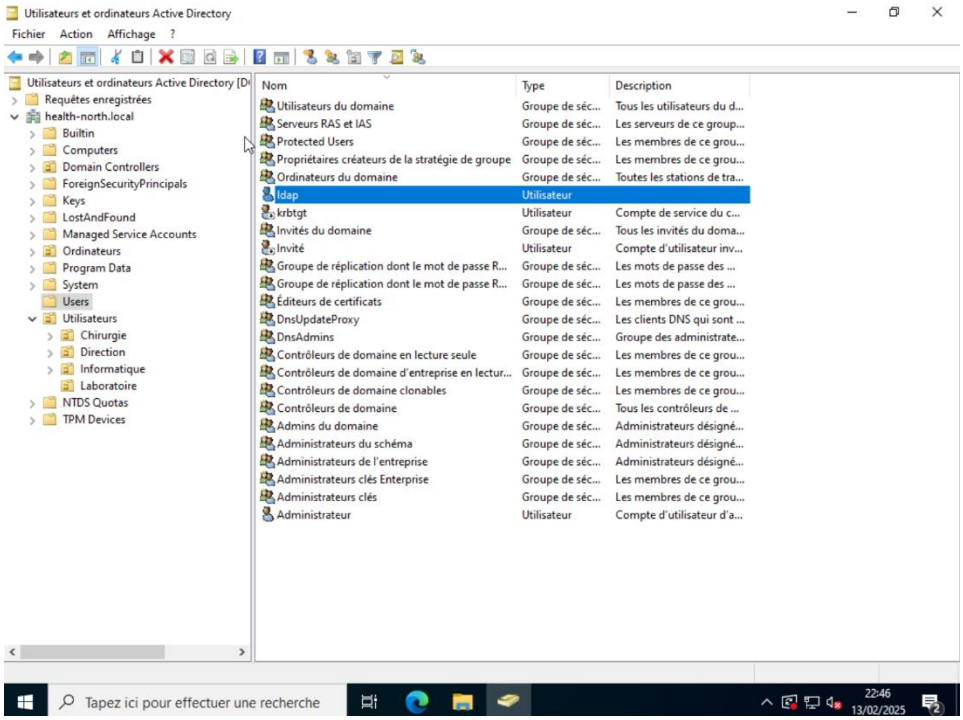
L'un des enjeux majeurs était de permettre aux utilisateurs du domaine de se connecter avec leurs **identifiants Active Directory**.

Pour cela, j'ai :

- Configuré **LDAP** dans GLPI en renseignant le nom du serveur AD (DC-01.health-north.local)
- Déterminé la structure des OU (Unités Organisationnelles) et le **Base DN**

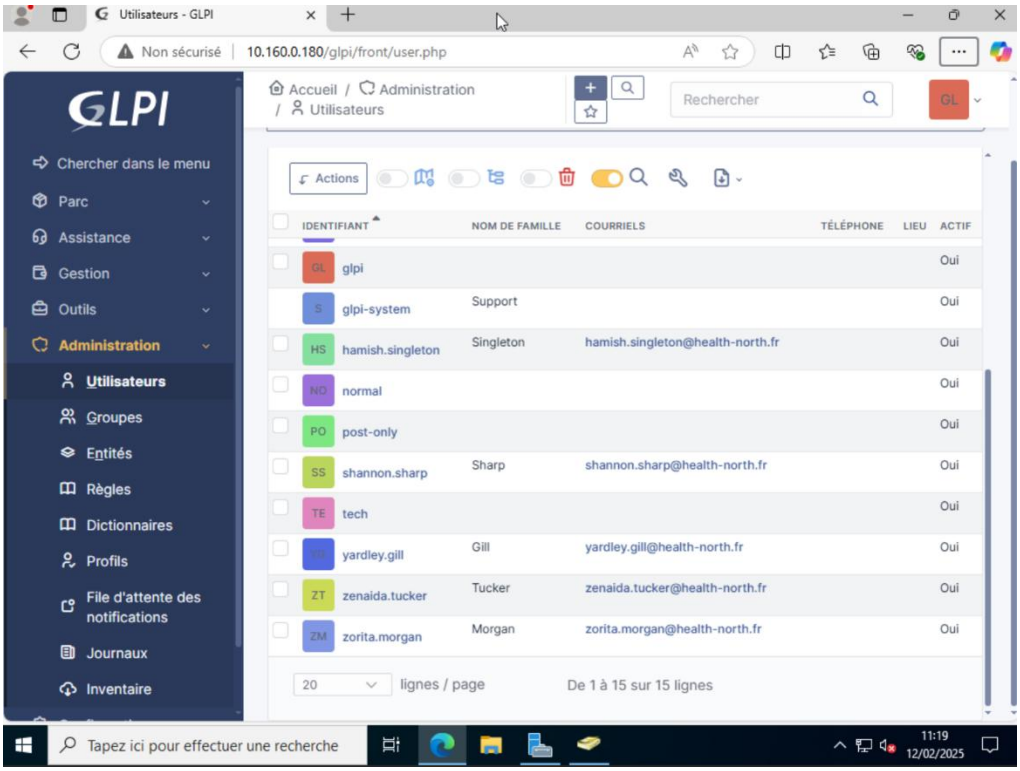


- Créé un **utilisateur dédié LDAP** dans l'AD pour gérer les synchronisations



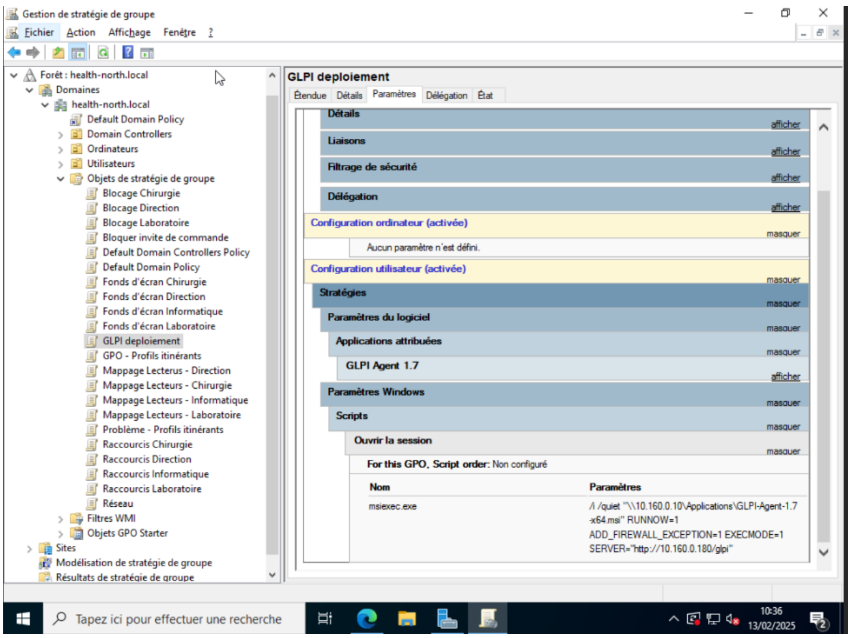
J'ai ensuite fais un test LDAP pour voir si GLPI arrivait à joindre l'AD avec les identifiants de l'utilisateur que j'avais créé exprès :

J’ai importé tous les utilisateurs du domaine :

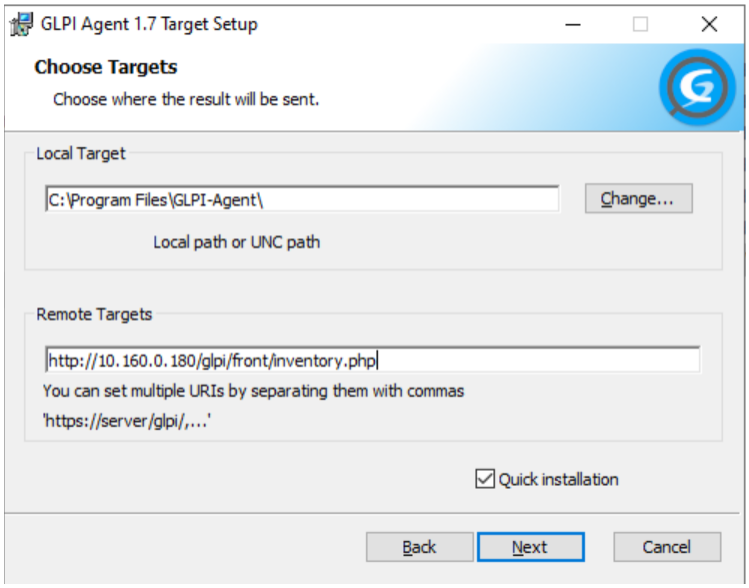


4. Déploiement de l’agent GLPI sur les machines clientes

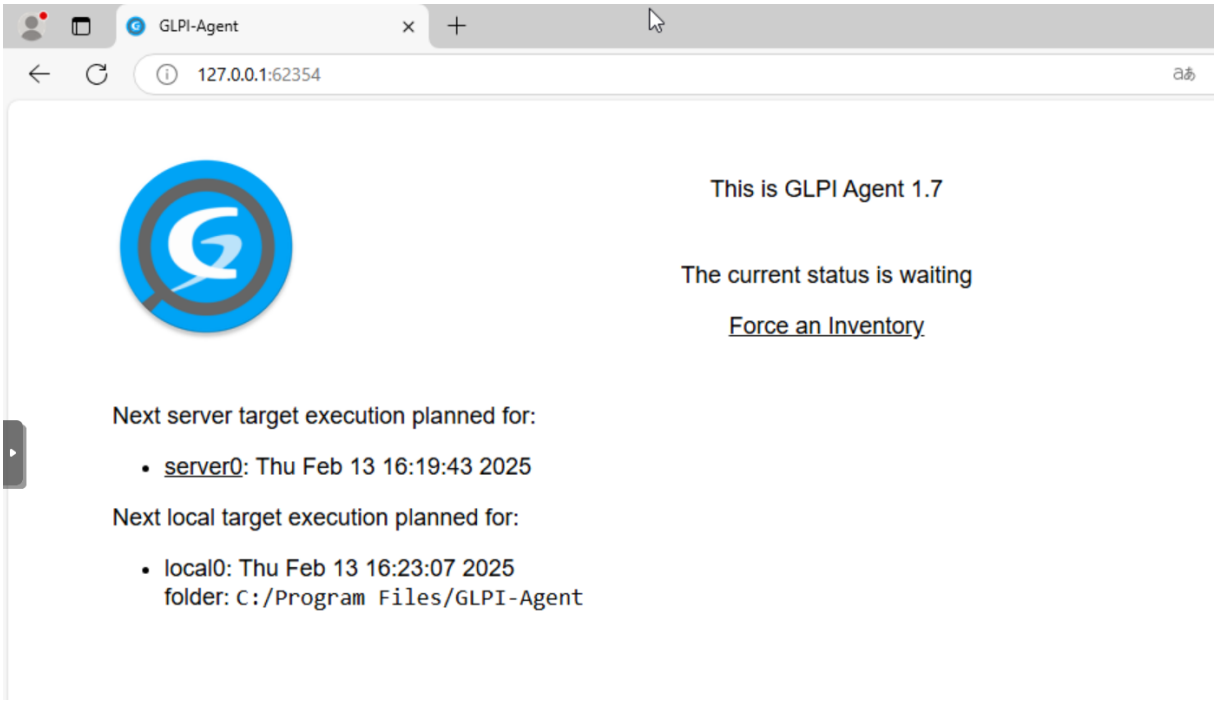
Pour assurer la remontée des informations matérielles et logicielles, j’ai tenté un **déploiement par GPO** de l’agent **GLPI 1.7** en mode **Configuration Utilisateur**. Cependant, cette méthode n’a pas fonctionné, résultant par un échec de l’installation via la stratégie de groupe.



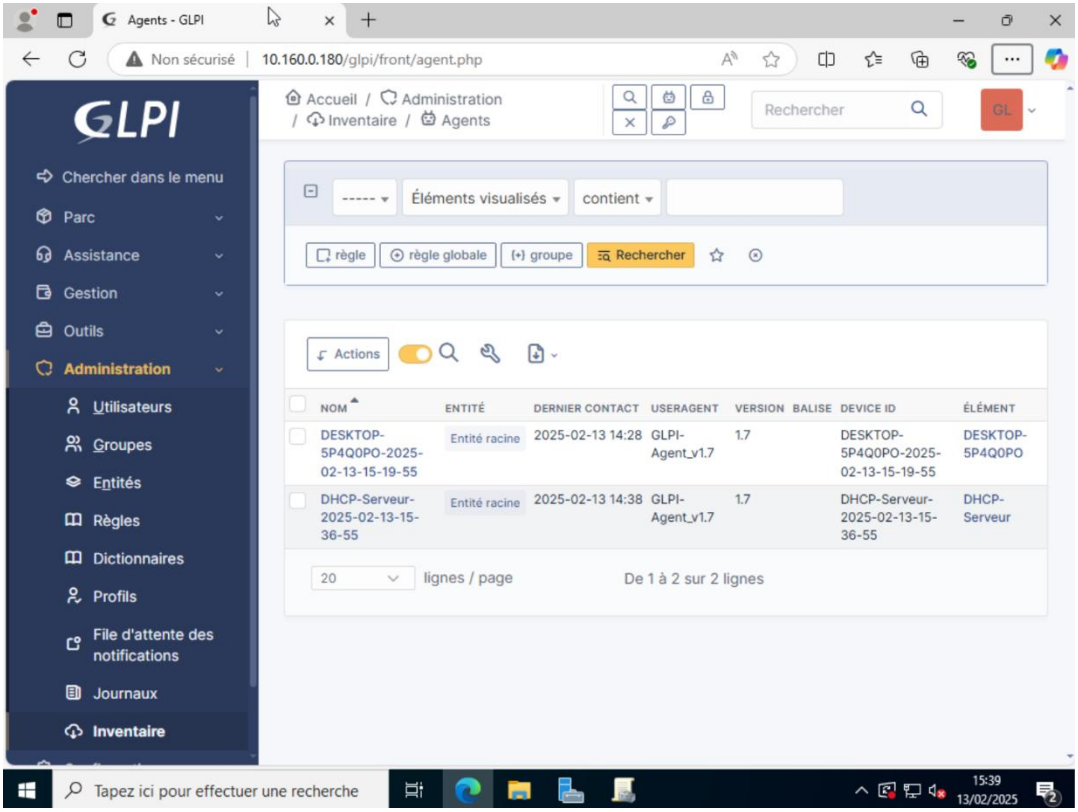
En conséquence, j'ai opté pour une **installation manuelle** de l'agent sur chaque machine :



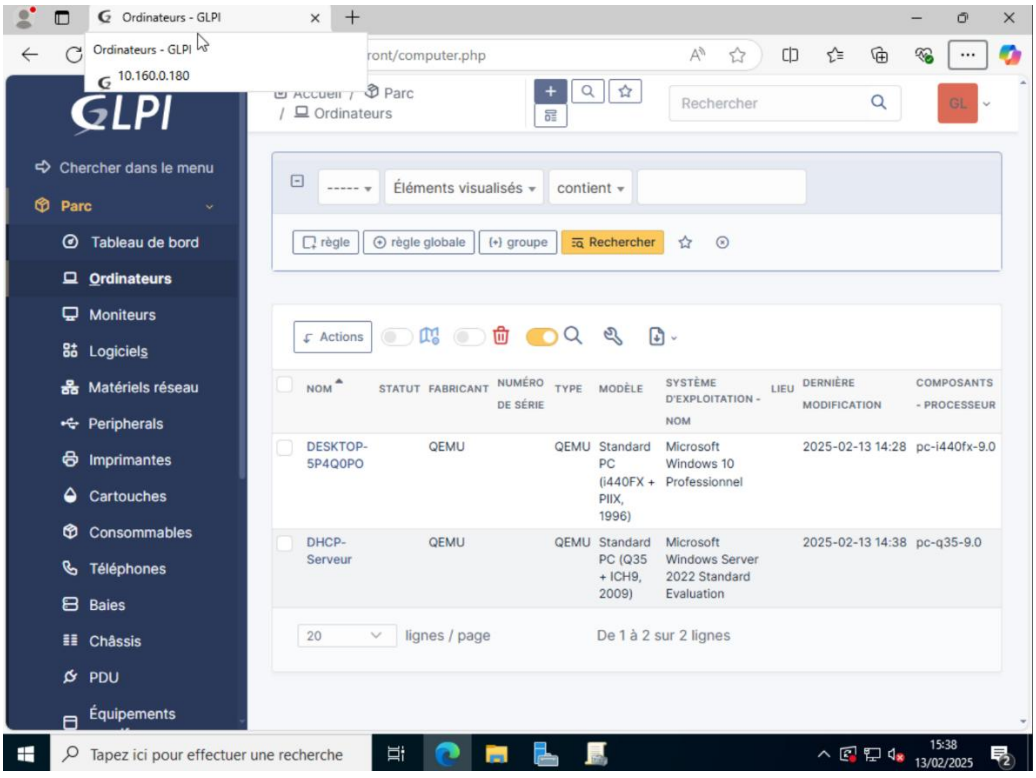
Une fois installé, je me suis rendu sur un navigateur web et j'ai taper l'adresse ip du localhost pour accéder à l'interface de l'agent GLPI pour ensuite pouvoir forcer un inventaire :



On peut par la suite voir la remontée des agents dans l'interface web de GLPI :



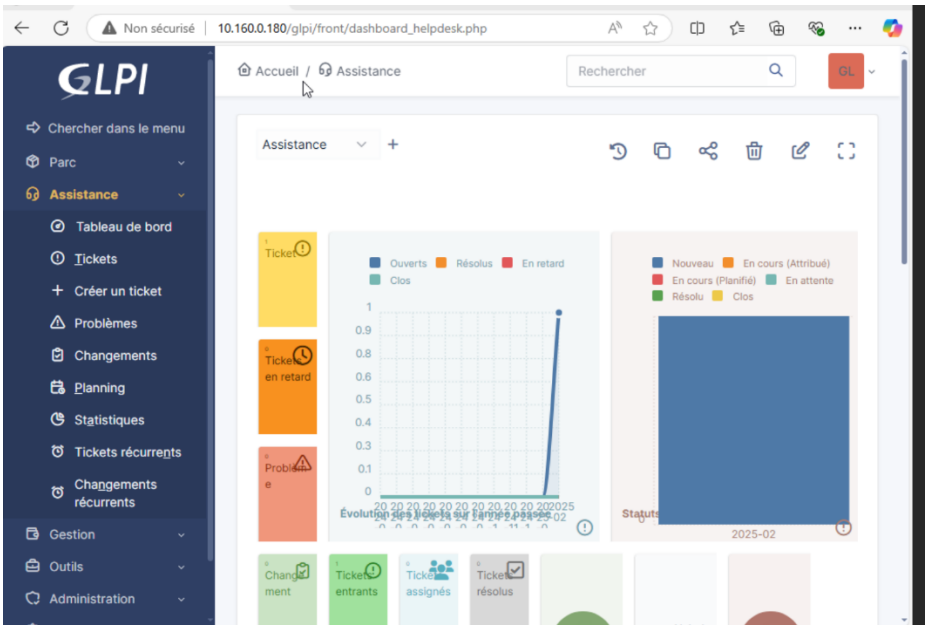
Ainsi que la remontée des différents ordinateurs et donc serveurs :

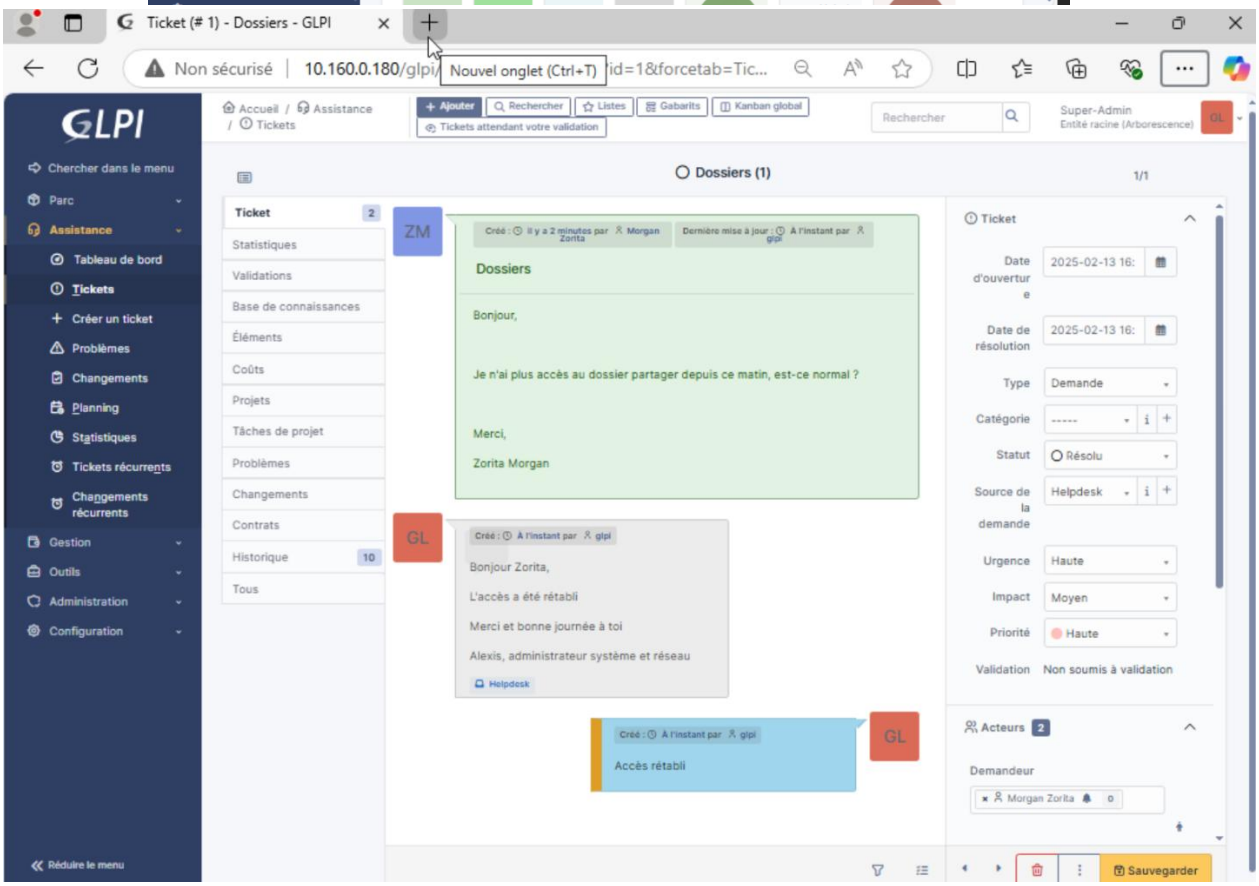


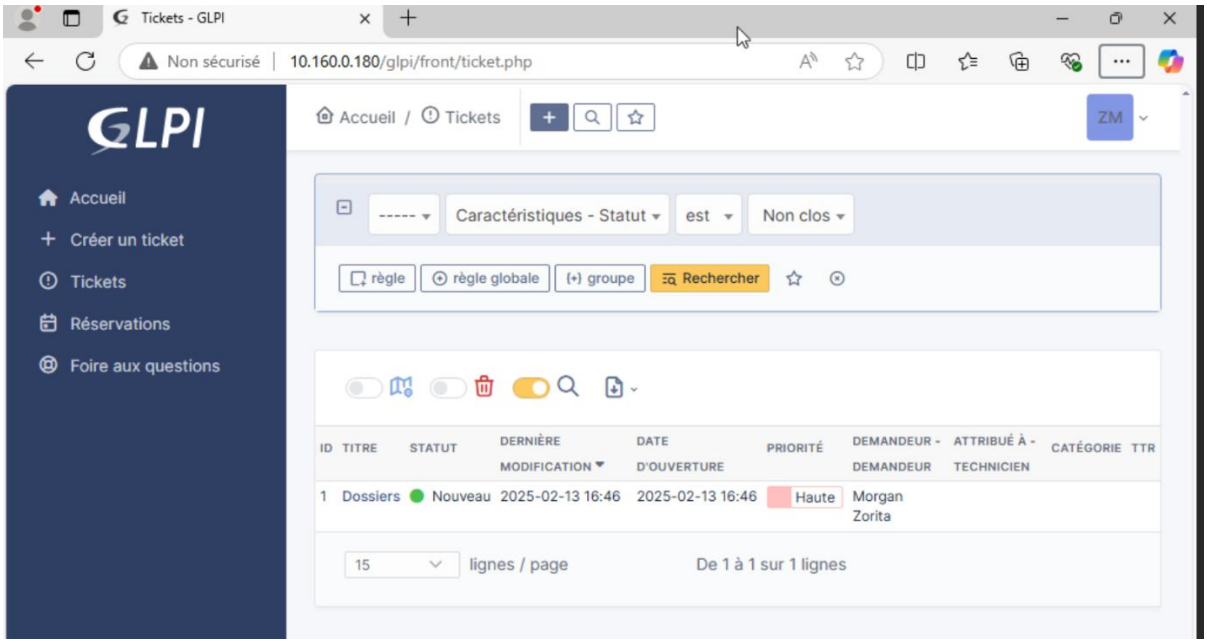
5. Résolution des derniers problèmes et finalisation

Une fois les agents installés et connectés au serveur, j'ai effectué plusieurs tests :

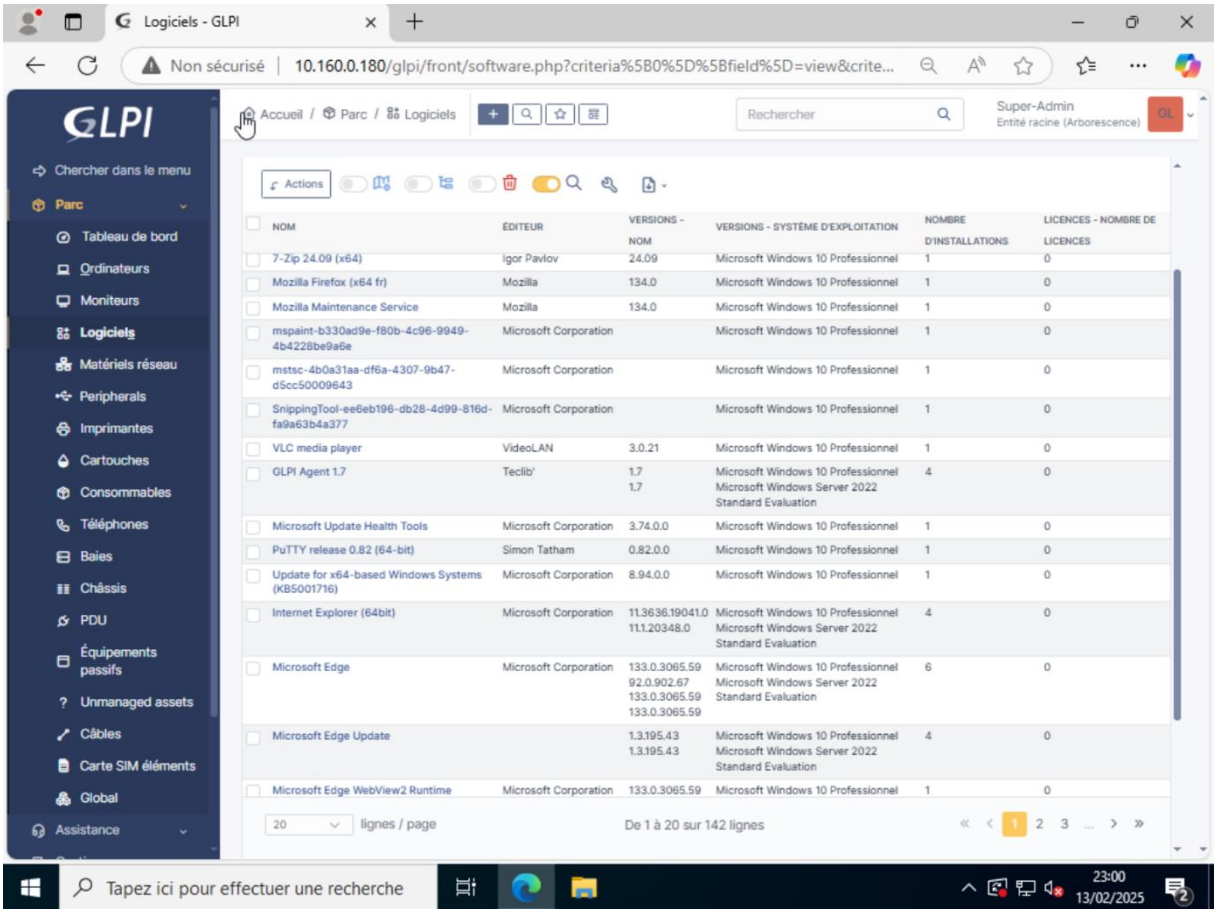
- **Connexion des utilisateurs LDAP** (qui a nécessité une suppression et réimportation des comptes après un problème de synchronisation)
- **Test d'un ticket d'incident** pour valider le fonctionnement du **helpdesk**







- **Forçage d'un inventaire** via l'interface web pour confirmer la remontée des équipements, logiciels …



L’installation et la configuration de GLPI sont donc terminées, et le système est pleinement fonctionnel. Il permet désormais de **gérer les incidents, suivre les équipements, et centraliser la gestion des actifs informatiques et des interventions.**