CYCLE DE FORMATION

TECHNICIEN RÉSEAUX,SYSTEME ET SECURITÉ INFORMATIQUE

Titre RNCP – Niveau 5

PROJET DE FIN D'ETUDES

Sujet :

Centralisation des accès et gestion des tickets IT via Active Directory et GLPI

RÉALISÉ par : Darius ILOKI

Période du 11 Février 2025 – 11 Mars 2025

Remerciements

Ce projet a été réalisé de manière autonome, mais il n'aurait pas été possible sans le soutien et les précieux conseils de ceux qui m'ont accompagné. Je tiens particulièrement à remercier **M. SIDIBE MAMOUTOU**, mon formateur support informatique, pour son accompagnement et son expertise, ainsi que **M. Lahoucine EL KAMEL**, mon responsable de stage, pour son encadrement et ses conseils précieux. Leur aide a été déterminante dans la réussite de ce projet.

TABLE DES MATIERES

1. INTRODUCTION

1.1 RESUME

Dans le cadre de mon BTS SIO option SISR, j'ai choisi de réaliser un projet visant à intégrer Active Directory avec GLPI afin d'automatiser la gestion des utilisateurs IT et d'améliorer le support technique. Ce projet répond à un besoin concret en entreprise : centraliser la gestion des accès et optimiser l'attribution des tickets IT. Pour cela, j'ai mis en place un environnement virtualisé comprenant :

- Un serveur Windows Server 2022 (Active Directory, DNS),
- Serveur Debian 12 (GLPI, Apache, MariaDB, PHP)
- Une connexion sécurisée via LDAP.

Ce projet m'a permis d'acquérir des compétences en administration système, gestion des droits d'accès, automatisation avec PowerShell et intégration d'outils ITSM.

1.2 ABSTRACT

This project aims to **integrate Active Directory (AD) users into GLPI** to optimize IT support management. By synchronizing AD with GLPI, user authentication and access management are streamlined, improving support response time and tracking.

1.3 PRESENTATION PERSONNELLE

Après une formation initiale en économie et commerce international, j'ai choisi de me réorienter vers l'informatique en 2023 en intégrant un **BTS Services Informatiques aux Organisations (SIO), option SISR** à **Webitech Paris**.

Au cours de mon stage chez **EasyFormer**, j'ai renforcé mes compétences en **administration des systèmes et réseaux**, notamment la gestion d'Active Directory et la mise en réseau des équipements. Mon alternance chez **NexFormation** m'a permis d'approfondir mes connaissances en **support informatique, configuration réseau et cybersécurité**.

Ce projet de **centralisation des utilisateurs AD dans GLPI** est l'occasion pour moi d'appliquer mes compétences en **administration système, gestion des accès et sécurité informatique** dans un environnement virtualisé.

2. PRESENTATION DU PROJET

2.1 CONTEXTE ET ENJEUX

Dans un environnement où la gestion des utilisateurs et du support IT devient de plus en plus complexe, l'objectif est de simplifier **l'authentification et l'organisation des accès** en intégrant **Active Directory à GLPI**.

Enjeux du projet :

- **Fiabilité** : Assurer une authentification centralisée et sécurisée.
- **Efficacité** : Réduire les délais de traitement des tickets grâce à l'automatisation.
- **Traçabilité** : Obtenir une meilleure visibilité sur les interventions du support IT.
- **Évolutivité** : Préparer l'infrastructure à une montée en charge future.

2.2 OBJECTIFS ET PROBLEMATIQUE

2.2.1 OBJECTIFS :

- Automatiser l'importation des utilisateurs depuis AD.
- Améliorer la gestion des tickets en fonction des groupes AD.
- Garantir une meilleure traçabilité des interventions.

2.2.2 PROBLEMATIQUE : Comment automatiser la gestion des utilisateurs IT et optimiser le support technique grâce à la centralisation dans GLPI ?

3. EXPRESSION DES BESOINS

3.1 ANALYSE FONCTIONNELLE

L'analyse fonctionnelle a permis d'identifier les éléments suivants :

- Les utilisateurs doivent être authentifiés automatiquement via **Active Directory**.
- Les tickets doivent être assignés en fonction des groupes **AD**.
- L'administration doit pouvoir **gérer les droits et les accès facilement**.

3.2 CAHIER DES CHARGES

Section	**Détails**
1. Présentation du projet	Mise en place d'une infrastructure IT sur un serveur physique hébergeant des machines virtuelles locales sous VMware Workstation Pro - Windows Server 2022 (Active Directory, DNS). - Debian 12 (Apache, MariaDB, PHP, GLPI).
Objectifs	- Déploiement d'un contrôleur de domaine (ADDS) et d'un serveur DNS sous Windows Server. - Automatisation de la création des utilisateurs, groupes et OU via PowerShell. - Installation et configuration d'un serveur Debian 12 pour Apache, MariaDB, PHP et GLPI.
	Infrastructure : - Serveur physique hébergeant des VMs sous VMware Workstation Pro.

2. Périmètre du projet	- VM 1 : Windows Server 2022 (ADDS, DNS). - VM 2 : Debian 12 (Apache, MariaDB, PHP, GLPI). Services à déployer : - Windows Server : ADDS, DNS. - Debian 12 : Apache, MariaDB, PHP, GLPI.
Contraintes techniques	Stockage local : Les VMs sont hébergées sur le disque du serveur physique. - Sécurité : Pare-feu Windows et iptables/UFW sous Debian.
3. Plan de réalisation	Étapes du projet : 1Installation de VMware Workstation Pro sur le serveur physique. 2 Création de la VM Windows Server et installation du rôle ADDS + DNS. 3 Configuration du domaine et des utilisateurs via PowerShell. 4 Création de la VM Debian et installation des services Apache, MariaDB, PHP. 5 Déploiement et configuration de GLPI. 6 Optimisation des performances des VMs (RAM, CPU, disque). 7 Tests et validation des services.
4. Ressources nécessaires	Ressources matérielles : - 1 serveur physique avec 16+ Go RAM, CPU 6+ cœurs, 1 To SSD. - Disque SSD/NVMe pour stocker les VMs. - Switch réseau physique ou virtuel sous VMware. Ressources logicielles : - VMware Workstation Pro. - Windows Server 2022 (ADDS, DNS). - Debian 12. - Apache, MariaDB, PHP, GLPI (100% open source).
6. Critères de validation	Windows Server 2022 opérationnel avec ADDS et DNS. Automatisation de la gestion des utilisateurs via PowerShell. Debian 12 + services web fonctionnels. GLPI installé et accessible.
7. Enjeux du projet	**Enjeux techniques** : **Fiabilité** : Garantir un fonctionnement optimal et sécurisé des VMs. **Scalabilité** : Permettre une évolution future (ajout de services, utilisateurs, ressources).
8. Livrables	Captures d'écran des étapes clés.
9. Conclusion	Ce projet repose sur un serveur physique stockant des VMs localement sous **VMware Workstation Pro**, garantissant performance, sécurité et flexibilité. Il met l'accent sur l'optimisation des performances, la réduction des coûts grâce à l'open source et la sécurisation des données via des stratégies de sauvegarde et de protection contre les cyberattaques.

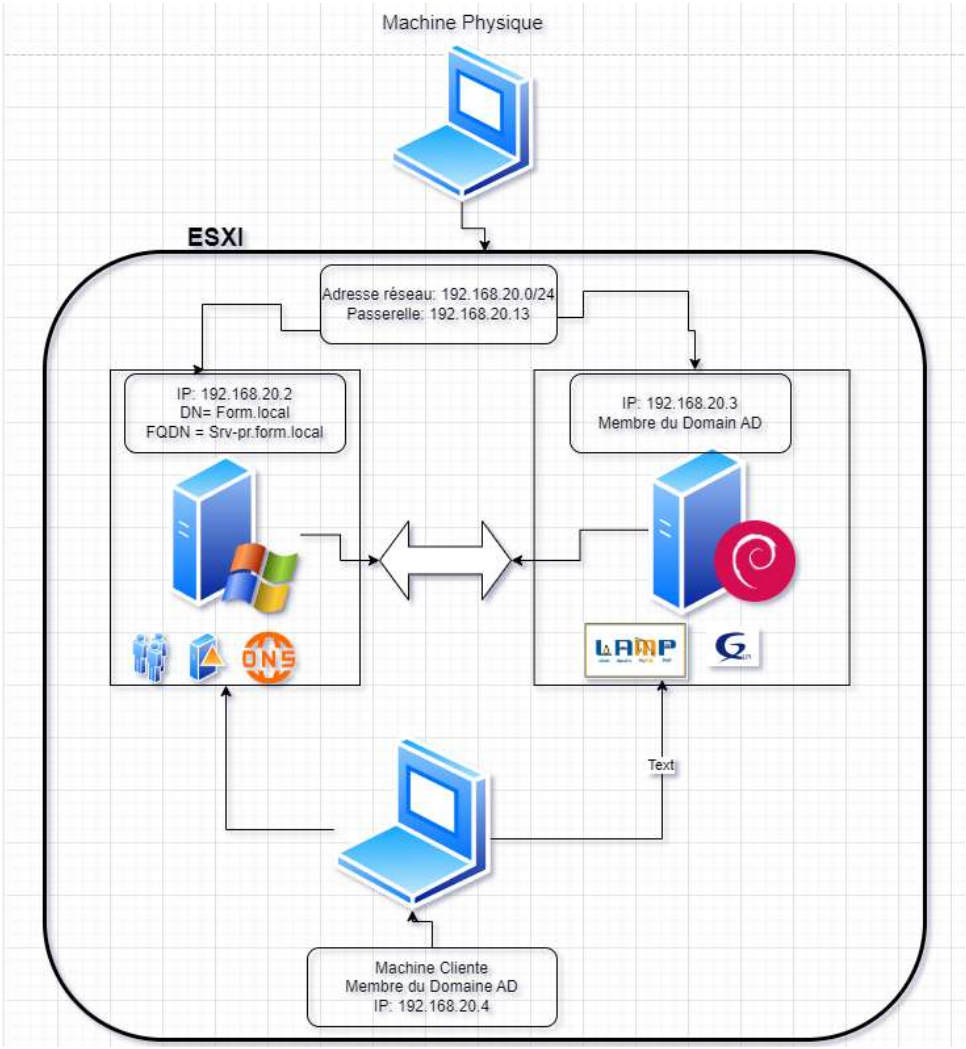
3.3 GESTION DES RISQUES ET DES ENJEUX

ID	Risque/Enjeu	Détails	Impact	Probabilité	Criticité
R1	Échec de la synchronisation AD-GLPI	Accès bloqués pour certains utilisateurs.	Moyenne	Élevée	Élevée
R2	Mauvaise attribution des tickets	Perte d'efficacité du support IT.	Haute	Moyenne	Moyenne
R3	Problèmes de performance sur GLPI	Temps de réponse allongé.	Faible	Moyenne	Moyenne
R4	Failles de sécurité dans la gestion des droits	Risque d'accès non autorisé	Élevée	Élevée	Élevée
E1	Fiabilité	Garantir un fonctionnement optimal et sécurisé des VMs.	Fonctionnement fluide et sécurisé	-	-
E2	Scalabilité	Permettre une évolution future (ajout de services, utilisateurs, ressources).	Évolution et croissance possibles	-	-
E3	Performance	Optimisation des ressources pour éviter la surcharge des serveurs et assurer une expérience fluide.	Réduction des temps d'arrêt	-	-
E4	Rentabilité financière	Réduire les coûts en utilisant des solutions open-source et en optimisant les ressources matérielles.	Réduction des coûts d'exploitation	-	-

R = Risque

E = Enjeux

3.4 TOPOLOGIE LOGIQUE DU PROJET



4 PLAN D'IMPLEMENTATION

Étape1: Installation et configuration de Windows Server avec AD
Étape 2: Installation de Debian et des services nécessaires (GLPI, Apache, MariaDB, PHP)
Étape3: Configuration de l'intégration LDAP entre AD et GLPI
Étape 4: Test et validation de la synchronisation

5. REALISATION

5.1 INSTALLATION ET CONFIGURATION

5.1.**1 :** INSTALLATION ET CONFIGURATION DE L'ACTIVE DIRECTORY, DU DNS ET GESTION VIA POWERSHELL ET DU CLIENT WINDOWS 10

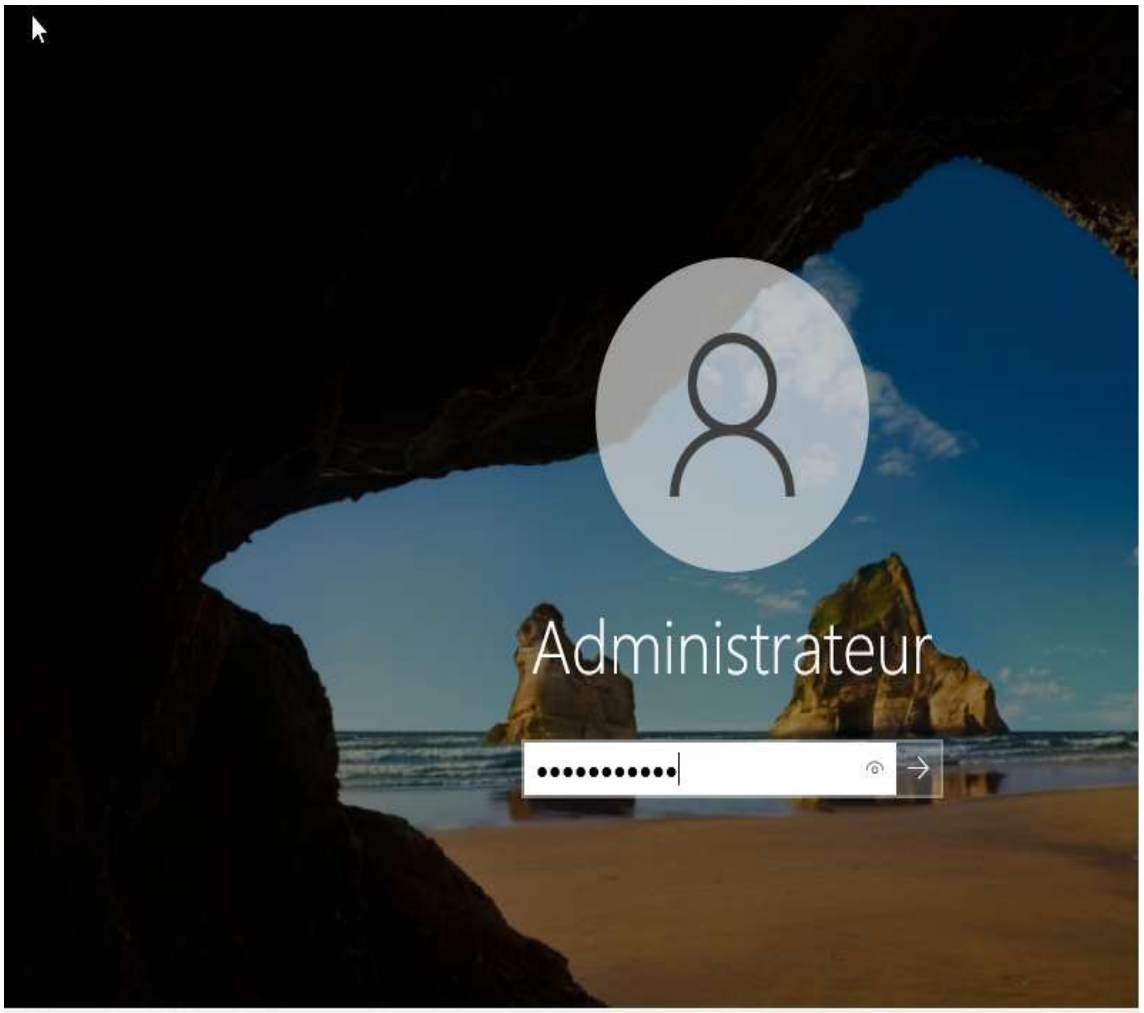


FIGURE 1

Dans ce projet, l'installation de l'infrastructure **Active Directory (AD)** a été réalisée de manière autonome. Le processus a suivi les étapes suivantes :

- **Installation du rôle Active Directory Domain Services (ADDS)**
- **Création du domaine Active Directory**
- **Configuration du serveur DNS pour l'Active Directory**
- **Gestion des utilisateurs, groupes et unités d'organisation (OU) via PowerShell**

5.1.1.1 INSTALLATION DU ROLE ACTIVE DIRECTORY DOMAIN SERVICES (ADDS)

La première étape a été l'installation du rôle **Active Directory Domain Services (ADDS)**. Ce rôle est essentiel pour permettre à un serveur d'agir en tant que **contrôleur de domaine** dans l'infrastructure de l'entreprise.

Étapes suivies :

1. **Ouverture du Gestionnaire de serveur.**

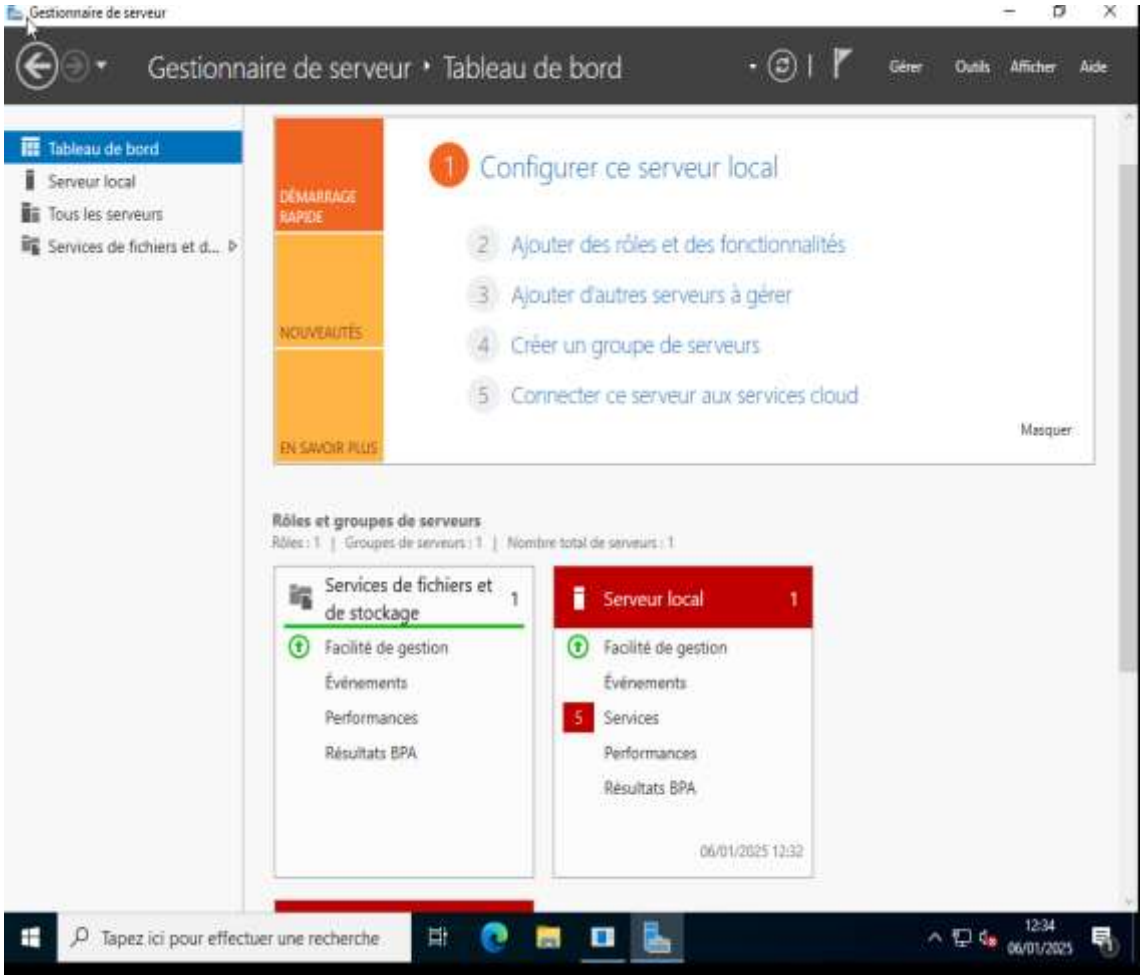
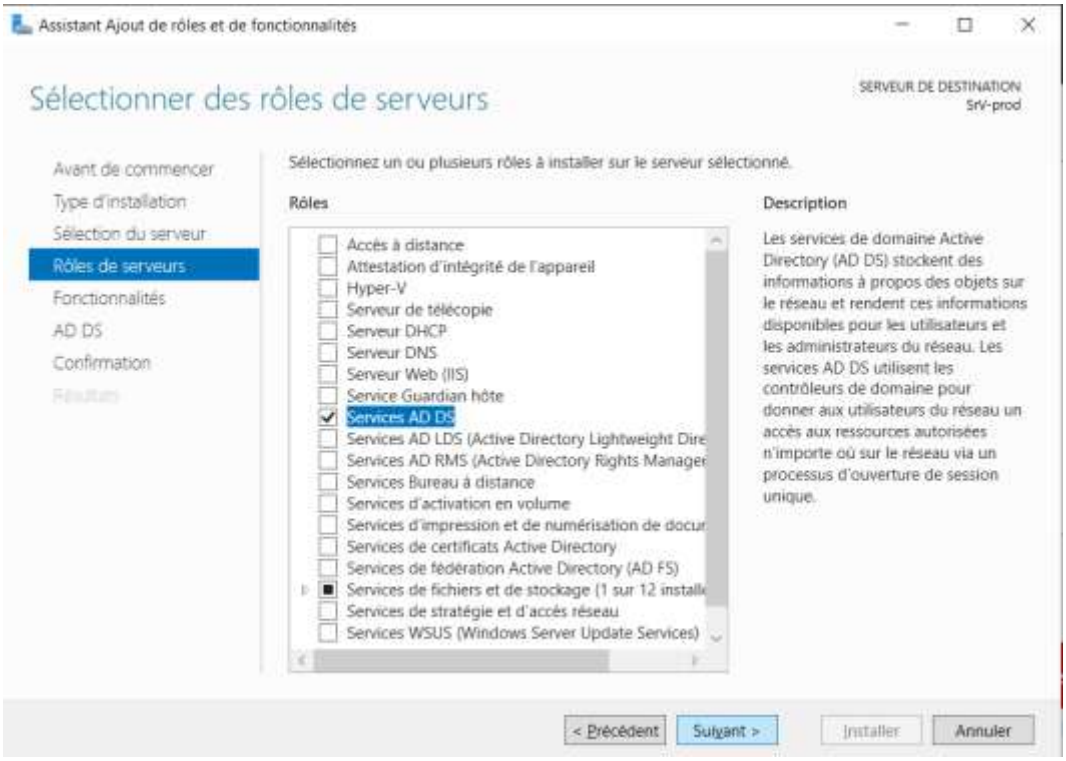


FIGURE 2

2. Sélection de **Ajouter des rôles et fonctionnalités**.

FIGURE 3

3. Choix du rôle **Active Directory Domain Services (ADDS)** et finalisation de l'installation.



CETTE CAPTURE MONTRE L'AJOUT DU ROLE ACTIVE DIRECTORY DOMAIN SERVICES (ADDS) VIA LE GESTIONNAIRE DE SERVEUR SOUS WINDOWS SERVER 2022. CE ROLE PERMET DE TRANSFORMER LE SERVEUR EN CONTROLEUR DE DOMAINE, ESSENTIEL POUR LA GESTION DES UTILISATEURS ET DES RESSOURCES RESEAU.

Valider l'installation et redémarrer le serveur pour appliquer les modifications.

5.1.1.2 CREATION DU DOMAINE ACTIVE DIRECTORY ET PROMOUVOIR CE SERVEUR EN CONTROLEUR DE DOMAINE.

Rappel :

Un domaine, en informatique, est une structure logique utilisée pour regrouper et gérer des ressources réseau sous une même autorité. Dans les environnements Windows, un domaine Active Directory (AD), introduit par Microsoft en 1999, permet l'authentification centralisée et la gestion des utilisateurs, ordinateurs et politiques de sécurité. Il repose sur des composants comme **le contrôleur de domaine (DC)** (serveur d'authentification), **LDAP** (protocole d'annuaire), **Kerberos** (authentification sécurisée), **Group Policy** (gestion des stratégies) et **DNS** (résolution des noms). Grâce à son administration centralisée et sa scalabilité, un domaine est essentiel pour les entreprises, assurant la sécurité, la gestion des accès et la cohérence des configurations sur un réseau.

Après l'installation du rôle **ADDS**, j'ai promu le serveur en **contrôleur de domaine** et créé le domaine Form.local.

Étapes suivies :

1. Lancement de l'**Assistant de promotion de contrôleur de domaine**.

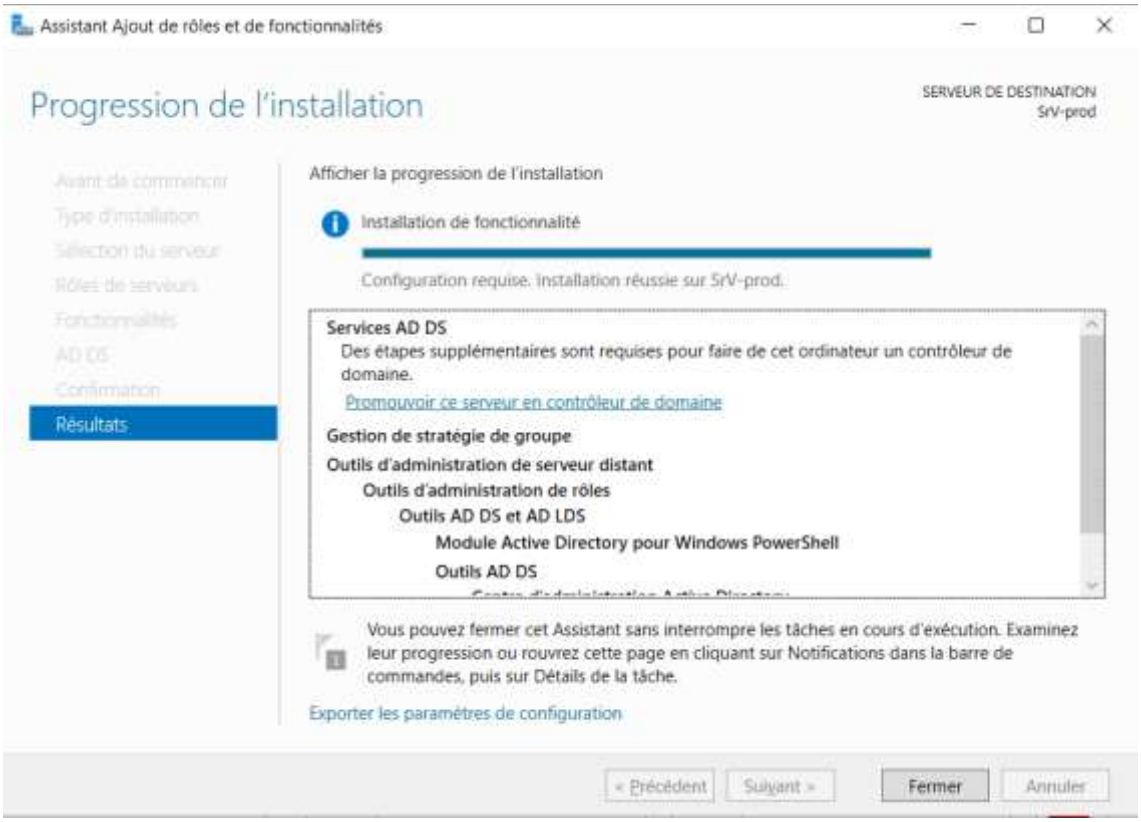


FIGURE 4

2. Sélection de l'option pour **créer un nouveau domaine dans une nouvelle forêt**. Et Configuration du nom du domaine **SISR.local**.

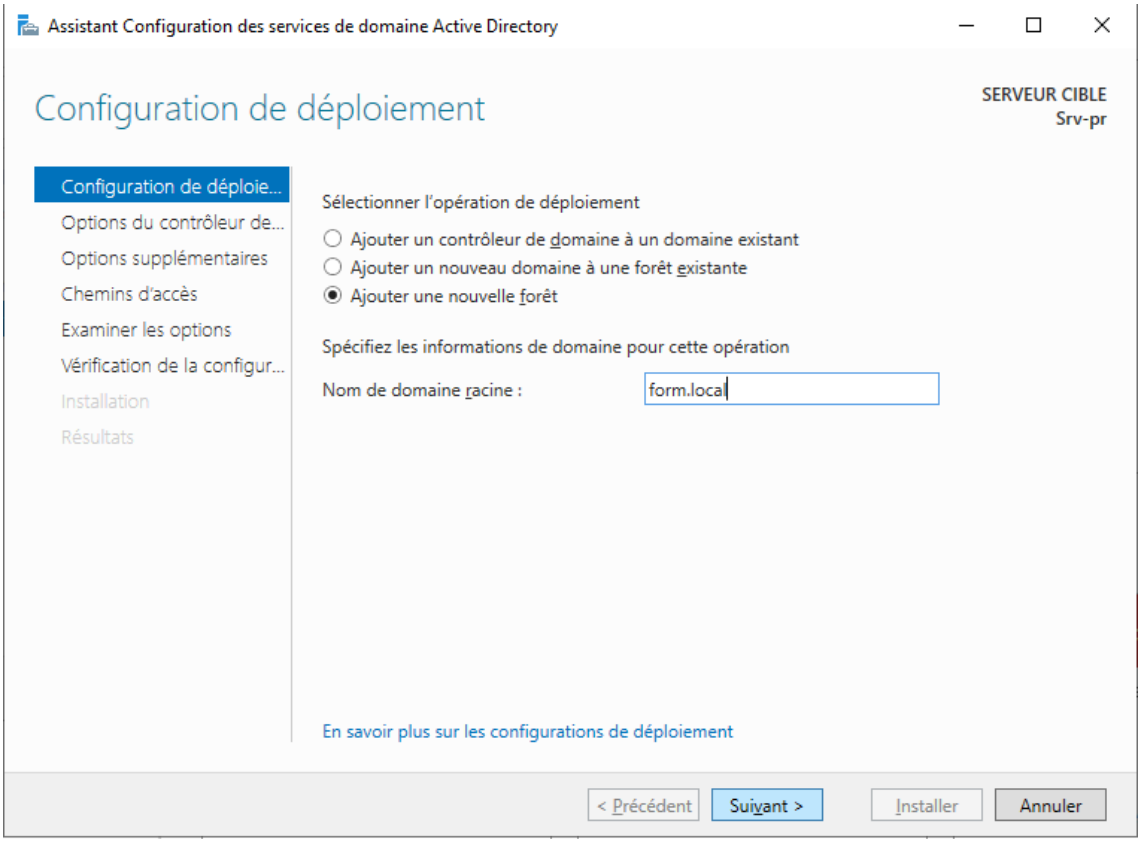


FIGURE 5

3. Finalisation de la promotion et redémarrage du serveur pour appliquer les changements.

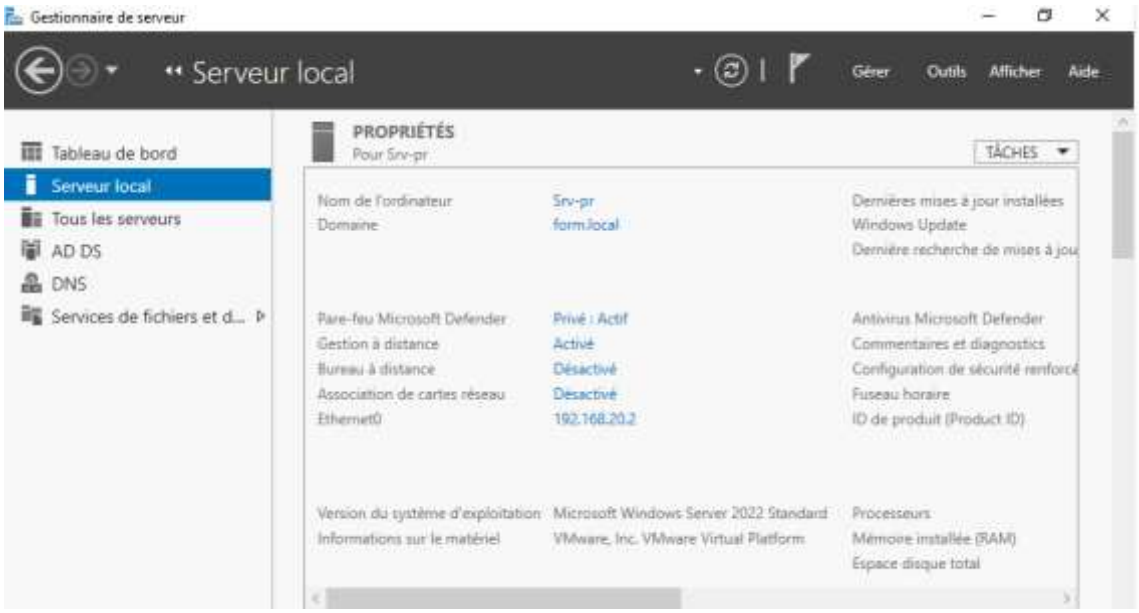


FIGURE 6

5.1.1.3 CONFIGURATION DU SERVEUR DNS

Rappel :

Le DNS (Domain Name System), créé en 1983 par Paul Mockapetris, est un système permettant de traduire les noms de domaine lisibles par les humains (ex. www.example.com) en adresses IP compréhensibles par les machines. Il fonctionne sur un modèle hiérarchique et distribué, avec des serveurs racine, des serveurs TLD (Top-Level Domain) et des serveurs faisant autorité. Le DNS utilise principalement le protocole UDP sur le port 53 pour les requêtes rapides et TCP pour les transferts de zones. Il est essentiel pour la navigation sur Internet et joue un rôle clé dans les services réseau, y compris l'Active Directory, où il est utilisé pour localiser les contrôleurs de domaine.

Une fois le domaine créé, j'ai configuré le serveur en tant que serveur **DNS** pour qu'il puisse résoudre les noms de domaine du réseau. Le serveur DNS est une partie intégrante de l'Active Directory, car il permet la résolution des noms des machines du domaine.

Étapes suivies :

1. Ouverture du **Gestionnaire de serveur**.

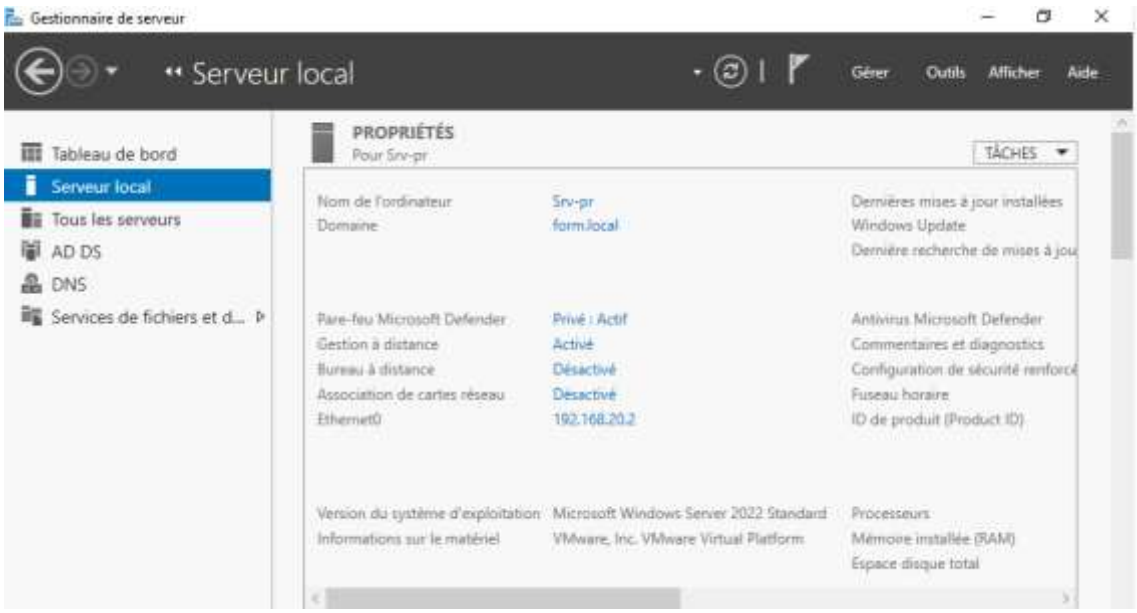


FIGURE 7

2. • Sélection **DNS** puis Gestionnaire DNS.

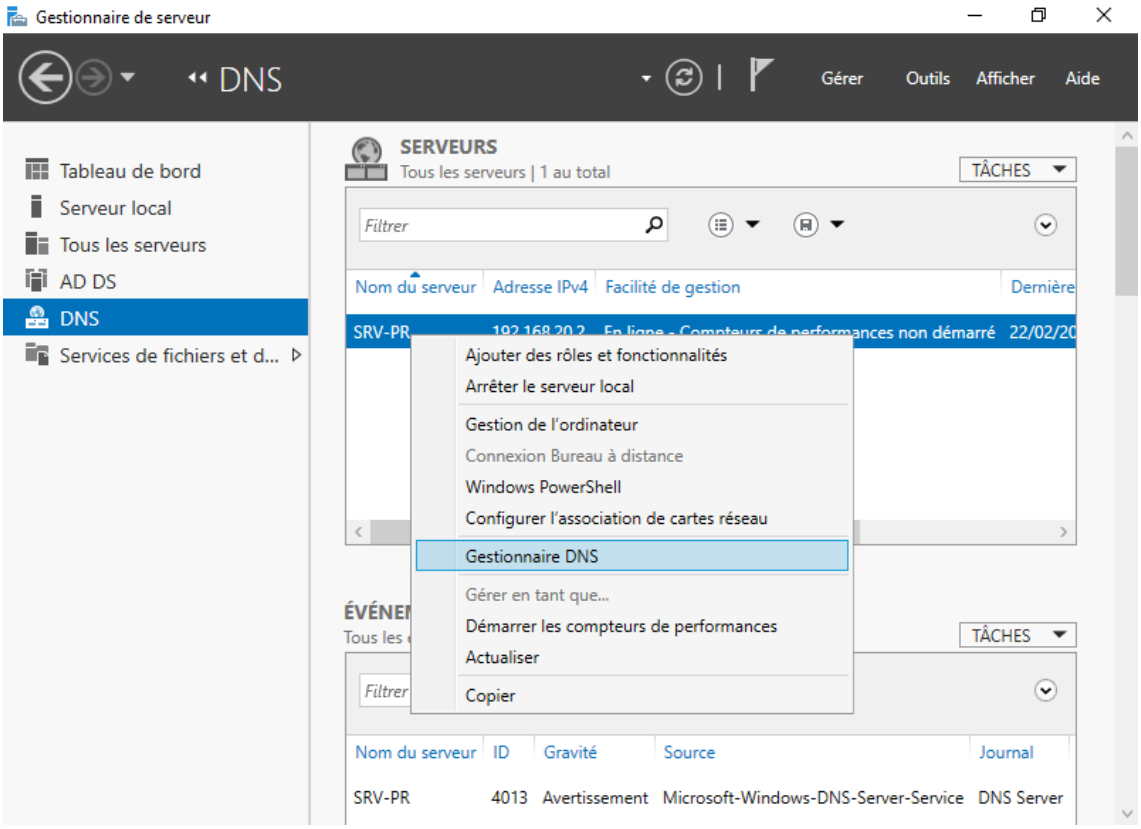


FIGURE 8

3. • Configuration de la **zone de recherche indirecte** pour **form.local**.

FIGURE 9

4. Configuration de la **zone de recherche directe** pour **form.local**.

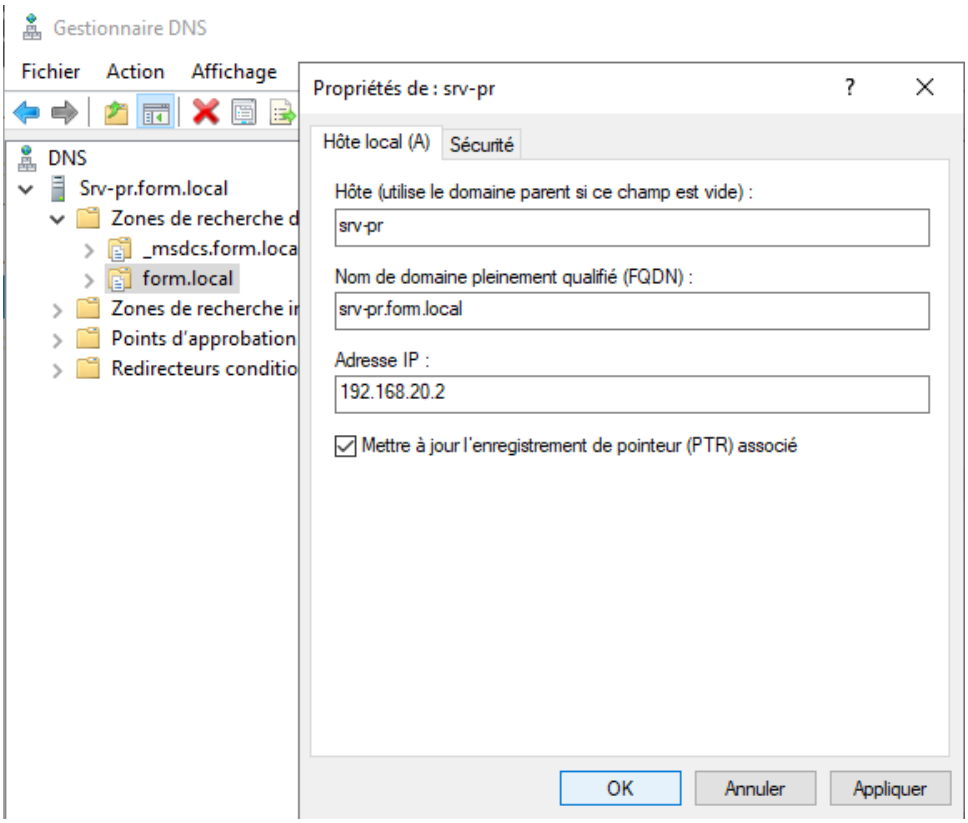


FIGURE 10

Cette capture montre la configuration du serveur DNS, essentiel pour la bonne communication entre les machines du domaine.

5. • Vérification des enregistrements SRV nécessaires pour le bon fonctionnement de l'Active Directory.

FIGURE 11

5.1.1.4 GESTION DES UTILISATEURS, GROUPES ET UNITES D'ORGANISATION VIA POWERSHELL.

Rappel :

Le **PowerShell**, créé par **Microsoft** en **2006**, est un shell et un langage de script basé sur **.NET**, conçu pour l'automatisation et l'administration des systèmes Windows. Il utilise des **cmdlets** comme Get-Command (liste des commandes), Get-Help (aide sur une commande), Get-Process (affichage des processus), Set-ExecutionPolicy (gestion de la politique d'exécution des scripts) et Invoke-Command (exécution à distance). Grâce à son système de **pipelines** et sa prise en charge des **scripts avancés**, il est essentiel pour l'administration de **Windows Server, Active Directory et Azure**, offrant un contrôle puissant sur les infrastructures IT.

Après avoir configuré le rôle **Active Directory Domain Services (ADDS)**, créé le domaine et configuré le serveur **DNS**, j'ai utilisé **PowerShell** pour automatiser la gestion des utilisateurs, des groupes et des unités d'organisation (OU).

Toutes les commandes PowerShell nécessaires à la gestion de l'Active Directory ont été exécutées dans une seule session PowerShell.

Sans titre1.ps1*

```powershell
#Création d'une unité d'organisation (OU)
New-ADOrganizationalUnit -Name "TECHNIQUE" -Path "DC=form,DC=local" -ProtectedFromAccidentalDeletion $true -Server "Srv-pr.form.local"
New-ADOrganizationalUnit -Name "COMPTABILITE" -Path "DC=form,DC=local" -ProtectedFromAccidentalDeletion $true -Server "Srv-pr.form.local"
New-ADOrganizationalUnit -Name "DIRECTION" -Path "DC=form,DC=local" -ProtectedFromAccidentalDeletion $true -Server "Srv-pr.form.local"

#Création des utilisateur et ajout des utilisateur dans les OU
New-ADUser -Name "Synetra Alex" -GivenName "Synetra" -Surname "alex" -SamAccountName "S.alex" -UserPrincipalName "S.alex@form.local" `
-Path "OU=TECHNIQUE,DC=form,DC=local" -AccountPassword (ConvertTo-SecureString "Respons11!" -AsPlainText -Force) -Enabled $true

New-ADUser -Name "CYBERD Sam" -GivenName "Sybero" -Surname "sam" -SamAccountName "S.sam" -UserPrincipalName "S.sam@form.local" `
-Path "OU=TECHNIQUE,DC=form,DC=local" -AccountPassword (ConvertTo-SecureString "Respons11!" -AsPlainText -Force) -Enabled $true

New-ADUser -Name "NETZIA Jordan" -GivenName "Netzia" -Surname "jordan" -SamAccountName "N.jordan" -UserPrincipalName "N.jordan@form.local" `
-Path "OU=COMPTABILITE,DC=form,DC=local" -AccountPassword (ConvertTo-SecureString "Respons11!" -AsPlainText -Force) -Enabled $true

New-ADUser -Name "VIRTECH Max" -GivenName "Virtech" -Surname "max" -SamAccountName "V.max" -UserPrincipalName "V.maxx@form.local" `
-Path "OU=COMPTABILITE,DC=form,DC=local" -AccountPassword (ConvertTo-SecureString "Respons11!" -AsPlainText -Force) -Enabled $true

New-ADUser -Name "DATEX Leo" -GivenName "Datex" -Surname "leo" -SamAccountName "D.leo" -UserPrincipalName "D.leo@form.local" `
-Path "OU=DIRECTION,DC=form,DC=local" -AccountPassword (ConvertTo-SecureString "Respons11!" -AsPlainText -Force) -Enabled $true

New-ADUser -Name "SECURIA Nova" -GivenName "Securia" -Surname "nova" -SamAccountName "S.nova" -UserPrincipalName "S.nova@form.local" `
-Path "OU=DIRECTION,DC=form,DC=local" -AccountPassword (ConvertTo-SecureString "Respons11!" -AsPlainText -Force) -Enabled $true

#Création des Groupes
New-ADGroup -Name "Tech" -SamAccountName "Tech" -GroupCategory "Security" -GroupScope "Global" -Path "OU=TECHNIQUE,DC=form,DC=local"
New-ADGroup -Name "Compta" -SamAccountName "Compta" -GroupCategory "Security" -GroupScope "Global" -Path "OU=COMPTABILITE,DC=form,DC=local"
New-ADGroup -Name "Direct" -SamAccountName "Direct" -GroupCategory "Security" -GroupScope "Global" -Path "OU=DIRECTION,DC=form,DC=local"

#Ajout des utilisateurs dans les Groupes
Add-ADGroupMember -Identity "Tech" -Members "S.alex", "S.sam", "D.leo", "S.nova"
Add-ADGroupMember -Identity "Compta" -Members "N.jordan", "V.max", "D.leo", "S.nova"
Add-ADGroupMember -Identity "Direct" -Members "D.leo", "S.nova"
```

```
PS C:\Users\Administrateur>  #Création d'une unité d'organisation (OU)
New-ADOrganizationalUnit -Name "TECHNIQUE" -Path "DC=form,DC=local" -ProtectedFromAccidentalDeletion $true -Server "Srv-pr.form.local"
New-ADOrganizationalUnit -Name "COMPTABILITE" -Path "DC=form,DC=local" -ProtectedFromAccidentalDeletion $true -Server "Srv-pr.form.local"
New-ADOrganizationalUnit -Name "DIRECTION" -Path "DC=form,DC=local" -ProtectedFromAccidentalDeletion $true -Server "Srv-pr.form.local"

#Création des utilisateur et ajout des utilisateur dans les OU
New-ADUser -Name "Synetra Alex" -GivenName "Synetra" -Surname "alex" -SamAccountName "S.alex" -UserPrincipalName "S.alex@form.local" `
-Path "OU=TECHNIQUE,DC=form,DC=local" -AccountPassword (ConvertTo-SecureString "Respons11!" -AsPlainText -Force) -Enabled $true

New-ADUser -Name "CYBERD Sam" -GivenName "Sybero" -Surname "sam" -SamAccountName "S.sam" -UserPrincipalName "S.sam@form.local" `
-Path "OU=TECHNIQUE,DC=form,DC=local" -AccountPassword (ConvertTo-SecureString "Respons11!" -AsPlainText -Force) -Enabled $true

New-ADUser -Name "NETZIA Jordan" -GivenName "Netzia" -Surname "jordan" -SamAccountName "N.jordan" -UserPrincipalName "N.jordan@form.local" `
-Path "OU=COMPTABILITE,DC=form,DC=local" -AccountPassword (ConvertTo-SecureString "Respons11!" -AsPlainText -Force) -Enabled $true

New-ADUser -Name "VIRTECH Max" -GivenName "Virtech" -Surname "max" -SamAccountName "V.max" -UserPrincipalName "V.maxx@form.local" `
-Path "OU=COMPTABILITE,DC=form,DC=local" -AccountPassword (ConvertTo-SecureString "Respons11!" -AsPlainText -Force) -Enabled $true

New-ADUser -Name "DATEX Leo" -GivenName "Datex" -Surname "leo" -SamAccountName "D.leo" -UserPrincipalName "D.leo@form.local" `
-Path "OU=DIRECTION,DC=form,DC=local" -AccountPassword (ConvertTo-SecureString "Respons11!" -AsPlainText -Force) -Enabled $true

New-ADUser -Name "SECURIA Nova" -GivenName "Securia" -Surname "nova" -SamAccountName "S.nova" -UserPrincipalName "S.nova@form.local" `
-Path "OU=DIRECTION,DC=form,DC=local" -AccountPassword (ConvertTo-SecureString "Respons11!" -AsPlainText -Force) -Enabled $true

#Création des Groupes
New-ADGroup -Name "Tech" -SamAccountName "Tech" -GroupCategory "Security" -GroupScope "Global" -Path "OU=TECHNIQUE,DC=form,DC=local"
New-ADGroup -Name "Compta" -SamAccountName "Compta" -GroupCategory "Security" -GroupScope "Global" -Path "OU=COMPTABILITE,DC=form,DC=local"
New-ADGroup -Name "Direct" -SamAccountName "Direct" -GroupCategory "Security" -GroupScope "Global" -Path "OU=DIRECTION,DC=form,DC=local"

#Ajout des utilisateurs dans les Groupes
Add-ADGroupMember -Identity "Tech" -Members "S.alex", "S.sam", "D.leo", "S.nova"
Add-ADGroupMember -Identity "Compta" -Members "N.jordan", "V.max", "D.leo", "S.nova"
Add-ADGroupMember -Identity "Direct" -Members "D.leo", "S.nova"

PS C:\Users\Administrateur>
```

FIGURE 12

5.1.1.4.1 VERIFICATION DES ENREGISTREMENTS SRV NECESSAIRES POUR LE BON FONCTIONNEMENT DE L'ACTIVE DIRECTORY.

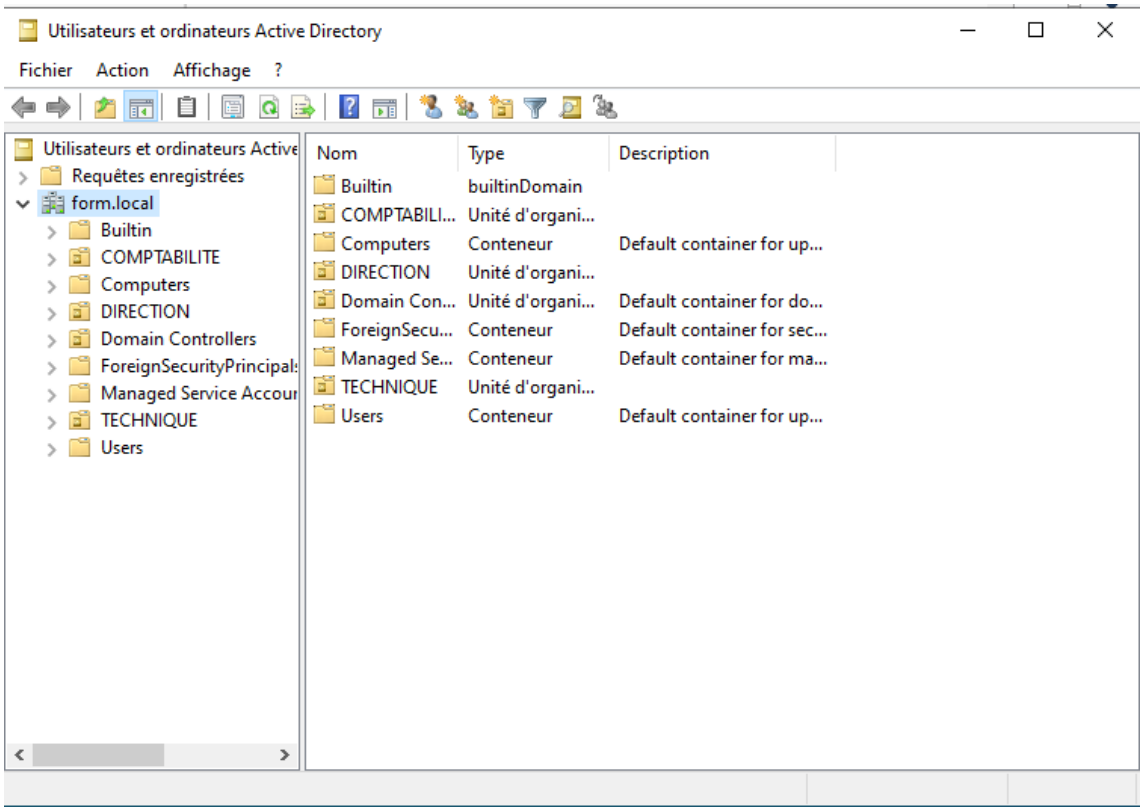


FIGURE 13

5.1.1.2 INSTALLATION DU CLIENT WINDOWS 10

1. **Téléchargement de l'image Windows 10** depuis le site officiel.
2. **Installation de windows 10** sur le serveur en utilisant l'installateur standard.

L'installation du client Windows 10 vise à permettre aux utilisateurs du domaine Active Directory de s'authentifier automatiquement sur GLPI

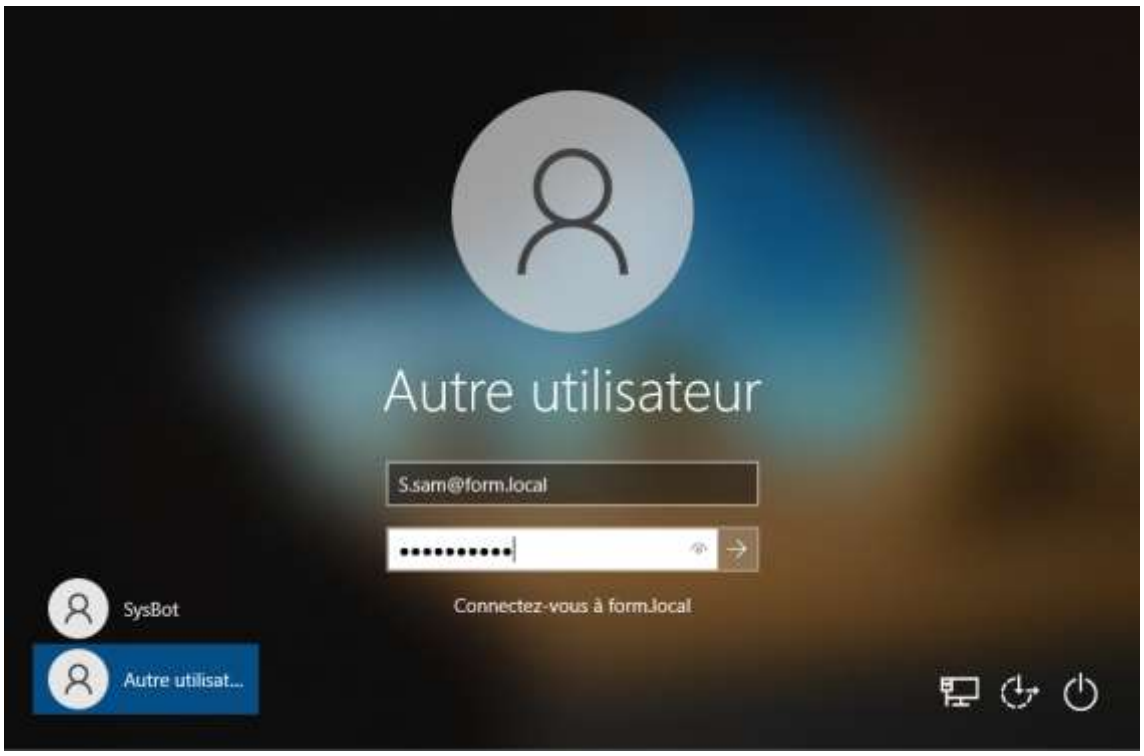


FIGURE 14

5.1.2 INSTALLATION DE DEBIAN ET DES SERVICES NECESSAIRES (GLPI, APACHE, MARIADB, PHP)

Cette section décrit l'installation de **Debian** et de plusieurs services nécessaires pour la gestion d'un environnement IT, y compris **GLPI** (un outil de gestion de parc informatique), **Apache** (serveur web), **MariaDB** (système de gestion de bases de données), et **PHP** (langage de programmation nécessaire pour faire fonctionner GLPI).

5.1.2.1 INSTALLATION DE DEBIAN

Rappel :

Debian, lancé en 1993 par Ian Murdock, est une distribution Linux reconnue pour sa stabilité, sa sécurité et sa philosophie open-source. Elle repose sur un système de gestion de paquets robuste avec **APT** (apt update, apt upgrade, apt install pour gérer les logiciels), et **dpkg** pour manipuler les paquets individuellement. Debian utilise **systemd** (systemctl start, systemctl enable) pour l'administration des services et intègre des outils comme adduser (création d'utilisateurs) et passwd (gestion des mots de passe). Son cycle de développement rigoureux et son large support matériel en font un choix idéal pour les serveurs, les environnements de développement et les utilisateurs recherchant un système fiable et personnalisable.

Avant d'installer les services, j'ai d'abord mis en place un système Debian, en suivant les étapes classiques d'installation à partir d'une image ISO de **Debian**.

3. **Téléchargement de l'image Debian** depuis le site officiel.
4. **Installation de Debian** sur le serveur en utilisant l'installateur standard.

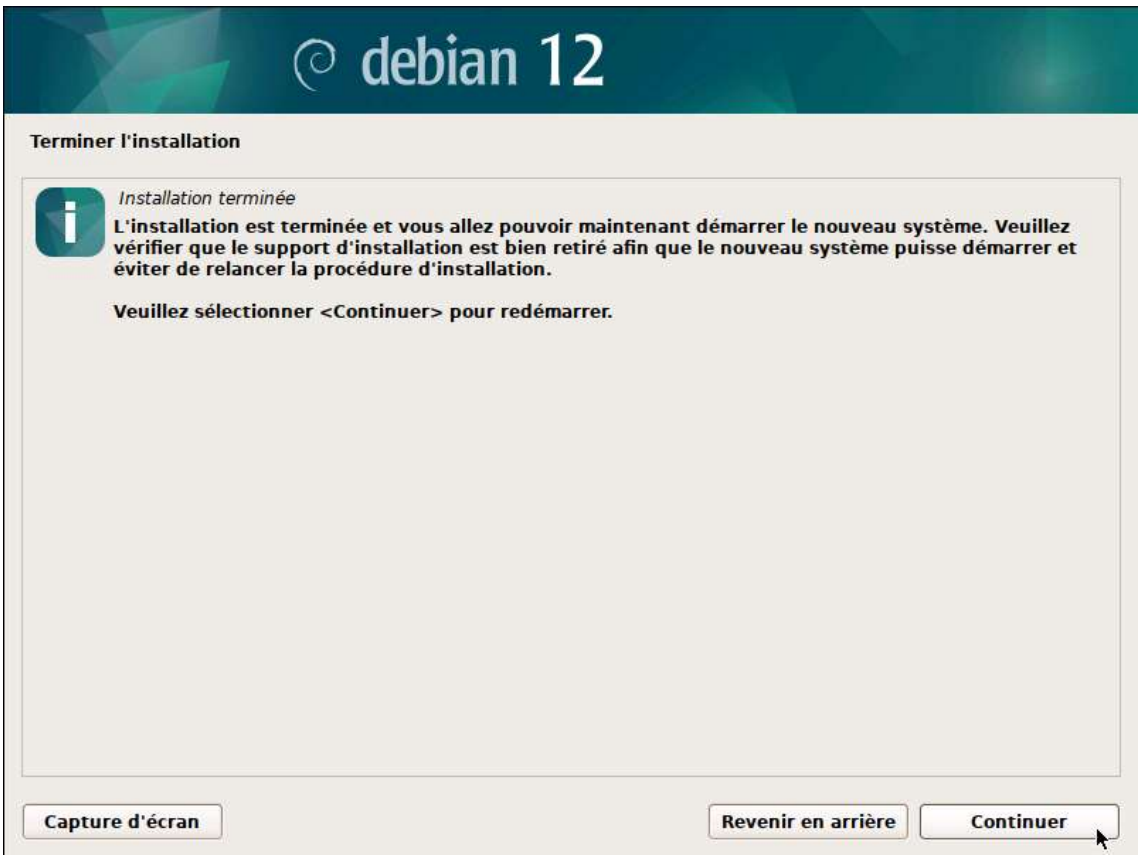


FIGURE 15

5. **Mise à jour du système** après l'installation de base.

```
# apt-get update && apt-get upgrade
```

Le **#** au début de chaque ligne indique que vous devez être connecté au terminal avec les privilèges du compte **root** pour exécuter la commande

5.1.2. INSTALLATION DES SERVICES NECESSAIRES

Une fois Debian installé, j'ai installé et configuré les services nécessaires : **Apache**, **MariaDB**, **PHP**, et **GLPI**.

Installation d'Apache

Rappel : Apache2, développé par la fondation Apache Software Foundation, est un serveur web open-source réputé pour sa robustesse, sa modularité et sa flexibilité. Il repose sur une architecture modulaire permettant d'activer des fonctionnalités via des modules (comme mod_rewrite pour la réécriture d'URL et mod_ssl pour le support HTTPS). Apache2 est géré sous Debian via le système de paquets APT (apt update, apt install apache2 pour l'installation) et administré avec systemd (systemctl start apache2

pour démarrer le service, systemctl enable apache2 pour l'activer au démarrage). Sa configuration repose sur des fichiers situés dans /etc/apache2/, avec des hôtes virtuels définis dans /etc/apache2/sites-available/. Grâce à sa compatibilité avec divers langages comme PHP et Python, ainsi que son support natif pour TLS/SSL, Apache2 est un choix privilégié pour l'hébergement de sites web, d'applications et d'API sécurisées.

1. **Installation du serveur Apache2** :

Pour l'installation du serveur web Apache2 utilisé la commande :

```
# apt-get install apache2 php libapache2-mod-php
```

2. **Vérification du bon fonctionnement d'Apache** :

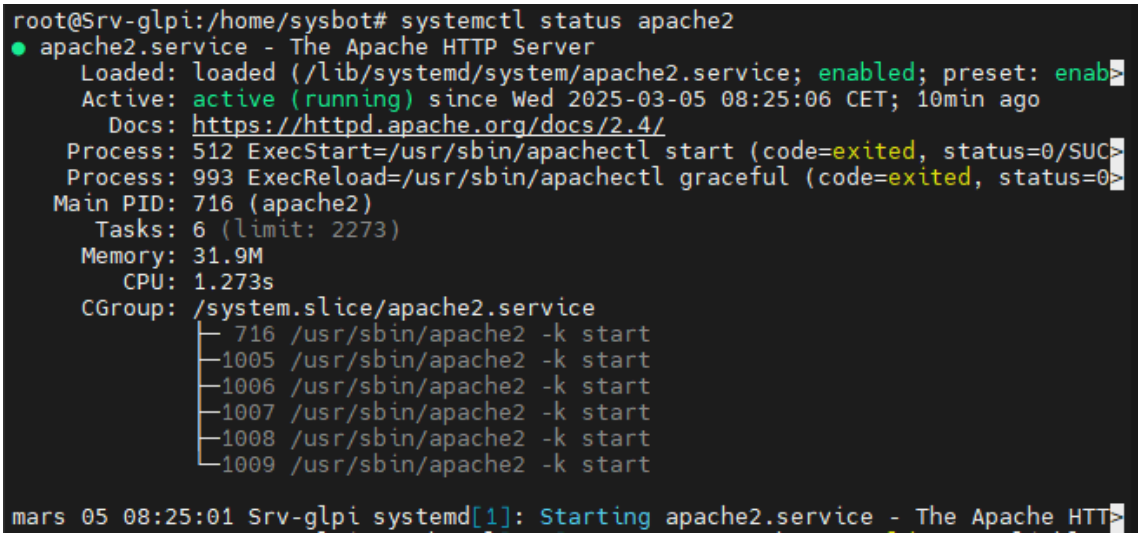


FIGURE 16

Installation de PHP

Rappel :

PHP, créé en 1994 par Rasmus Lerdorf, est un langage de script open-source largement utilisé pour le développement web dynamique. Il s'intègre parfaitement aux serveurs web comme Apache2 et Nginx, permettant l'exécution de scripts côté serveur. Sous Debian, PHP est géré via APT (apt update, apt install php pour l'installation) et peut être étendu grâce à des modules comme php-mysql (connexion aux bases de données MySQL) et php-gd (gestion des images). Sa configuration se trouve dans /etc/php/, avec des paramètres modifiables dans php.ini. PHP prend en charge les connexions sécurisées via OpenSSL et peut être exécuté en mode FPM (FastCGI Process Manager) pour améliorer les performances. Grâce à son écosystème riche (Laravel, Symfony, WordPress) et sa compatibilité avec

les bases de données, PHP est un choix incontournable pour le développement d'applications web dynamiques et évolutives.

1. **Installation de PHP et des modules nécessaires pour GLPI** :

Pour l'installation de PHP utilisé la commande :

```
# apt-get install php-imap php-ldap php-curl php-xmlrpc php-gd php-mysql php-cas php-dom php-simplexml php-intl php-bz2 php-zip php-mbstrig -y
```

2. **Vérification de l'installation de PHP** :

```
root@Srv-glpi:/home/sysbot# php -v
PHP 8.2.26 (cli) (built: Nov 25 2024 17:21:51) (NTS)
Copyright (c) The PHP Group
Zend Engine v4.2.26, Copyright (c) Zend Technologies
    with Zend OPcache v8.2.26, Copyright (c), by Zend Technologies
root@Srv-glpi:/home/sysbot#
```

FIGURE 17

Installation de MariaDB

Rappel :

MariaDB, fork de MySQL créé en 2009 par Michael Widenius, est un système de gestion de bases de données relationnelles (SGBD) open-source, réputé pour sa performance et sa fiabilité. Il est utilisé pour stocker et gérer des données de manière sécurisée, notamment dans les applications web et les systèmes d'information. Sous Debian, MariaDB est installé et géré via APT (apt update, apt install mariadb-server pour l'installation). Son fichier de configuration principal se trouve dans /etc/mysql/mariadb.conf.d/, et le service est contrôlable avec systemctl (systemctl start mariadb pour démarrer). Il prend en charge les connexions sécurisées via SSL/TLS et propose des fonctionnalités avancées comme le clustering (Galera Cluster) et les moteurs de stockage optimisés (InnoDB, Aria). Compatible

avec MySQL, MariaDB est un choix privilégié pour les développeurs recherchant une base de données performante, évolutive et adaptée aux environnements cloud et web.

1. **Installation de MariaDB** :

```
# apt-get install mariadb-server -y
```

2. **Sécurisation de MariaDB**

```
# mysql_secure_installation
```

Répondez **"Y"** à toutes les questions.

En ce qui concerne le mot de passe créé, il s'agit du compte **root** de **MariaDB**. Assurez-vous de bien le conserver, car on va avoir besoin plus tard.

3. **Vérification de MariaDB** :

```
root@Srv-glpi:/home/sysbot# systemctl status mariadb
● mariadb.service - MariaDB 10.11.6 database server
     Loaded: loaded (/lib/systemd/system/mariadb.service; enabled; preset: enab>
     Active: active (running) since Wed 2025-03-05 08:25:17 CET; 27min ago
       Docs: man:mariadbd(8)
             https://mariadb.com/kb/en/library/systemd/
    Process: 515 ExecStartPre=/usr/bin/install -m 755 -o mysql -g root -d /var/>
    Process: 558 ExecStartPre=/bin/sh -c systemctl unset-environment _WSREP_STA>
    Process: 598 ExecStartPre=/bin/sh -c [ ! -e /usr/bin/galera_recovery ] && V>
    Process: 971 ExecStartPost=/bin/sh -c systemctl unset-environment _WSREP_ST>
    Process: 974 ExecStartPost=/etc/mysql/debian-start (code=exited, status=0/S>
   Main PID: 754 (mariadbd)
     Status: "Taking your SQL requests now..."
      Tasks: 9 (limit: 2273)
     Memory: 307.9M
        CPU: 5.745s
     CGroup: /system.slice/mariadb.service
             └─754 /usr/sbin/mariadbd
```

FIGURE 18

1. **Création de la base de données GLPI** dans MariaDB :

```
# mysql -u root -p
```

Lorsque le mot de passe vous est demandé, entrez celui que vous avez précédemment conservé

```
MariaDB [(none)]> create database glpidrb;

MariaDB [(none)]> grant all privileges on glpidrb.* to glpiuser@localhost identified by "Respons11 !@";

MariaDB [(none)]> quit
```

Installation de GLPI

Rappel :

GLPI, lancé en 2003, est une solution open-source de gestion des services informatiques (ITSM) utilisée pour l'inventaire, le suivi et la gestion des tickets d'assistance. Il repose sur une architecture web basée sur PHP et MySQL/MariaDB, fonctionnant avec des serveurs comme Apache2 ou Nginx. Sous Debian, GLPI est installé via APT ou manuellement en téléchargeant les fichiers sources, et nécessite l'installation de PHP et ses extensions (php-mysql, php-gd, php-curl). Sa configuration se trouve dans /var/www/html/glpi/ et ses paramètres sont ajustables via config_db.php. GLPI offre un tableau de bord complet pour la gestion des équipements, utilisateurs et interventions, avec des fonctionnalités avancées comme la synchronisation LDAP et l'intégration avec OCS Inventory. Grâce à son extensibilité via des plugins et son interface web intuitive, GLPI est une solution idéale pour les services IT cherchant à optimiser leur gestion des actifs et du support technique.

L'installation de GLPI est très rapide et se fait en deux étapes.

La première étape consiste à récupérer les paquets GLPI depuis le serveur miroir via la ligne de commande.

- Pour ce faire, entrez les trois commandes suivantes :

```
# cd /usr/src/

# wget https://github.com/glpi
project/glpi/releases/download/10.0.15/glpi-10.0.15.tgz

# tar -xvzf glpi-10.0.15.tgz -C /var/www/html
```

Ensuite, une fois les paquets téléchargés et décompressés, nous attribuons les droits nécessaires au serveur **LAMP** pour agir sur les fichiers. Vous pourrez alors poursuivre avec l'installation graphique. Pour ce faire, entrez la commande suivante :

```
#chown -R www-data /var/www/html/glpi/
```

Tout est en place ! Maintenant, ouvrez votre navigateur et entrez l'adresse suivante :

```
http://192.168.20.3/glpi
```

Et nous voici dans la page d'acceuil de GLPI

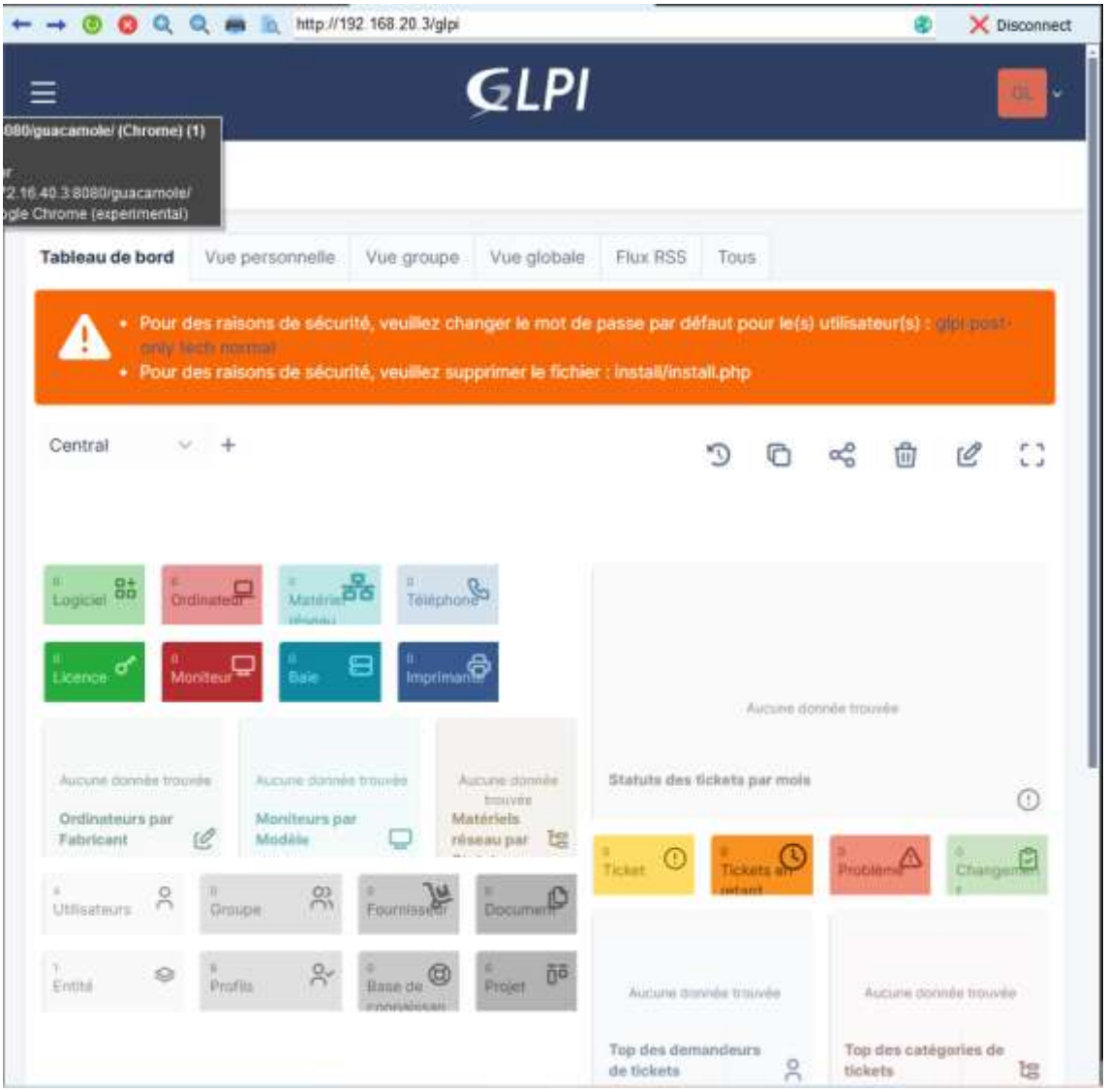


FIGURE 19

6 CONFIGURATION DU SERVICE LDAP POUR LA CONNEXION AVEC AD

6.1 INTEGRATION DE GLPI AVEC ACTIVE DIRECTORY

Connexion de GLPI au serveur AD via **LDAP**

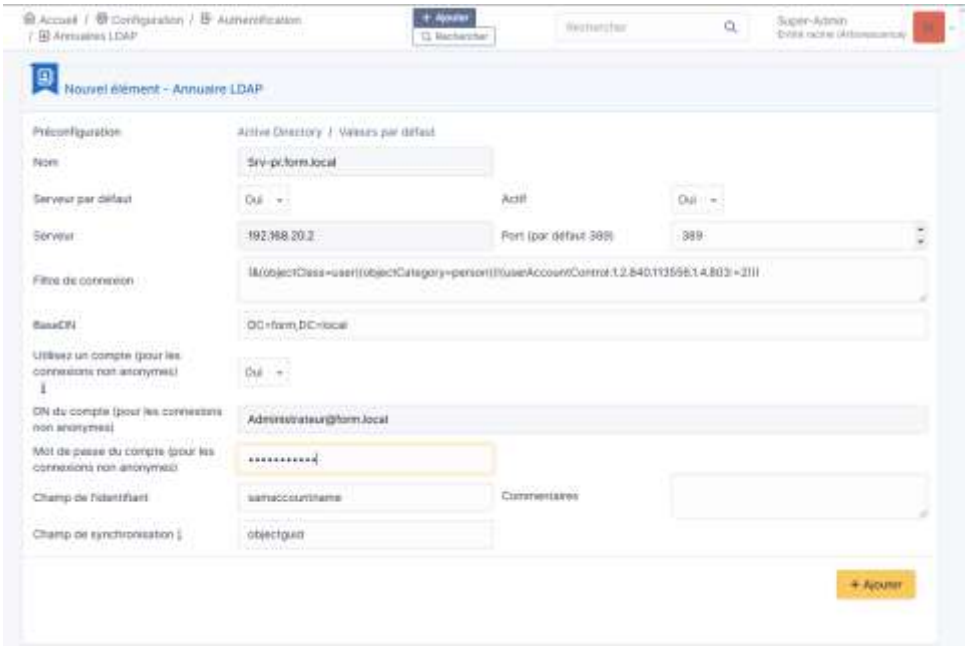


FIGURE 20

Pour permettre aux utilisateurs de se connecter à GLPI avec leurs identifiants Active Directory, il faut **configurer l'authentification LDAP**. Cette capture montre la configuration du serveur LDAP dans GLPI.

1. Vérification de la connexion de GLPI au serveur AD via **LDAP**

GLPI va alors lancer un **test de connexion LDAP** et afficher le résultat indiquant si la connexion à l'annuaire a réussi ou non

FIGURE 21

2 Synchronisation automatique des utilisateurs

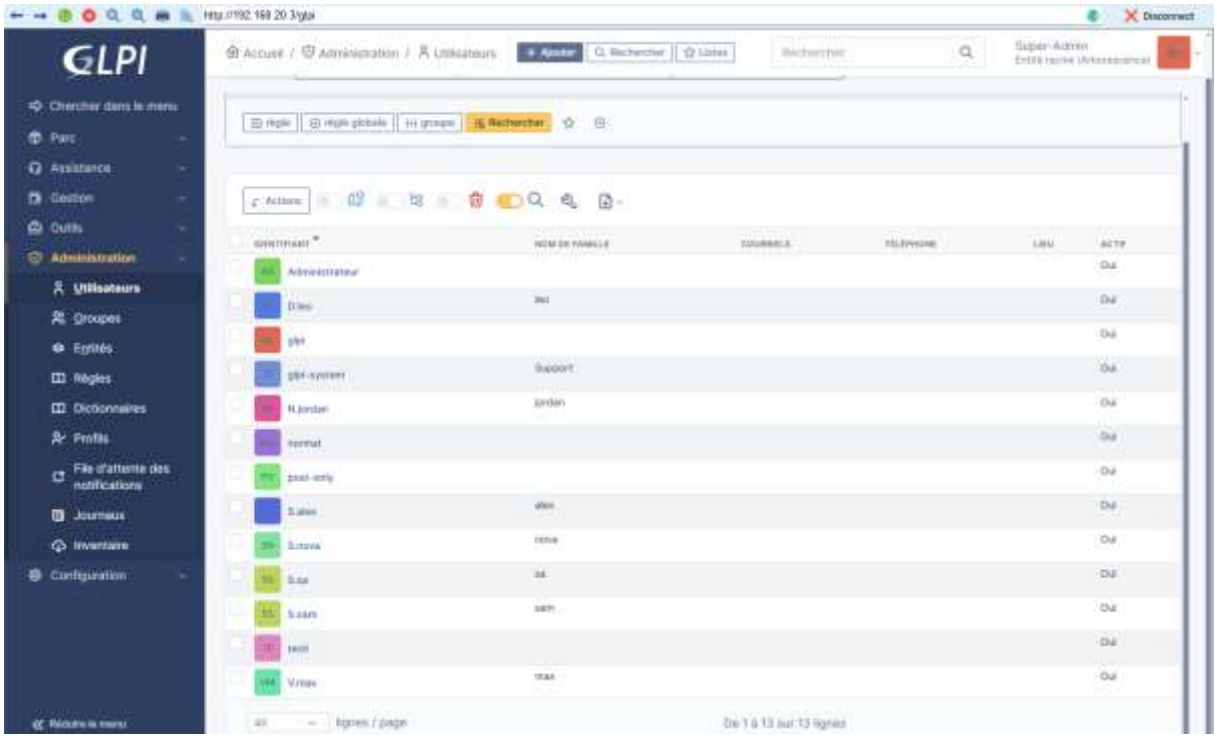


FIGURE 22

3 Attribution des permissions et rôles en fonction des groupes AD

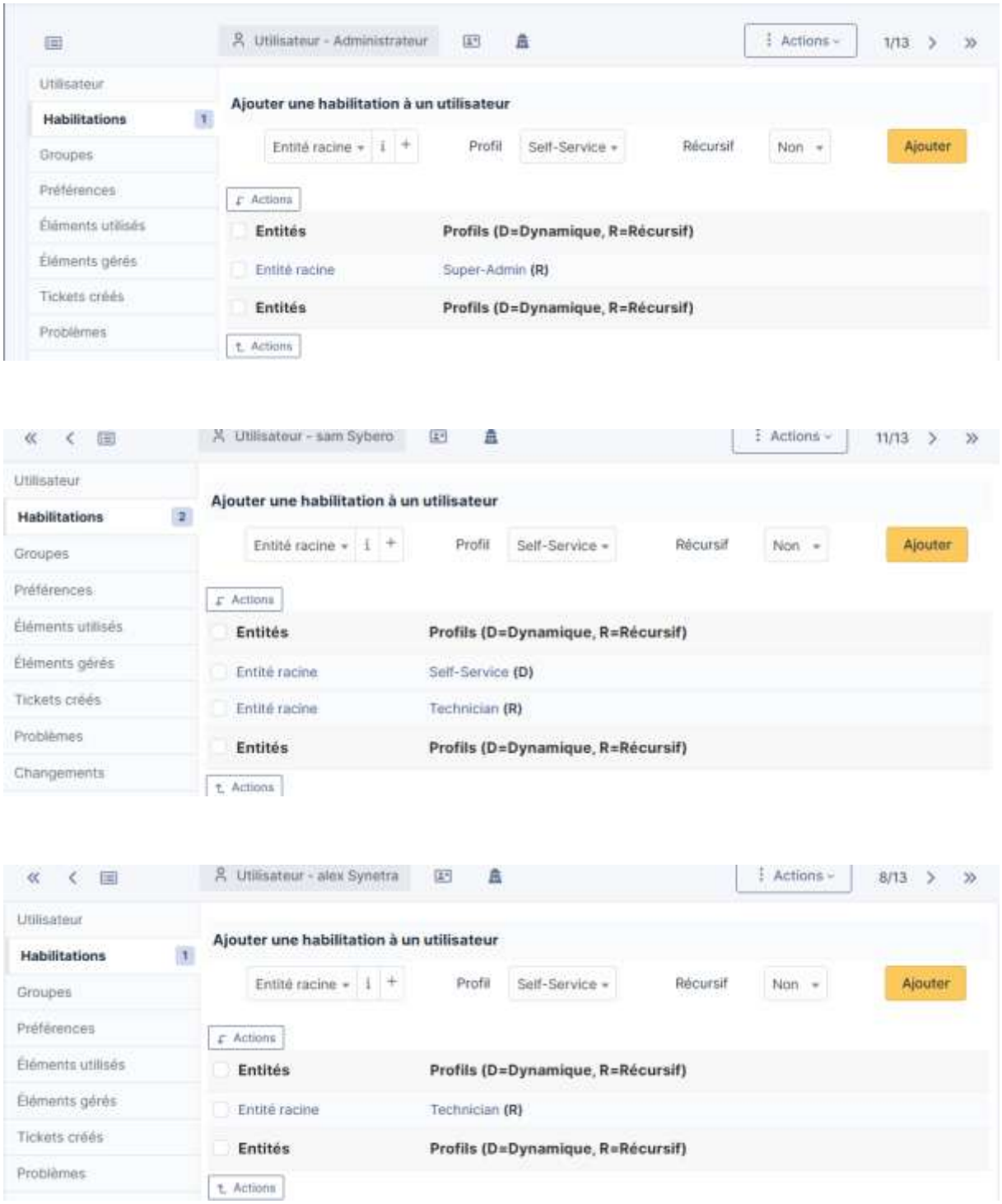


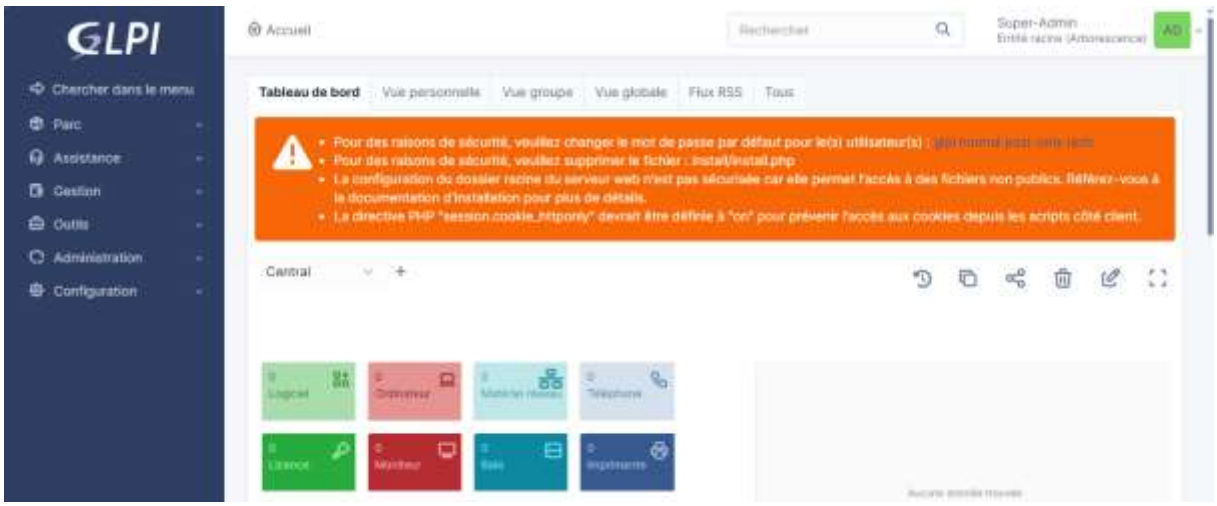
FIGURE 23

6.2 TESTS ET VALIDATION

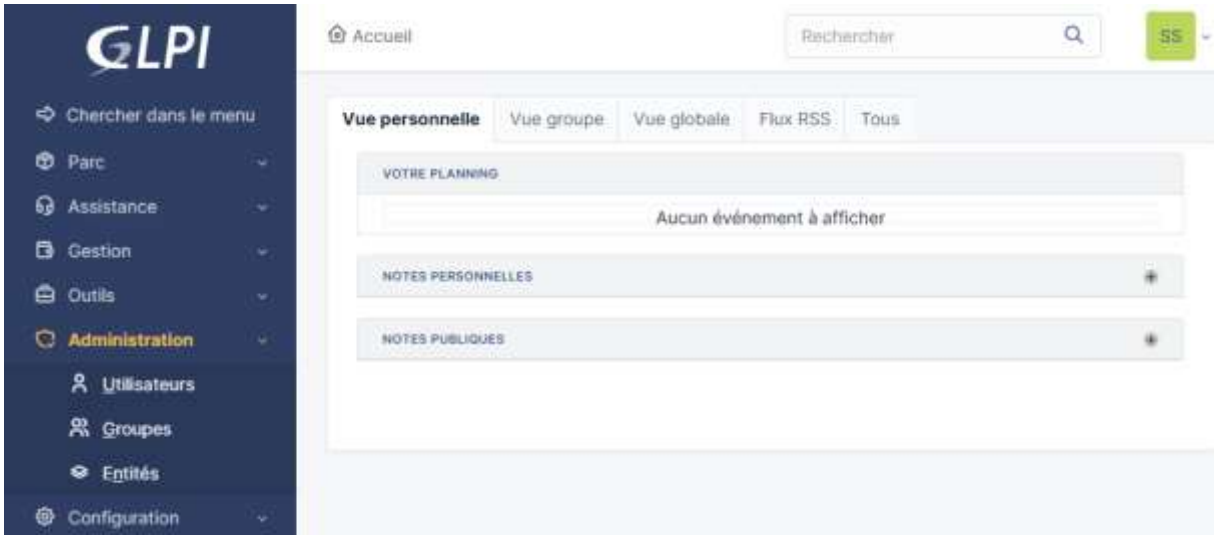
Tests effectués :

Vérification de l'authentification des utilisateurs AD dans GLPI

- Utilisateur AD avec le profil super-Admin



- Utilisateur AD avec le profil Technicien



❖ Utilisateur AD avec le profil Self-service

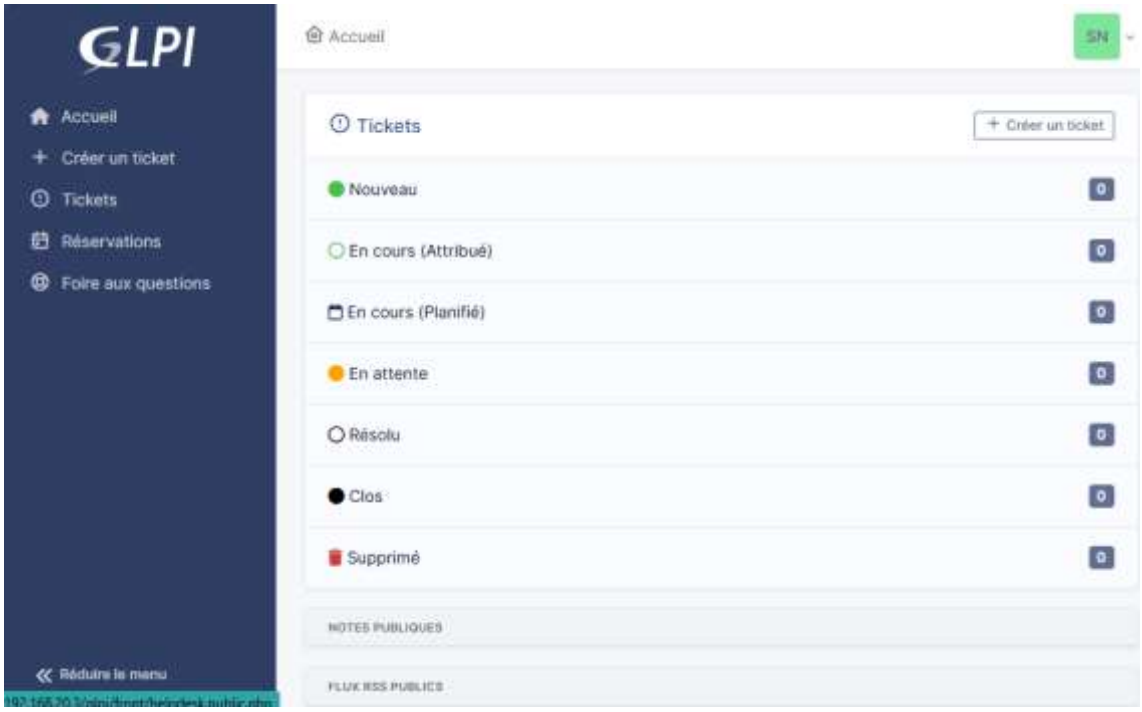
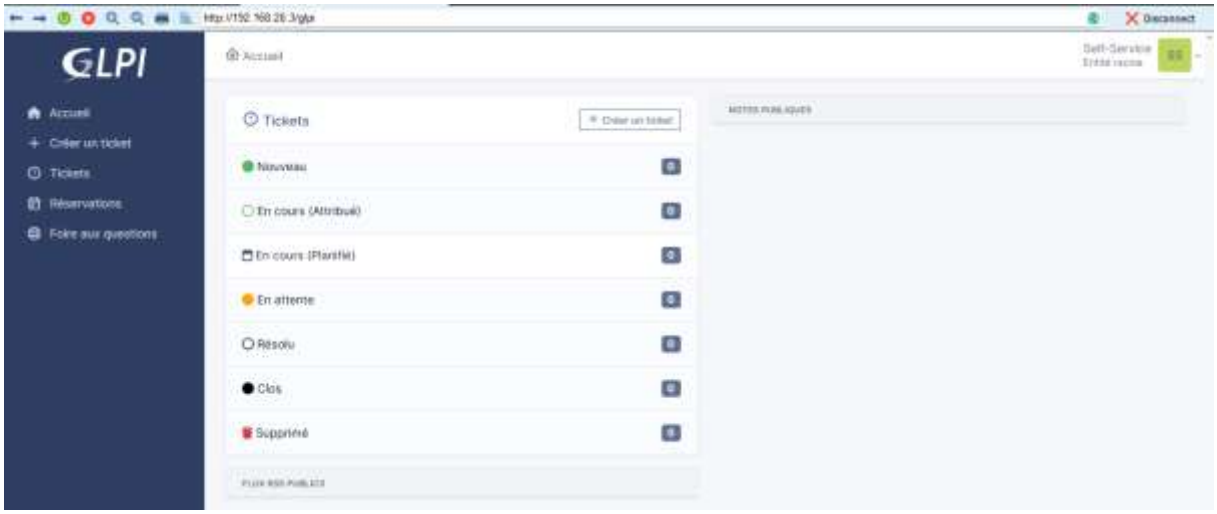
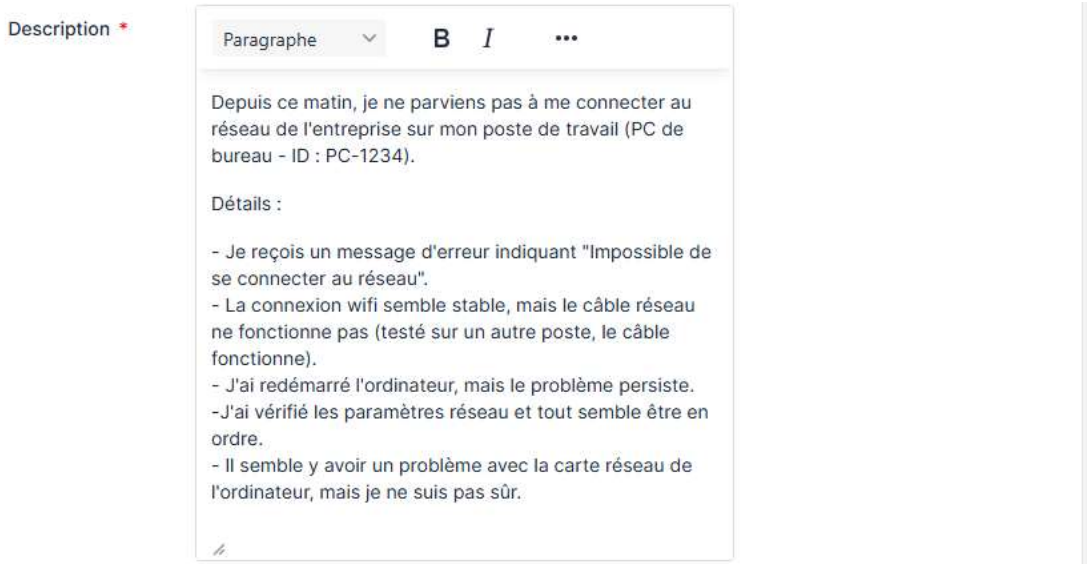


FIGURE 24

L'authentification est réussie ! L'utilisateur **Administrateur** a pu se connecter avec ses identifiants du compte **Active Directory** en utilisant le rôle **"Super-admin"** , L'utilisateur **SYBERO Sam** a pu se connecter avec son profil **Active Directory** avec le rôle **"Technician",** L'utilisateur **SECURIA nova** a pu se connecter avec son compte **AD** et a automatiquement hérité du rôle **"Self-service"** dans **GLPI**.

❖ Test d’attribution automatique des tickets





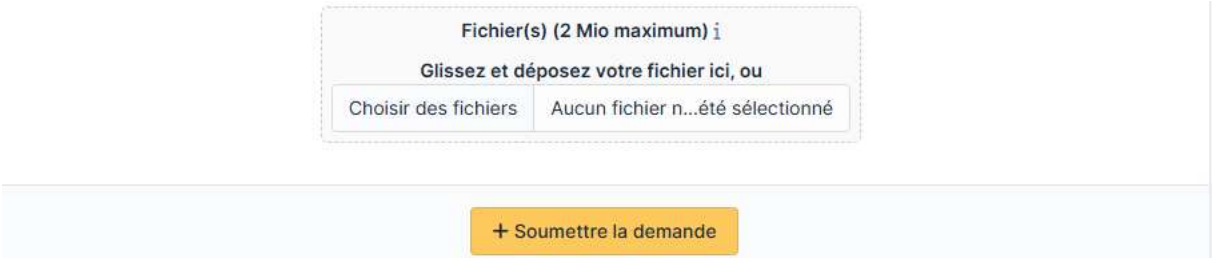


FIGURE 25

Ces captures montrent qu'un ticket a été attribué au groupe **Support IT** en fonction de la configuration LDAP.

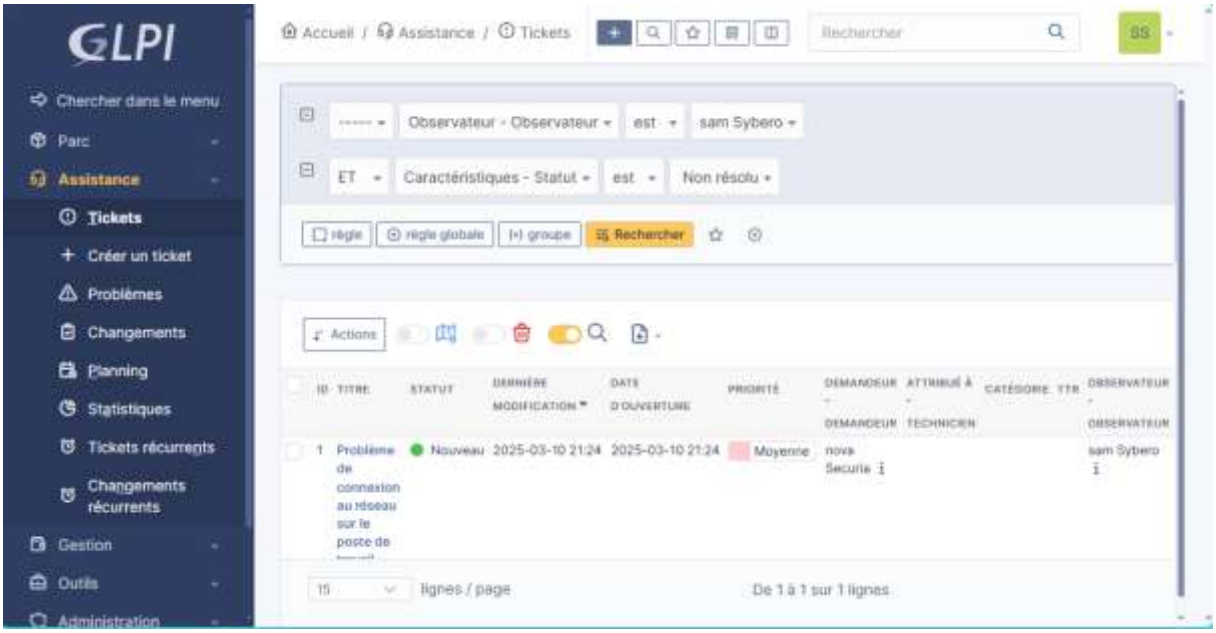


FIGURE 26

❖ Vérification des droits et restrictions d'accès

➢ Profil Technicien

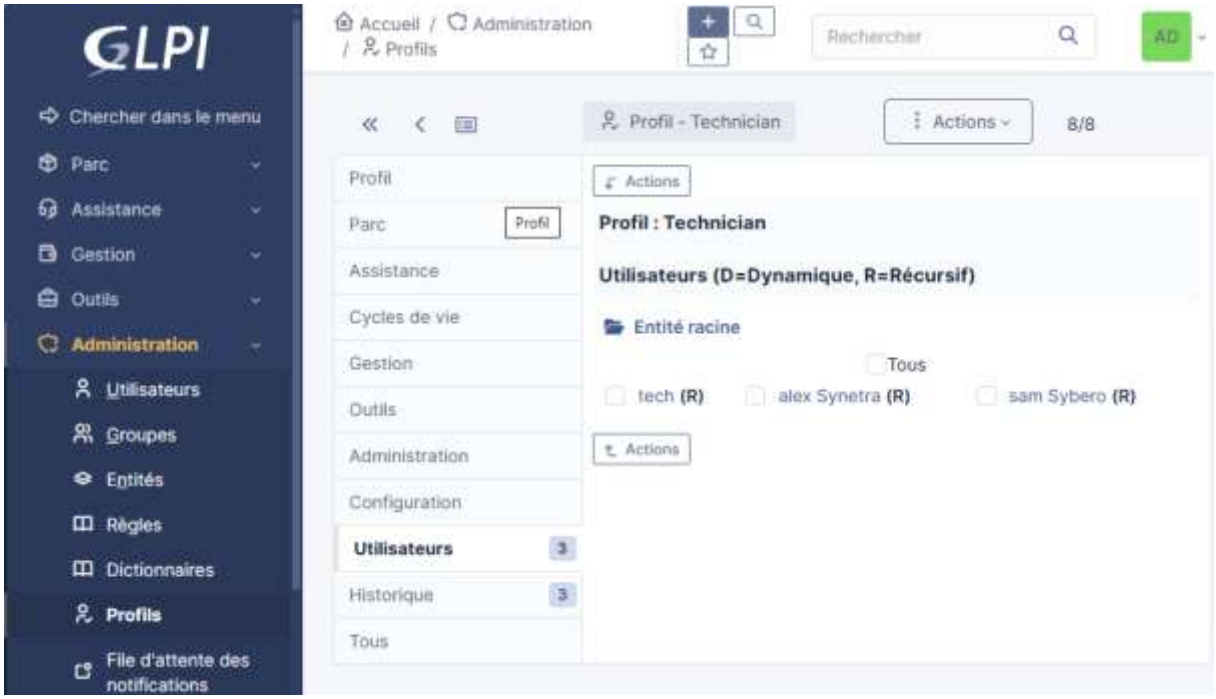


FIGURE 27

- Profil Self-service

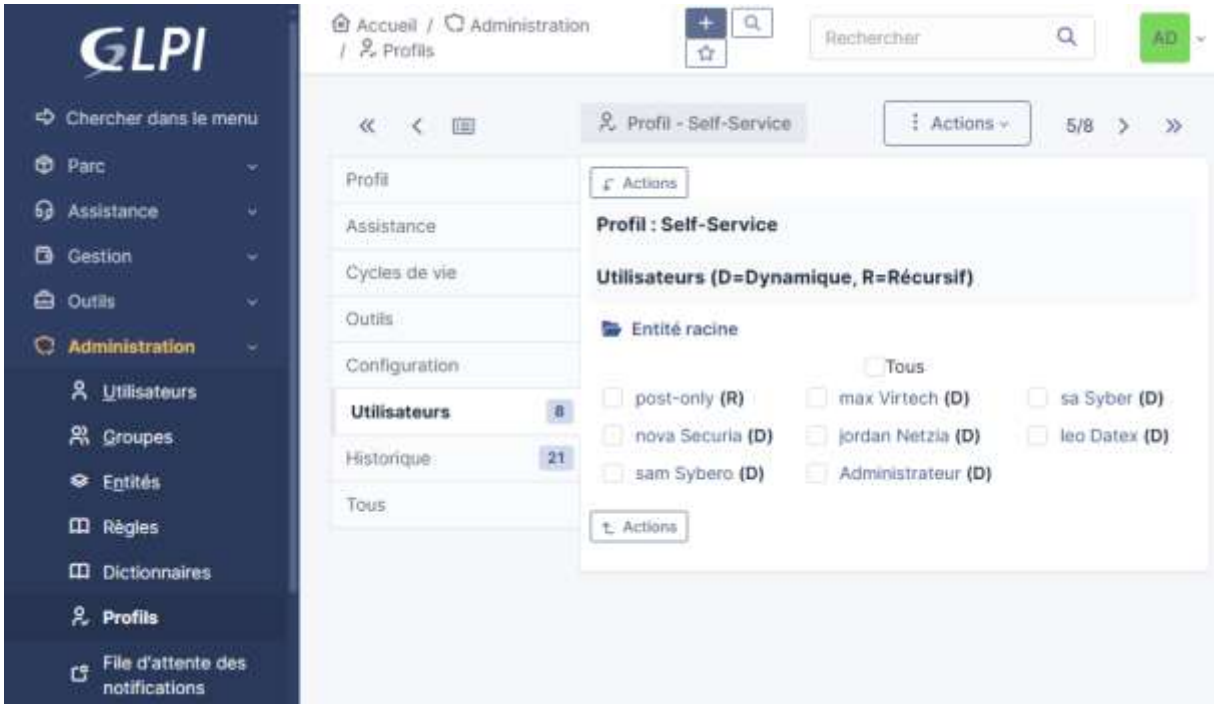


FIGURE 28

- Profil Super-Admin

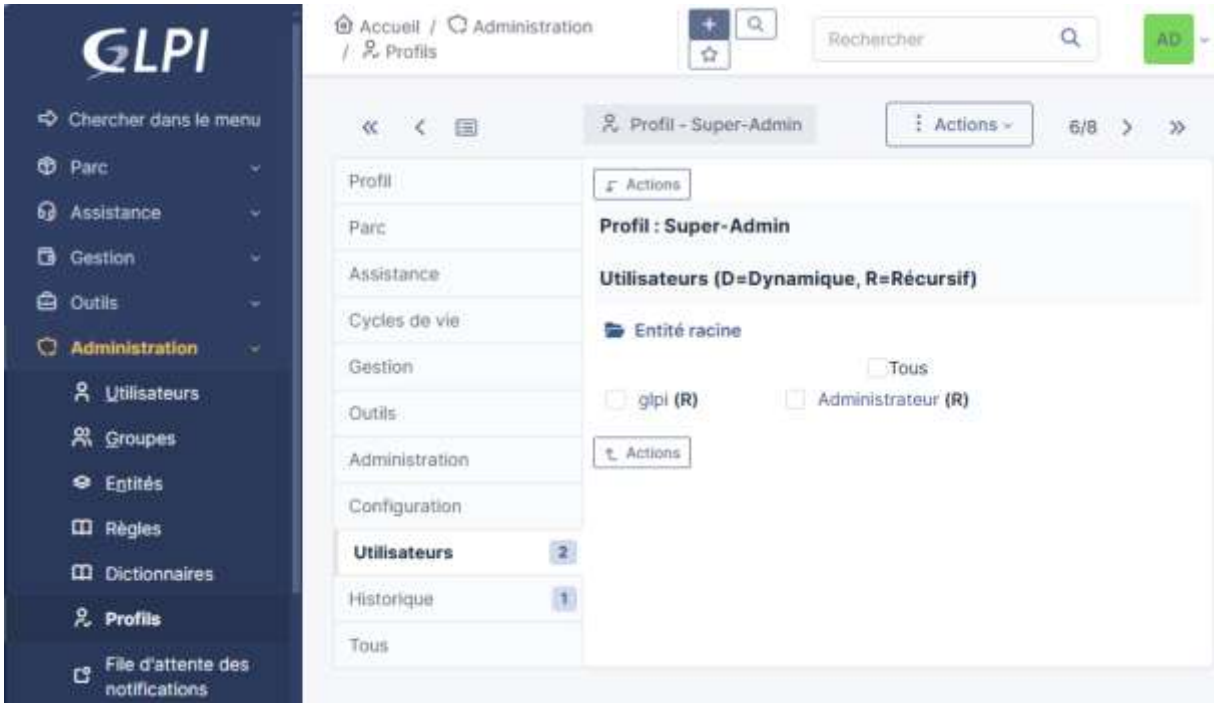


FIGURE 29

GLPI gère plusieurs **profils utilisateurs**, chacun ayant des **droits spécifiques**. Ces capture montre l'accès **différent** selon les profils :

7. PERSPECTIVES D'EVOLUTIONS

7.1 SUPERVISION ET REPORTING

- Supervision et reporting avancé avec Nagios, Grafana, Centreon ou Zabbix
- Création de tableaux de bord analytiques pour suivre les performances du support IT
- Génération de rapports détaillés sur les demandes et les interventions
- Mise en place d'un environnement de secours avec un serveur GLPI en haute disponibilité
- Sécurisation renforcée avec des pare-feux, VPN et authentification à double facteur (2FA)

8. CONCLUSION

Ce projet a été une excellente opportunité pour mettre en pratique mes compétences en administration système, virtualisation et gestion des accès IT. L'intégration entre Active Directory et GLPI permet d'améliorer l'efficacité du support IT en centralisant les utilisateurs et automatisant la gestion des tickets.

L'un des défis majeurs a été la configuration correcte de LDAP et l'automatisation avec PowerShell, mais en explorant les bonnes pratiques et en réalisant plusieurs tests, j'ai pu surmonter ces difficultés.

Pour aller plus loin, il serait intéressant d'ajouter un système de supervision (Nagios, Zabbix, Grafana), une gestion des accès plus sécurisée (2FA) et une meilleure automatisation des rôles utilisateurs dans GLPI.