

**Réalisé par :**

QUENTIN GAVILAN

M1 IRS-Groupe 1



**12 Avril 2024**

**Encadré par :**  
Synoptik Labs

# INTRODUCTION

Le présent rapport technique détaille la mise en place d'une infrastructure Active Directory et de ses Rôles afin de simuler et de créer un environnement d'entreprise d'après un Travaux Pratique réalisé en classe. Ce projet a été conduit en utilisant un hyperviseur basé sur KVM avec une interface de gestion Proxmox, connecté au réseau interne de mon domicile. Pour une gestion à distance efficace et sécurisée, une connexion VPN depuis la classe vers un serveur Debian à domicile a été établie, permettant de gérer les machines virtuelles (VM). Cette solution a été adoptée pour éviter les problèmes de performance qui auraient pu survenir avec l'utilisation d'un ordinateur portable. Le projet consistait à construire et configurer quatre machines virtuelles (VM) sous Windows Server 2022 sans interface graphique, avec une attention particulière portée à l'installation et à la configuration de l'Active Directory, la gestion du DNS, la mise en œuvre des politiques de groupe (GPO), des services de certificats et d'un équilibrage de charge réseau sur un serveur Web.

## MISE EN PLACE

### Configuration des Serveurs et plan IPAM

Je vais configurer trois serveurs Windows Server 2022 sans interface graphique. Le premier, en tant que contrôleur de domaine, gèrera la forêt, les GPO, les utilisateurs et le DNS. Le second, dédié à l'autorité de certification et au serveur Web, vise à séparer les services critiques tier0. Le dernier servira à l'équilibrage de charge Web.

Ces serveurs, nommés BOBA, JANGO et FETT, avec respectivement les adresses IP suivante 192.168.1.200, 192.168.1.201 et 192.168.1.202, avec une gateway à 192.168.1.254, tous configurés en /24. Les comptes d'essai seront ceux de Luke et Gavie, et le compte Administrateur restera actif pendant la simulation.

Un client Windows 11 Pro 23H2 appelé Client1 sera installé pour gérer à distance les serveurs via RSAT-Tools, depuis ma machine portable en classe, qui se connectera au réseau des serveurs via VPN. Cette configuration complexe est sous ma responsabilité.

Nous allons maintenant nous concentrer sur la configuration technique de ces serveurs et du client.

## Mise en place des Serveurs et Client

J'installe l'ISO Windows selon les étapes prescrites, en optant pour la version Standard sans interface graphique et en définissant un mot de passe pour l'administrateur local. Sur la console Sconfig, je configure ensuite les noms et adresses IP des trois serveurs, conformément au plan IPAM établi. Puis, je m'attaque à la création d'un client Windows 11 Pro.

Pour l'installation de Windows 11 Pro sur ma machine locale virtualisée (sans Proxmox), un premier problème apparaît Windows nous oblige à renseigner un compte Microsoft pour créer un utilisateur, ce qui ne me convient pas. Pour contourner cette problématique, on utilise le raccourci Shift + F10 pour lancer le cmd et on exécute la commande suivante : oobe/bypassnro. L'option bypassnro est conçue pour ignorer les vérifications de ressources réseau durant le processus OOB de l'installation. On redémarre ensuite l'installation et on débranche le câble réseau de la VM pour faire apparaître l'option de bypass et entrer un compte local.

Après l'installation du client, je renomme la machine et vérifie sa connexion Internet pour installer les RSAT, nécessaires au contrôle à distance des serveurs. J'utilise les commandes PowerShell suivantes pour l'installation :

```
Get-WindowsCapability -Name *DNS* -Online | Add-WindowsCapability -Online  
Get-WindowsCapability -Name *groupe* -Online | Add-WindowsCapability -Online  
Get-WindowsCapability -Name *équilibre* -Online | Add-WindowsCapability -Online  
Get-WindowsCapability -Name *Active* -Online | Add-WindowsCapability -Online
```

Une fois cette installation terminée (~1H00), je peux lancer l'utilitaire Server Manager afin de gérer les serveurs à distance. Avant cela, j'ai configuré le VPN pour connecter Client1 au réseau de mon domicile, en installant SecurePointSSL.

En l'absence d'un domaine configuré, le système d'authentification Kerberos n'est pas encore opérationnel sur les serveurs et les clients ce qui nous empêche de le gérer depuis l'interface. Pour remédier à cela, il est nécessaire d'ajouter le serveur BOBA, qui sera le futur contrôleur de domaine, à la liste des TrustedHosts du client. Pour ce faire, les commandes PowerShell suivantes sont utilisées :

```
Set-Item WSMAN:\localhost\Client\TrustedHosts-Value BOBA.local
```

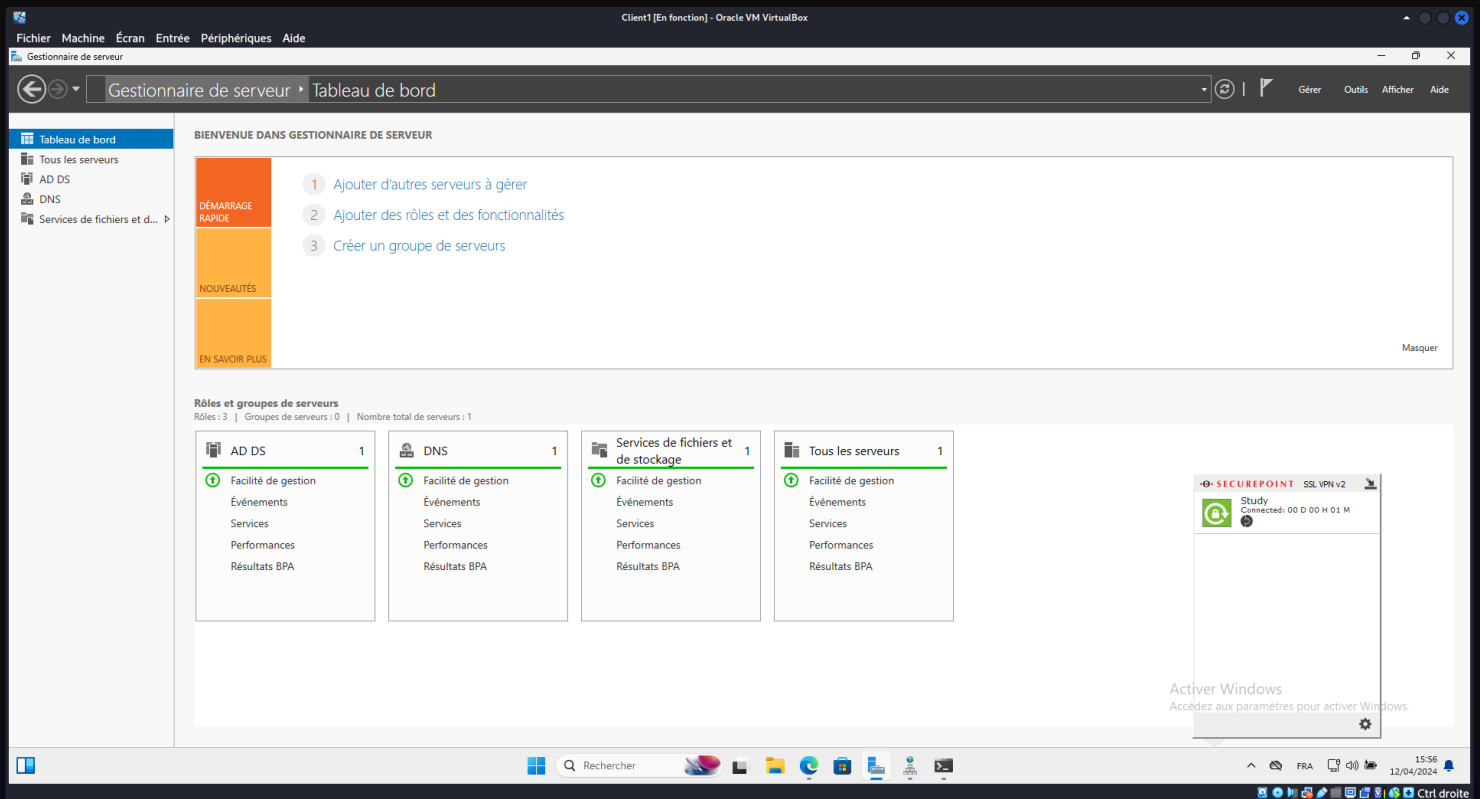
Pour autoriser WinRM et le contrôle à distance :

```
Enable-PSRemoting-Force
```

Pour intégrer le serveur BOBA dans le système de gestion, je passe par l'interface Server Manager et sélectionne l'option "Ajouter un serveur". Je localise BOBA en utilisant son adresse IP, 192.168.1.200, via l'onglet DNS. Après avoir sélectionné le serveur, je confirme par OK. Dans la section "Tous les serveurs", un clic droit sur BOBA me permet de choisir "Gérer en tant que", où je saisis les identifiants de sécurité locaux. Cela me permet d'installer le rôle Active Directory Domain Services (AD-DS) sur BOBA.

Une notification s'affiche ensuite pour proposer la promotion de BOBA en tant que contrôleur de domaine. Dans l'assistant de configuration, je choisis de créer une nouvelle forêt, que je nomme ad.secret.local, et j'ajuste le nom NetBIOS en SECRET, remplaçant ainsi le préfixe par défaut "ad". Après le redémarrage du serveur, le domaine est établi et le service DNS est configuré prêt à accueillir le premier client.

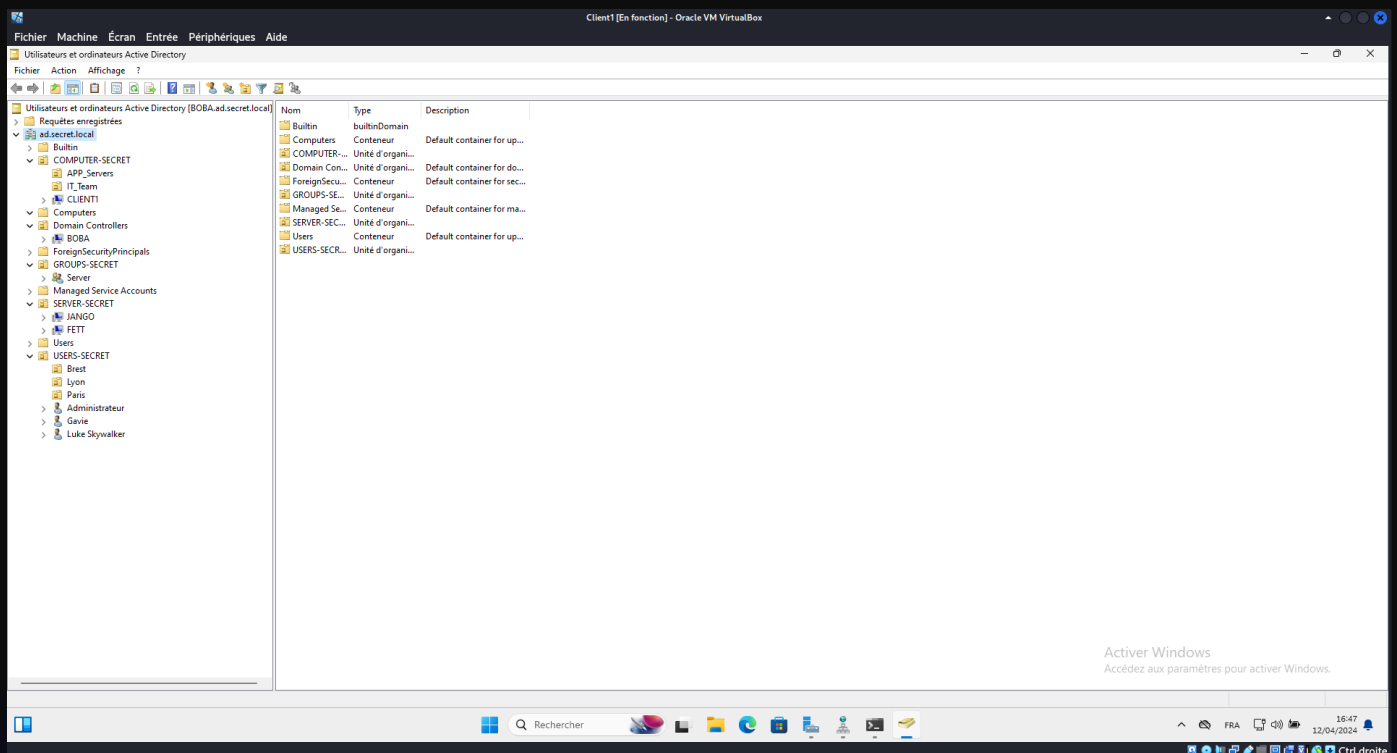
Avant cela on ajoute une zone inverse au service DNS afin de résoudre le nom de l'hôte grâce à l'IP. Zone inverse -> Ajouter Zone -> 192.168.1 -> Nouveau PTR -> boba 192.168.1.200  
Notre Server Manager ressemble donc à cela :



Je veux maintenant intégrer le client dans le domaine. Pour y parvenir, j'ajoute l'adresse IPv4 du serveur DNS aux paramètres réseau du client (Client1), ce qui est nécessaire pour que le client puisse résoudre le nom de domaine lors de l'ajout. Je me dirige ensuite vers le panneau de configuration pour ajouter le domaine aux paramètres système du client. Cependant, je rencontre un obstacle : une erreur signalant que le nom de domaine spécifié semble inexistant ou non créé. Il semble que le client ne parvienne pas à résoudre le nom de domaine à distance, même après avoir correctement configuré l'IPv4 en définissant 192.168.1.200 comme DNS. Par défaut, Windows ne reconnaît pas cette adresse comme DNS primaire, optant pour une autre.

Après une pause et une réévaluation de la situation, je découvre que le serveur AD DC, lors d'un test de ping sur ad.secret.local, utilise une adresse IPv6. J'ajuste donc les paramètres IPv6 du DNS sur l'interface du client, ce qui résout finalement le problème de résolution de nom de domaine.

J'ajoute donc le client et les serveurs Jango et Fett au domaine et créer les OU, USER-SECRET / PARIS & BREST & LYON, COMPUTER-SECRET / IT\_TEAMS & APP\_SERVER-SECRET, GROUPS-SECRET tout en plaçant les utilisateurs et les machines server et client dans les bonnes OU.

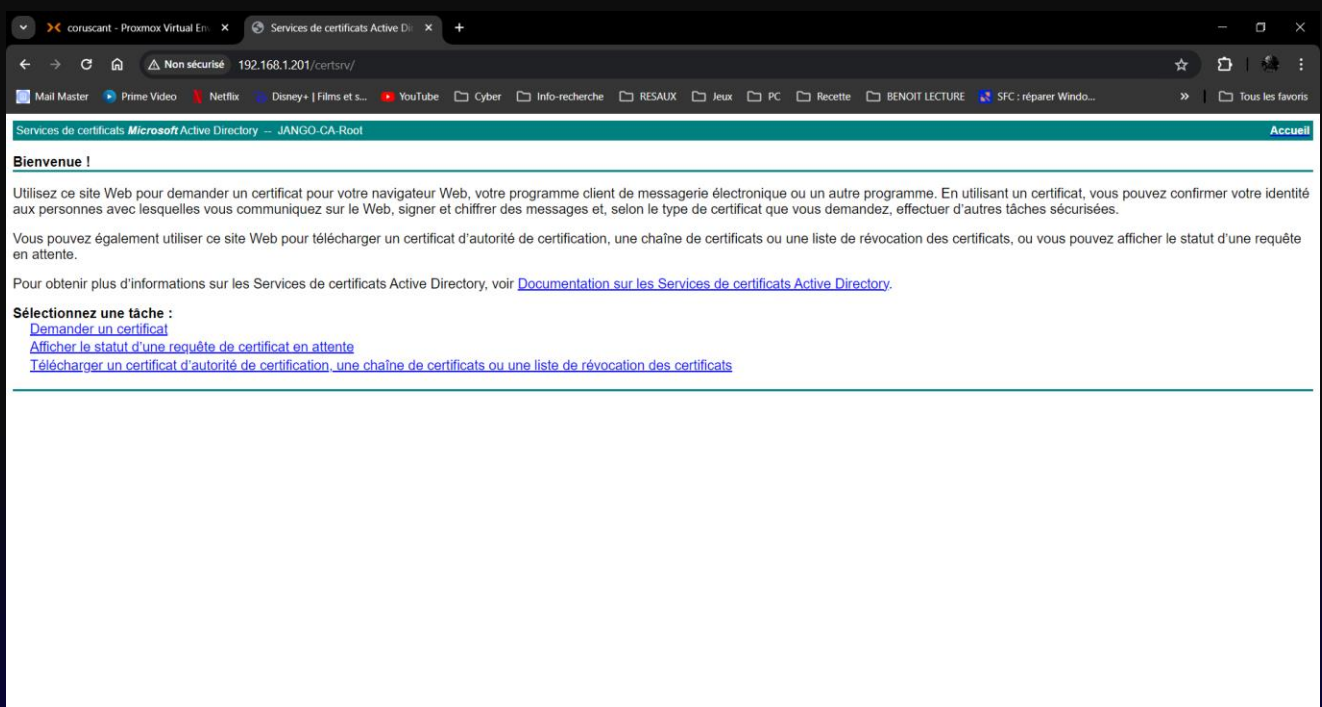


## Configuration du rôle AD-CS

Sur le serveur Jango, nous mettons en place le rôle AD-CS. Cette fonctionnalité clé dans une infrastructure d'entreprise permet de délivrer et de signer des certificats via une autorité de certification racine (CA ROOT). Cela renforce significativement la sécurité du réseau en validant l'intégrité des fichiers et en sécurisant les connexions SSL/TLS des sites web et les communications dans l'entreprise. Les bonnes pratiques de sécurité nous imposent plusieurs niveaux de CA dont un niveau racine offline, qui émet un certificat racine pour tous les serveurs subordonnés. Ces derniers, opérationnels en ligne, distribuent des certificats aux clients de l'Active Directory. La CA Root offline permet de protéger son intégrité elle n'est pas connectée à internet et est un serveur dit possédant un clavier propre soit n'étant pas sujet à la compromission

Nous procédons à l'installation des outils de l'Autorité de Certification (CA) sur le serveur et configurons les services nécessaires. Cela comprend en plus de l'autorité de certification, l'installation de la racine de la CA via le web. Pour le chiffrement, nous sélectionnons RSA avec une clé de 4096 bits pour garantir une bonne compatibilité et sécurité puis on finit la configuration en cliquant sur SUIVANT.

Ensuite, nous accedons le service de demande de certificat sur le serveur web à l'adresse 192.168.1.200/certsrv. Pour tester la fonctionnalité, nous effectuons une demande de certificat depuis le web dont la page ressemblant à ceci :





Afin de tester notre CA nous préparons une requête de certificat en créant un fichier .req sous linux, en commençant par la création de la clé client :

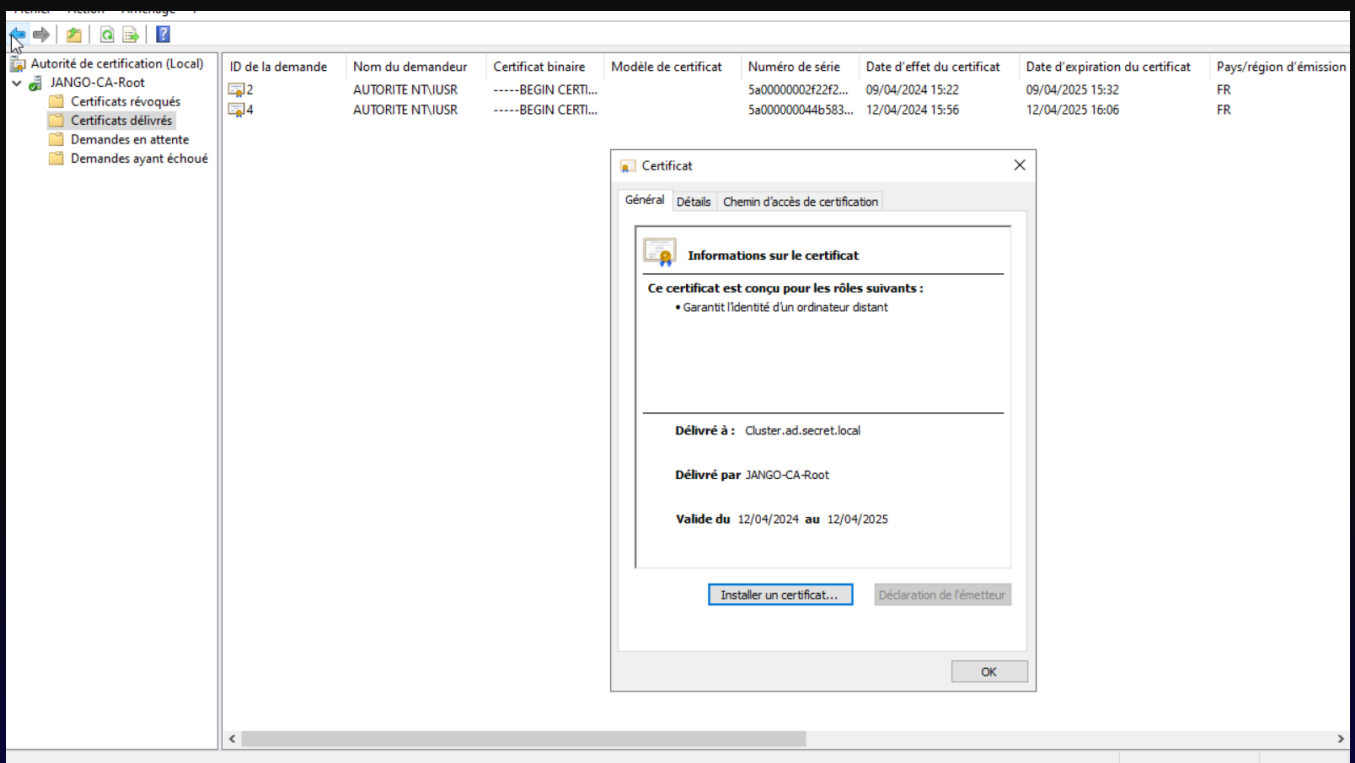
Pour générer une requête de certificat, j'utilise d'abord la commande `openssl req -key keybase.key`.

Ensuite, je procède à la création de la requête proprement dite avec `openssl req -new -key keybase.key -out request.req`.

Cette étape implique de remplir soigneusement divers champs pour s'assurer qu'ils correspondent aux spécificités de la machine cliente. Il s'agit notamment de renseigner le nom commun (Common Name), le pays, la région, et le nom de l'entreprise.

Dans le cadre d'un service web, il est crucial de préciser les noms DNS associés au site et l'adresse IP utilisée pour la connexion dans la demande de certificat. Ces informations sont essentielles pour la validation du certificat vis-à-vis du site web concerné et pour le bon fonctionnement du protocole SSL, garant de la sécurité des communications

Le contenu de ce fichier .req est ensuite inséré dans le formulaire de demande de certificat public sur <http://192.168.1.201/certsrv>, que nous remplissons et envoyons. Après cela, nous nous rendons sur le serveur d'autorité de certification pour valider et délivrer le certificat en attente, achevant ainsi la mise en place d'une Infrastructure à Clés Publiques (PKI) opérationnelle.



## Configuration des gestions de stratégie de groupe

Nous entamons la création des Group Policy Objects (GPO) afin d'améliorer la sécurité des utilisateurs et ordinateurs et de personnalisé son parc informatique.

La première GPO, nommée Policy-Computer-Password, est destinée à la gestion des mots de passe : exigence de 12 caractères minimum, interdire la possibilité de créer le même mot de passe que les 10 dernier, expiration au bout de 180 jours, et un verrouillage de 15 minutes après trois tentatives erronées. Elle est associée à l'OU COMPUTER-SECRET et on désactive la partie utilisateur de la gpo afin d'améliorer la performance lors du démarrage des ordinateurs client, ceci n'a pas d'impact sur cette petite infra mais plus tard avec beaucoup de gpo cela augmente l'optimisation.

La seconde GPO, Policy-User-Service-Desktop vise à améliorer l'expérience utilisateur sur le bureau, avec un fond d'écran stimulant la créativité et la prévention de suppressions accidentelles d'éléments importants.

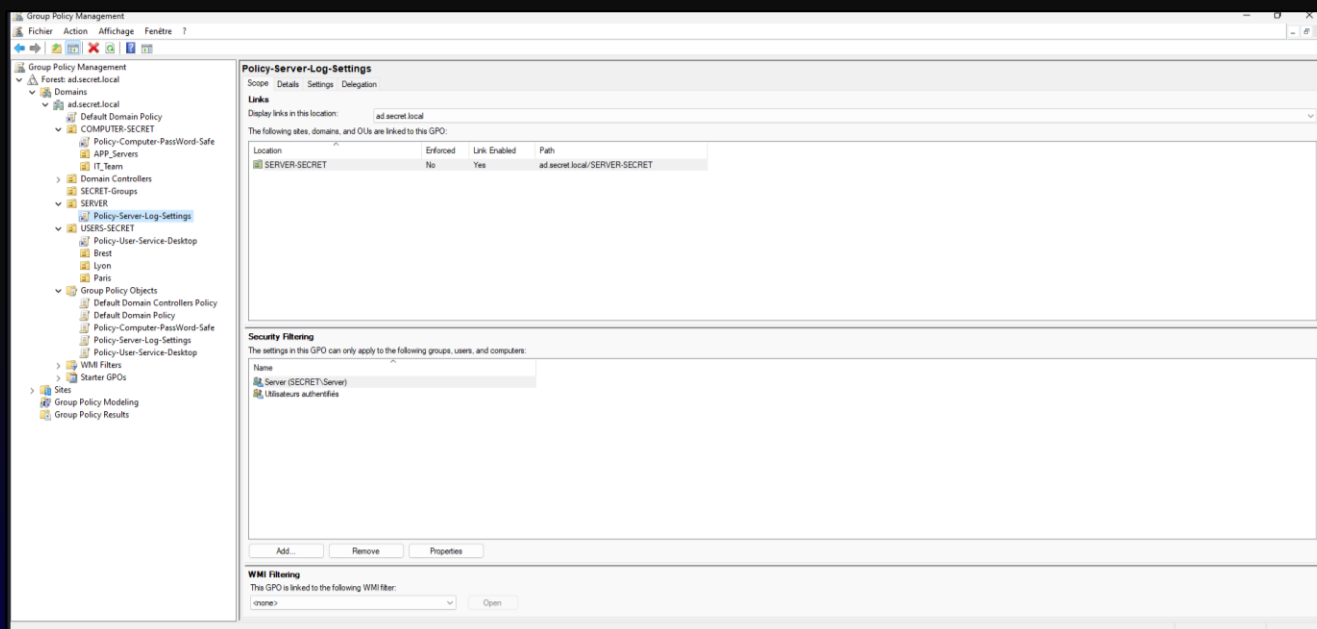
La GPO Policy-Server-Log-Setting est conçue pour contrôler la taille des journaux de logs, limitant les journaux d'application et système à 10 Mo et les journaux de sécurité à 20 Mo, ce qui aide à prévenir la surcharge du disque et assure l'intégrité des logs.

Une fois les GPO attribués dans les bonnes OU on doit vérifier leurs bonnes attributions afin d'être sûr d'avoir appliqué ce que l'on attend, pour cela on utilise 2 commandes :

```
gpupdate /force
```

```
gpresult /h Gpo.html
```

Afin de forcer l'application des GPO et de vérifier sa bonne application sur les client et serveur. Vous pouvez trouver les fichiers GPO\_User-Computer.html et GPO\_Server.html dans le dossier du rapport. Voici un screen des GPO attribués au OU :



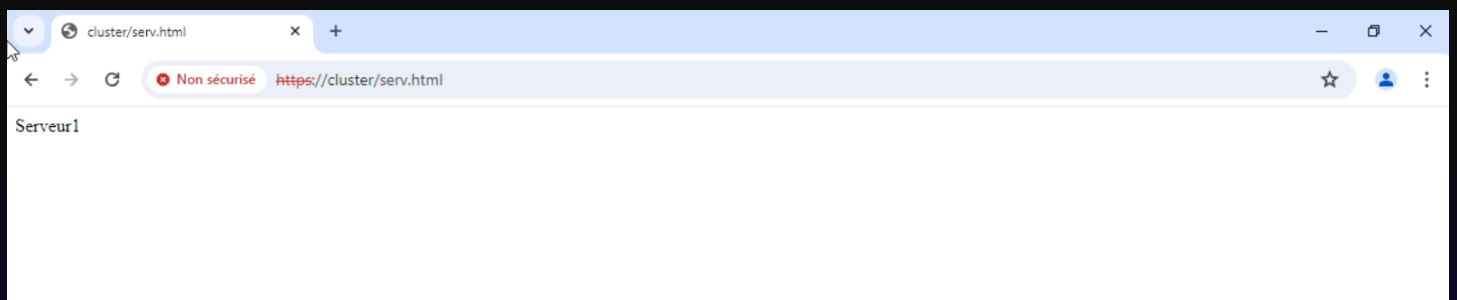


## Configuration d'équilibrage de charge

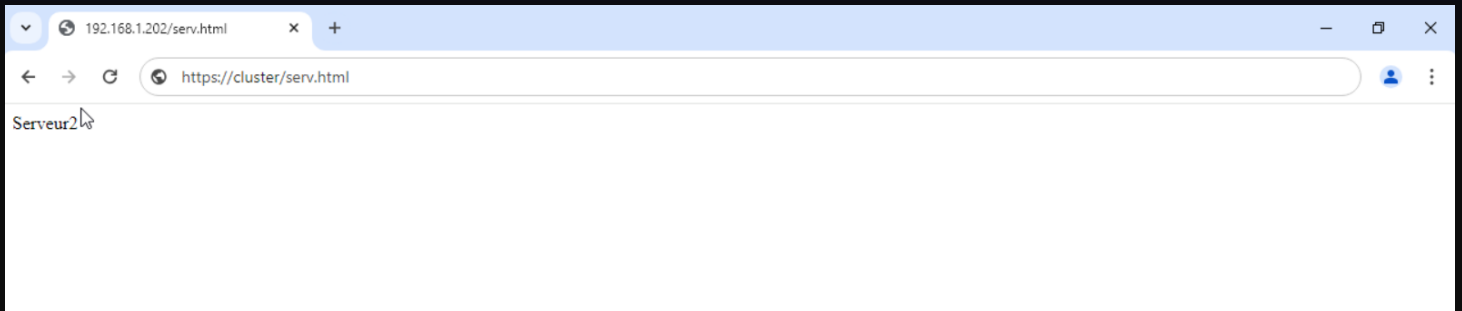
Windows a introduit un service de gestion de Load Balancing sur leur serveur Active Directory, un système qui utilise le concept de clustering d'applications via le réseau. L'objectif est d'utiliser une adresse IP unique pour un site web qui distribue les requêtes entre deux serveurs distincts. Ce dispositif augmente la disponibilité des services actifs, en l'occurrence le service web IIS que nous allons mettre à disposition sur le LAN grâce à une page web générique créée par IA.

Nous procédons à l'installation des rôles et services IIS et NLB sur les serveurs Jango et Fett et lançons le gestionnaire de configuration d'équilibrage de charge sur notre client, après avoir installé les RSAT-NLB. Nous configurons le cluster en commençant par Jango comme serveur principal, en sélectionnant sa carte réseau sur le LAN (192.168.1.201), puis nous attribuons l'IP de cluster (192.168.1.204) et ajustons les règles de port pour HTTP (80) et HTTPS (443) avec option filtrage hôte multiple Unique et les IP sont en Multicast. Après avoir établi le premier nœud, nous ajoutons Fett au cluster, utilisant des paramètres similaires pour sa carte réseau LAN. Avec les configurations complétées, nos serveurs fonctionnent en convergence, rendant notre service pleinement opérationnel. Nous pouvons maintenant passer à la mise en œuvre de la partie applicative.

Nous avons créé un fichier .html, nommé test.html, pour identifier quel serveur du cluster est actuellement sollicité. Nous plaçons un fichier différent dans le répertoire /inetpub/wwwroot/ de chacun des serveurs Jango et Fett. Chacun de ces fichiers indique respectivement Serveur1 ou Serveur2, permettant ainsi de déterminer quel serveur répond. Cette configuration utilise directement l'adresse IP du cluster qui intègre les services préalablement installés, sans besoin de configurations additionnelles. En accédant à <http://192.168.1.204/test.html> via un navigateur, la page affichée indique Serveur1 :



Faisont le test en eteignant Jango afin de vérifier que le NLB nous redirige vers Fett on aura donc une affichage differente avec ecrit Serveur2 sur la page ce qui est le cas on a donc effectivement un service d'équilibrage de charge opérationnel



Maintenant que notre autorité de certification est fonctionnelle, nous souhaitons implémenter le protocole HTTPS pour notre service web. Nous débutons par attribuer un nom de domaine, « Cluster », à notre cluster et réalisons les enregistrements A et PTR nécessaires sur notre serveur DNS. Suite à cela, nous lançons la création d'une demande de certificat via le service IIS. Nous naviguons jusqu'à l'onglet « Certificat de Serveur » de notre Serveur Web et sélectionnons « Créer une demande de certificat ». Dans le formulaire qui apparaît, nous renseignons toutes les informations requises, en utilisant le CN correspondant au DNS enregistré, soit Cluster.ad.secret.local.


Après avoir généré et copié le contenu de notre fichier .req, nous le soumettons via le service de demande de certificat, le faisons délivrer, et téléchargeons le certificat fini. Nous retournons ensuite sur IIS, complétons la procédure en cliquant sur « Terminer la demande de certificat », et installons le certificat téléchargé. Pour activer HTTPS, nous configurons les liaisons SSL de notre Serveur Web, associant le nouveau certificat.

Une fois ces étapes complétées, nous vérifions que notre site fonctionne en HTTPS, assurant ainsi une connexion sécurisée pour les utilisateurs.

# ANNEXE / PREUVE

?

×

**Propriétés du nom unique**

Indiquez les informations requises pour le certificat. Lorsque vous entrez le département ou région et la ville/localité, utilisez des noms complets et officiels, et n'employez aucune abréviation.

Nom commun :

Organisation :

Unité d'organisation :

Ville :

Département/région :

Pays/région :

Précédent


Suivant

Terminer

Annuler

?

×

**Indiquer la réponse de l'autorité de certification**

En récupérant le fichier contenant la réponse de l'autorité de certification, vous terminez le processus de demande de certificat entamé précédemment.

Nom du fichier comportant la réponse de l'autorité de certification :  ...

Nom convivial :

Sélectionnez un magasin de certificats pour le nouveau certificat :

OK

Annuler

