Configurations de votre serveur réel

Ce laboratoire doit être fait individuellement sur votre serveur réel

Objectifs

Configurer plusieurs paramètres sur le serveur réel

Travail

Vous allez explorer diverses facettes de Windows Serveur 2019 tout en configurant l'environnement afin de rendre votre travail d'administrateur plus efficace.

Les partitions pour les cours 420-C53-IN et 420-C53-VM

Démarrer la console "**Gestion des disques**" et vérifier que les partitions pour les cours 420-C53-IN et 420-C53-VM sont valides.

420-C53-IN		
420-C53-VM	**LOCAL A5.38**	**Disque1 - Partition 1**

Vous devez renommer votre ordinateur:

- Exemple pour l'ordinateur 5 sur la partition 1 du local A5.38 le nom sera **538S05P1**
- Exemple pour l'ordinateur 24 sur la partition 1 du local A5.38 le nom sera **538S24P1**

Vous devez changer le mot de passe du compte **Administrateur**.

Vous devez cocher "**Le mot de passe n'expire jamais**" pour les utilisateurs **Administrateur** et **PROF**.

Aucune installation de logiciel supplémentaire ne peut être effectuée sans la permission du professeur. Les systèmes non conformes seront formatés sans préavis.

VOICI LA LISTE DES MODIFICATIONS

Gestionnaire de serveur - Serveur local

Dans les **Propriétés** de votre serveur

- Désactiver la **Configuration de sécurité renforcée d'Internet Explorer** pour les administrateurs
- Désactiver la **Configuration de sécurité renforcée d'Internet Explorer** pour les utilisateurs

Explorateur de fichiers - Affichage - Options des dossiers

Onglet **Affichage**

- Sélectionner **Afficher les fichiers, dossiers et lecteurs cachés**
- Décocher **Masquer les extensions des fichiers dont le type est connu**
- Décocher **Masquer les fichiers protégés du système d'exploitation (recommandé)**
  - Il faut confirmer votre choix
- Décocher **Utiliser l'Assistant Partage (recommandé)**

Panneau de configuration - Outils d'administration - Services

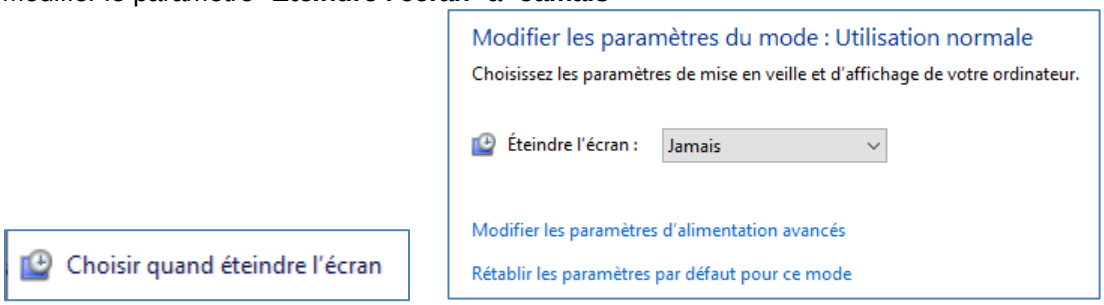
Sélectionner le service "Windows Update"

- Type de démarrage = Désactivé
- État du service = Arrêté

Panneau de configuration - Options d'alimentation

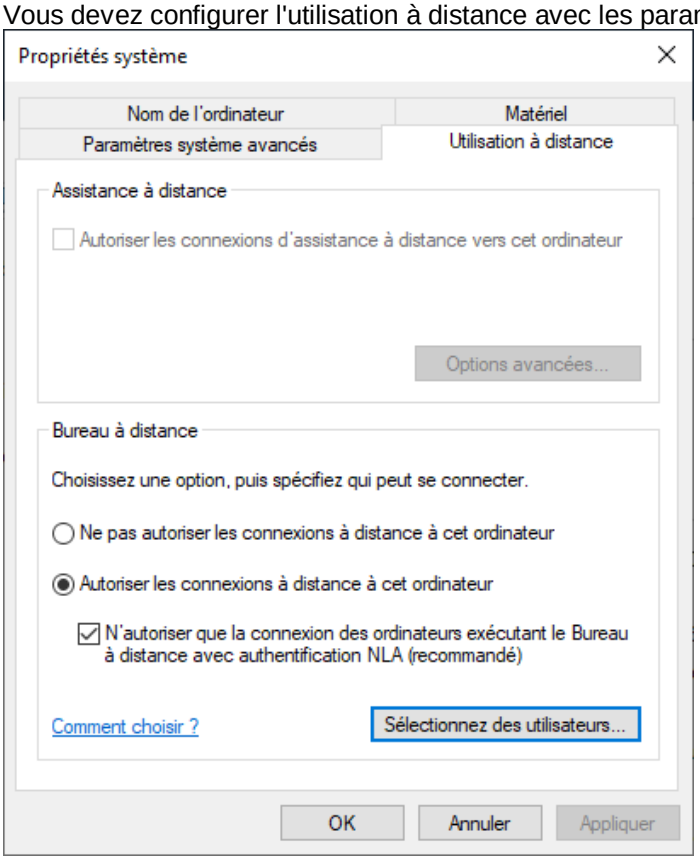
Vous devez cliquer sur "**Choisir quand éteindre l'écran**"

- Modifier le paramètre "**Éteindre l'écran**" à "**Jamais**"



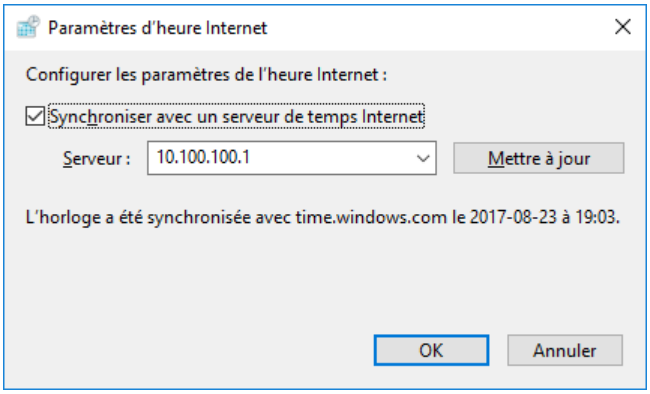
Panneau de configuration - Système

Vous devez configurer l'utilisation à distance avec les paramètres suivants



Synchroniser l'heure avec un serveur de temps

Démarrer la console "**Date et heure**" en exécutant "**timedate.cpl**"
Sélectionner l'onglet "**Temps Internet**"
Cliquer sur le bouton "**Modifier les paramètres…**"



Exécuter NCPA.CPL pour afficher la console "Connexions réseau"

Si **reseau.cvm** est sous le nom de la carte réseau, vous devez la renommer "**OnBoard**".

Vous devez renommer l'autre carte réseau "**PCI**".

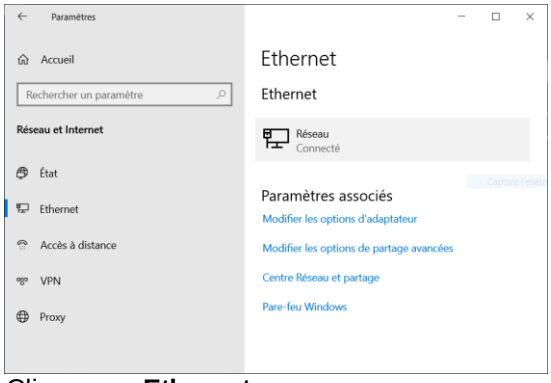
Panneau de configuration - Centre Réseau et partage

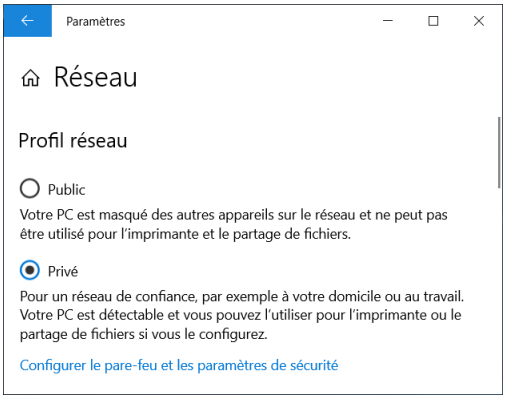
Dans la section "**Afficher vos réseaux actifs**" vérifiez que la carte réseau **OnBoard** utilise un "**Réseau privé**".

Si ce n'est pas le cas, vous pouvez effectuer la modification en utilisant une des deux méthodes.

- **Méthode 1 – démarrer PowerShell en tant qu'administrateur**
  `Set-NetConnectionProfile -InterfaceAlias "OnBoard" -NetworkCategory Private`

- **Méthode 2 – environnement graphique**
  - Utiliser les touches **Windows + i**
  - Cliquer sur l'icône "**Réseau et Internet**"



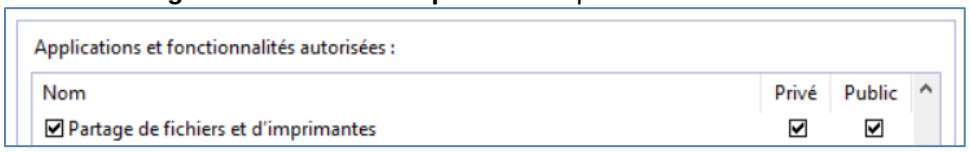


  - Cliquer sur **Ethernet**
  - Cliquer sur "**Réseau"** avec le sous-titre **"Connecté"**
  - Activer le bouton pour "**Privé"**
    résultat: la carte réseau OnBoard va utiliser un réseau "Privé"

Panneau de configuration - Pare-feu Windows

Cliquer sur **Autoriser une application ou une fonctionnalité via le Pare-feu Windows**

- Activer "**Partage de fichiers et d'imprimantes**" pour un réseau "**Privé**" et "**Public**"



Modification d'une clé du registre

L'activation de ce paramètre permet à votre professeur d'accéder à distance à votre disque dur avec le compte PROF.

Comportement par défaut
Par défaut, seul le compte Administrateur peut accéder aux partages administratifs (C$, D$, ADMIN$, ...) si le contrôle du compte d'utilisateur est activé.

Comportement après la modification de la clé de registre
Les membres du groupe Administrateurs peuvent accéder aux partages administratifs (C$, D$, ADMIN$, ...).

Copier le fichier **LocalAccountTokenFilterPolicy_ON.reg** qui est dans le dossier "**\\uranusprof.reseau.cvm\intranet\rjean\REGISTRE - Partages administratifs**" dans un dossier à la racine du C:\, par exemple C:_REGISTRE.

Double cliquer sur le fichier "**C:_REGISTRE\LocalAccountTokenFilterPolicy_ON.reg**" et répondre OUI aux questions.

Pour que la clé du registre fonctionne correctement, vous devez redémarrer l'ordinateur.

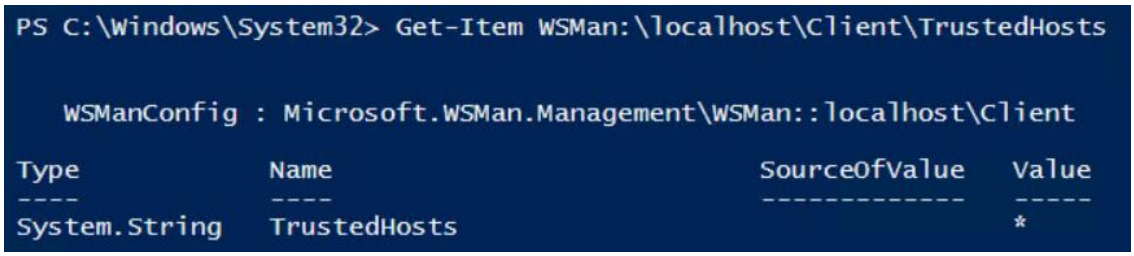
PowerShell - permettre l'exécution de script à distance

L'exécution de ces commandes vont autoriser votre professeur à exécuter des scripts PowerShell sur votre ordinateur à partir de son ordinateur.

Démarrer "**PowerShell ISE**" en tant qu'administrateur

- Exécuter les commandes suivantes

```
# Autorise l'ordinateur à recevoir des commandes à distance
Enable-PSRemoting -SkipNetworkProfileCheck -Force

# Cette commande ajoute les ordinateurs auxquels on a confiance
# * correspond à tous les ordinateurs
Set-Item WSMan:\localhost\Client\TrustedHosts -Value * -Force

# Redémarre le service "Gestion à distance de Windows (Gestion WSM)"
Restart-Service WinRM

# Cette commande affiche les ordinateurs auxquels on a confiance
Get-Item WSMan:\localhost\Client\TrustedHosts
```



Permettre l'arrêt de l'ordinateur sans avoir à se connecter

Dans le menu Démarrer taper **gpedit.msc**

- Dans la console qui s'est ouverte
  - Sélectionner **Stratégie Ordinateur local**
  - Ouvrir la section **Configuration ordinateur**
  - Ouvrir la section **Paramètres Windows**
  - Ouvrir **Paramètres de sécurité / Stratégies locales / Options de sécurité**
  - Dans le panneau de droite activer la stratégie
    Arrêt: permet au système d'être arrêté sans avoir à se connecter
- Fermer la console

Démonter les partitions sauf le disque C:\

1. Dans l'Explorateur de fichiers vérifier les lettres associées aux partitions du disque dur de l'ordinateur.

2. Démarrer l'invite de commandes (cmd.exe) en tant qu'administrateur

3. Dans l'invite de commandes, vous aurez à exécuter la commande MOUNTVOL.EXE plusieurs fois.

```
a) mountvol.exe e: /d
b) mountvol.exe f: /d
c) mountvol.exe g: /d
d) mountvol.exe h: /d
e) mountvol.exe i: /d
f) mountvol.exe j: /d
g) mountvol.exe k: /d
```

Panneau de configuration

Changer le mode d'affichage pour **Grandes icônes**

Gestionnaire de serveur

Dans l'option **Gérer**

- Sélectionner l'option **Propriétés du Gestionnaire de serveur**
- Cocher l'option **Ne pas démarrer automatiquement le Gestionnaire de serveur…**

Le rôle Hyper-V

Ce laboratoire doit être fait individuellement sur votre serveur réel

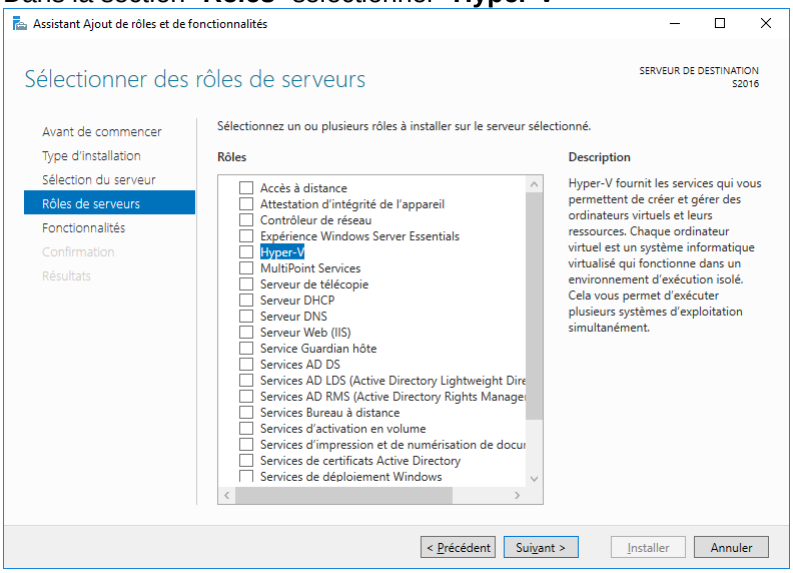
Objectif

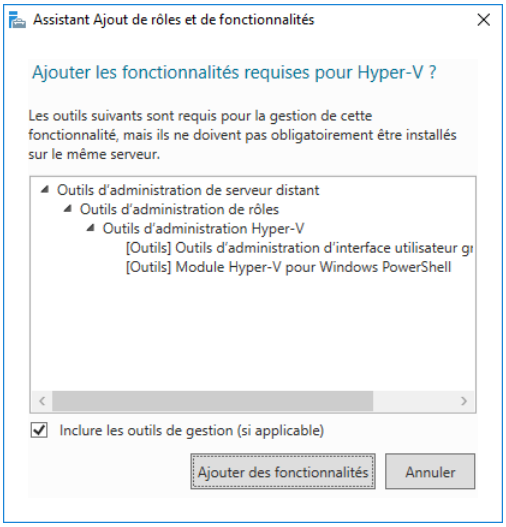
Ce laboratoire vise à installer et configurer le rôle Hyper-V.

Étape 1 - Installation du rôle Hyper-V sur le serveur réel

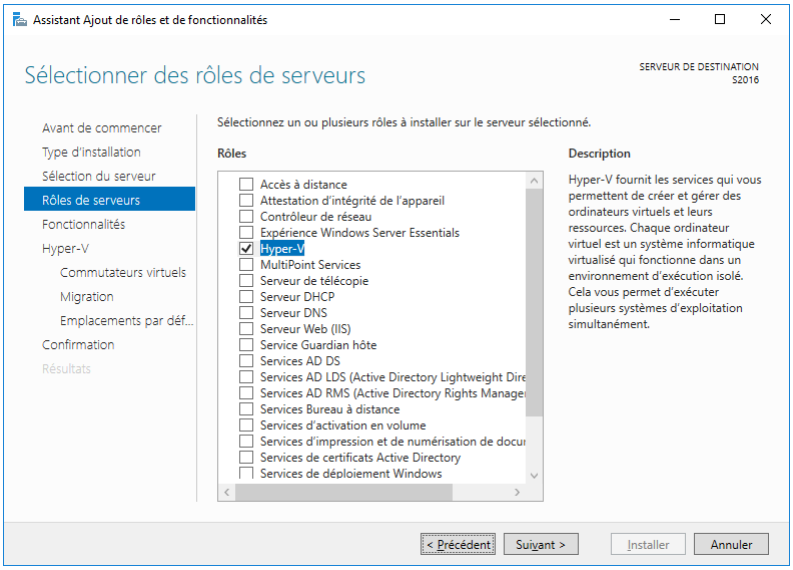
Démarrer la console "**Gestionnaire de serveur**"

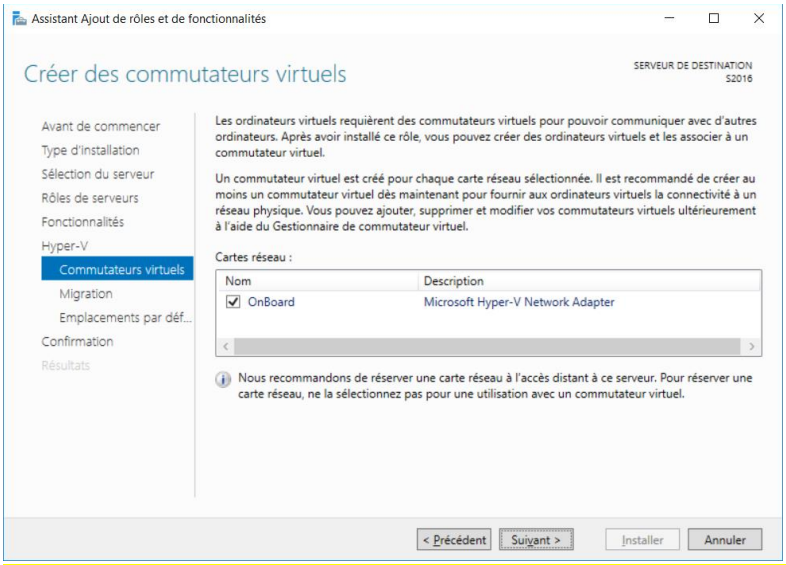
- Dans "**Gérer**" sélectionner "**Ajouter des rôles et fonctionnalités**"
  Dans la section "**Rôles**" sélectionner "**Hyper-V**"





Cliquer sur le bouton "**Ajouter des fonctionnalités**"

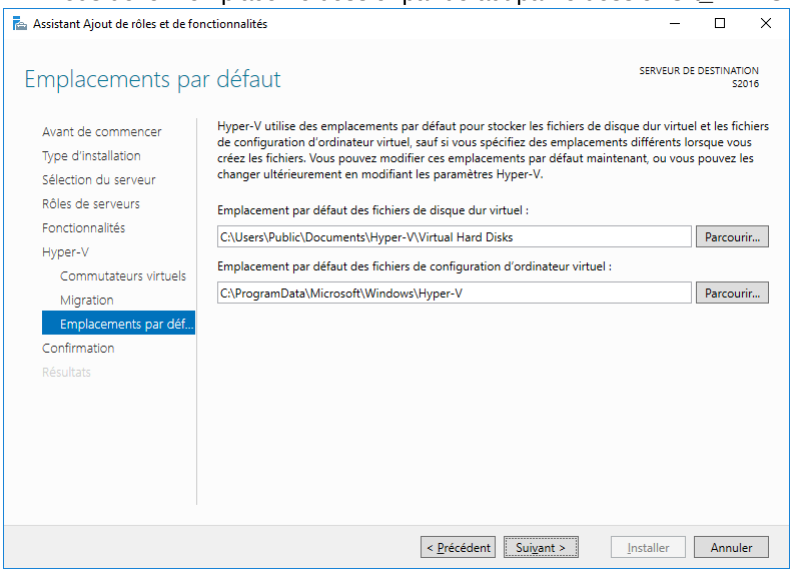


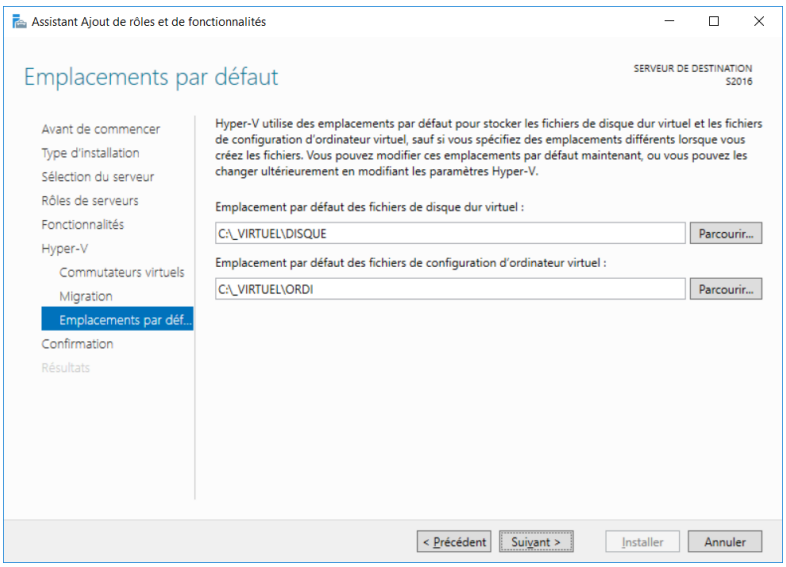


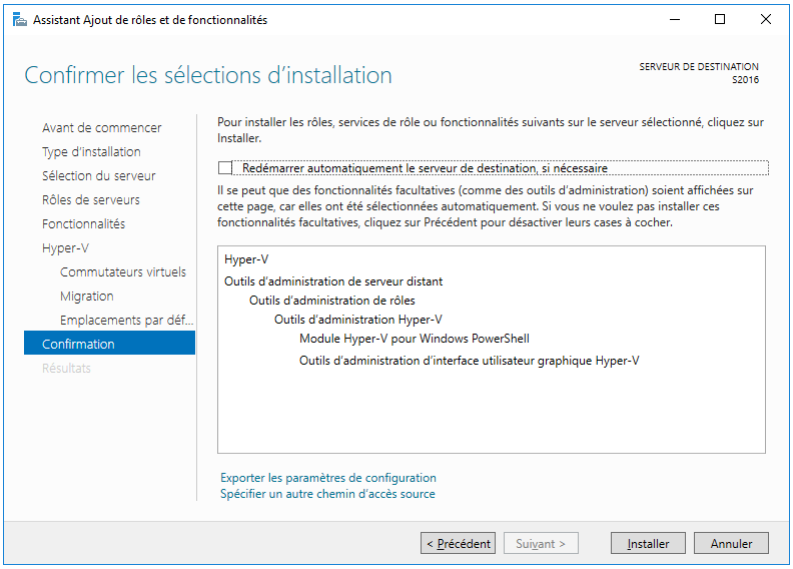
La description de votre carte réseau sera différente de la capture d'écran.

À cette étape, vous devez modifier les deux emplacements par défaut.

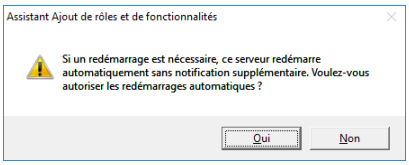
- "**Emplacement par défaut des fichiers de disque dur virtuel**"
  vous devez remplacer le dossier par défaut par le dossier **C:_VIRTUEL\DISQUE**
- "**Emplacement par défaut des fichiers de configuration d'ordinateur virtuel**"
  vous devez remplacer le dossier par défaut par le dossier **C:_VIRTUEL\ORDI**





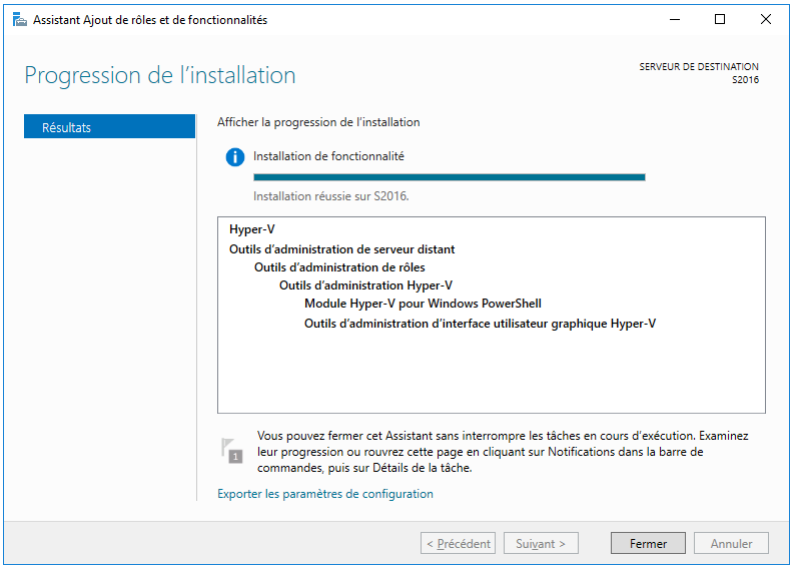


Sélectionner "**Redémarrer automatiquement le serveur de destination, si nécessaire**"



Cliquer sur le bouton "**Oui**"

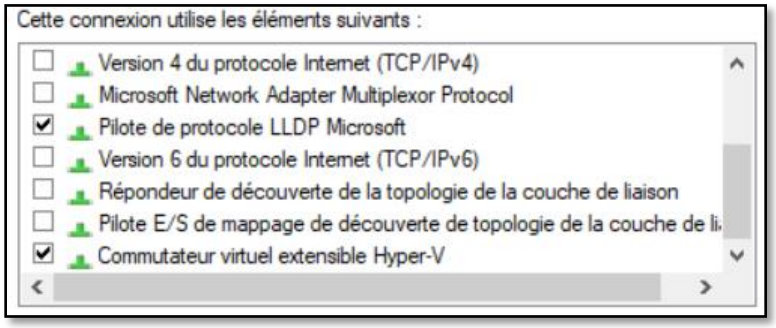
Cliquer sur le bouton "**Installer**"
note: l'ordinateur va redémarrer deux fois



Étape 2 - Hyper-V et les cartes réseau

- Démarrer la console "Connexions réseau" avec la commande NCPA.CPL
  Vous avez une carte réseau supplémentaire dont le nom débute par **vEthernet**.
  La carte réseau **vEthernet** a récupérée la configuration IPv4 de la carte réseau **OnBoard**.

  La carte réseau OnBoard a perdu sa configuration IPv4 au profit de la nouvelle carte réseau.

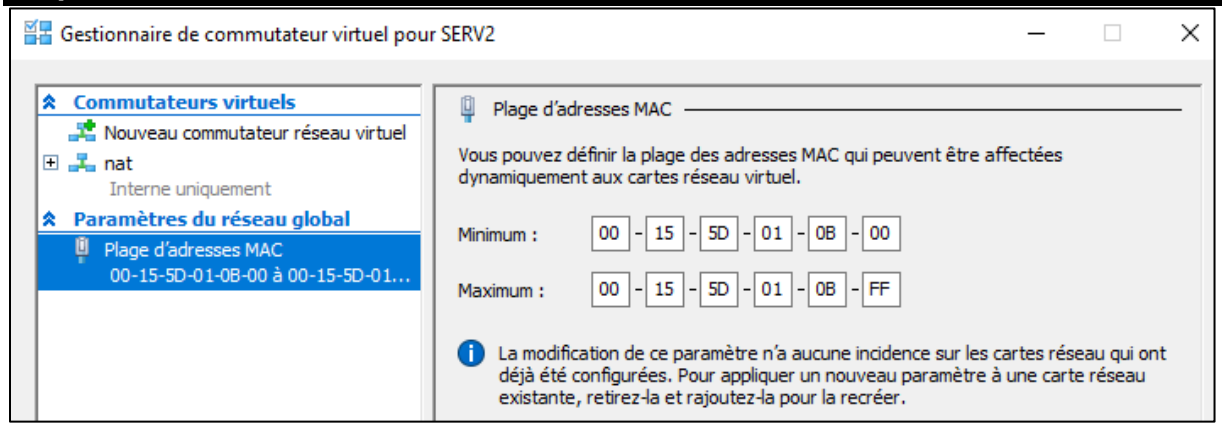


Étape 3 - Paramétrer les commutateurs virtuels

Démarrer la console "Gestionnaire Hyper-V".

- Dans la section de gauche, vous devez sélectionner votre serveur
  Dans le menu contextuel, choisir "**Gestionnaire de commutateur virtuel…**"
  - Sélectionner le commutateur que vous avez créé et modifier son nom pour "**EXTERNE**".
  - Vous devez créer un commutateur virtuel de type PRIVÉ dont le nom sera "**PRIVE**".

Étape 4 - Gestionnaire de commutateur virtuel...

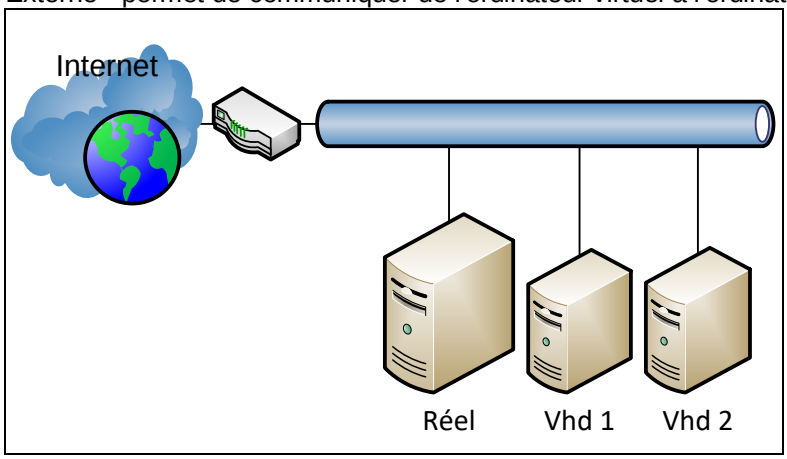


On désire modifier la plage des adresses MAC pour utiliser la plage suivante
00155Dccpp00 à **00155Dccpp**FF
cc = 37 pour le local A5.37
cc = 38 pour le local A5.38
cc = 47 pour le local C4.07
cc = 48 pour le local C4.08
pp = numéro de votre ordinateur réel en utilisant deux chiffres

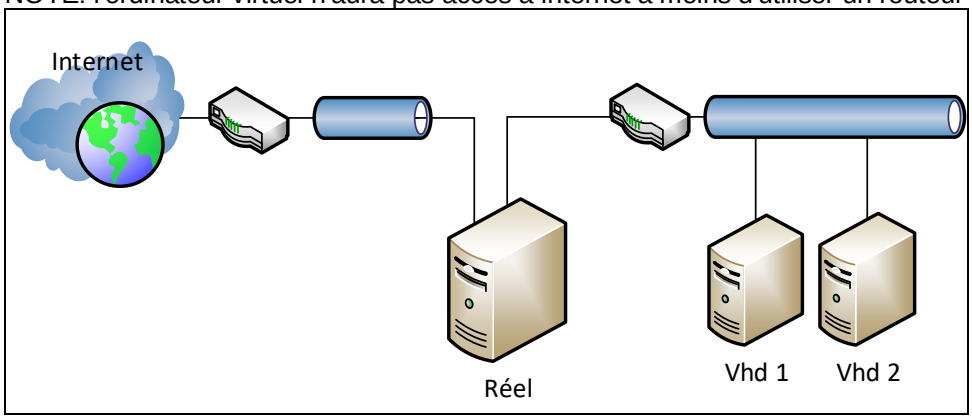
Les commutateurs dans Hyper-V

Les trois types de commutateurs réseaux qui sont possibles avec Hyper-V.

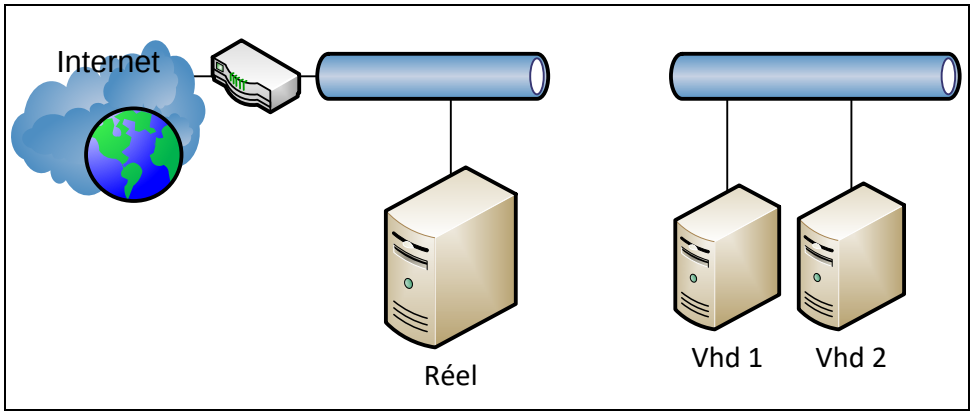
a) Externe - permet de communiquer de l'ordinateur virtuel à l'ordinateur réel et d'avoir accès à internet



b) Interne - permet de communiquer de l'ordinateur virtuel à l'ordinateur réel et vice versa
NOTE: l'ordinateur virtuel n'aura pas accès à internet à moins d'utiliser un routeur



c) Privé - aucune communication de l'ordinateur virtuel à l'ordinateur réel et vice versa



<u>Deux raccourcis clavier à connaître dans Hyper-V</u>
Lest touches **Ctrl-Alt gauche** pour relâcher la souris.
Les touches **Ctrl-Alt-Fin** pour **Ctrl-Alt-Del**

Installation du routeur pfSense dans Hyper-V

Ce laboratoire doit être fait individuellement sur votre serveur réel

Copier le fichier **pfSense 2.5.2.vhdx** qui est dans le dossier **\\uranusprof.reseau.cvm\intranet\rjean\Hyper-V\GEN1 - pfSense 2.5.2** dans le dossier **C:_VIRTUEL\DISQUE** de votre serveur réel. Je vous conseille de garder une copie de ce fichier.

Étape 1 - Créer un ordinateur virtuel avec pfSense

- Création de l'ordinateur virtuel "**ROUTEUR**"
  - Nom: **ROUTEUR**
    L'emplacement devrait être "C:_VIRTUEL\ORDI"
  - Génération 1
  - Mémoire de démarrage: 1024 Mo
    ne pas cocher "Utiliser la mémoire dynamique pour cet ordinateur virtuel."
  - Connexion: **EXTERNE**
  - Sélectionner l'option "**Utiliser un disque dur virtuel existant**"
    - Emplacement: C:_VIRTUEL\Disque
    - Nom du fichier: **routeur.vhdx**

À la fin de la création de l'ordinateur virtuel, vous devez ajouter une deuxième carte réseau.

- Cette carte réseau doit utiliser le commutateur virtuel "**PRIVE**"

Démarrer le routeur pfSense dans Hyper-V

```
Starting syslog...done.
Starting CRON... done.
pfSense 2.5.2-RELEASE amd64 Fri Jul 02 15:33:00 EDT 2021
Bootup complete

FreeBSD/amd64 (pfSense.home.arpa) (ttyv0)

Microsoft Azure - Netgate Device ID: 9ae8a755f87dbb89b57a

*** Welcome to pfSense 2.5.2-RELEASE (amd64) on pfSense ***

 WAN (wan)       -> hn0        -> v4/DHCP4: 10.57.251.42/16
 LAN (lan)       -> hn1        -> v4: 192.168.1.1/24

 0) Logout (SSH only)                  9) pfTop
 1) Assign Interfaces                 10) Filter Logs
 2) Set interface(s) IP address       11) Restart webConfigurator
 3) Reset webConfigurator password    12) PHP shell + pfSense tools
 4) Reset to factory defaults         13) Update from console
 5) Reboot system                     14) Enable Secure Shell (sshd)
 6) Halt system                       15) Restore recent configuration
 7) Ping host                         16) Restart PHP-FPM
 8) Shell

Enter an option:
```

La fenêtre principale affiche l'adresse IP du réseau WAN et l'adresse IP du réseau LAN.

- Par défaut, l'adresse IP du réseau LAN est 192.168.1.1/24.
- L'adresse IP du réseau WAN est obtenue par le serveur DHCP du cégep.
  L'adresse IP du réseau WAN sera 10.57.**x**.**y**/16

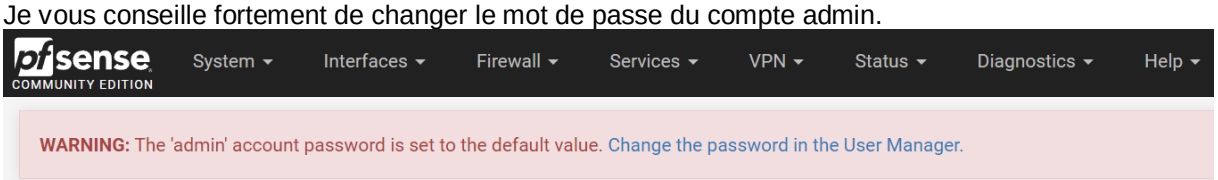
ANNEXE 1

La modification des paramètres du routeur pfSense se fait en utilisant l'interface WEB.

Pour modifier la configuration du routeur pfSense à partir de votre serveur réel.
https://10.57.x.y
nom d'utilisateur = admin
mot de passe = pfsense

Pour modifier la configuration du routeur pfSense à partir d'un ordinateur virtuel qui est dans le réseau LAN.
https://192.168.1.1
nom d'utilisateur = admin
mot de passe = pfsense

Je vous conseille fortement de changer le mot de passe du compte admin.



LE LABORATOIRE SE TERMINE ICI.

ANNEXE 2

pfSense Documentation
https://docs.netgate.com/pfsense/en/latest

Virtualizing pfSense Software with Hyper-V
https://docs.netgate.com/pfsense/en/latest/recipes/virtualize-hyper-v.html

---

Le fichier **pfSense 2.5.2.vhdx** contient déjà plusieurs configurations pour faciliter l'utilisation du routeur.

Voici la liste des paramètres déjà configurés dans le fichier "pfSense 2.5.2.vhdx".

Paramètre 1
Modifier le fuseau horaire

Paramètre 2
Désactiver l'option "Block private networks and loopback addresses"

Paramètre 3
Création de deux règles dans le pare-feu de la carte réseau WAN

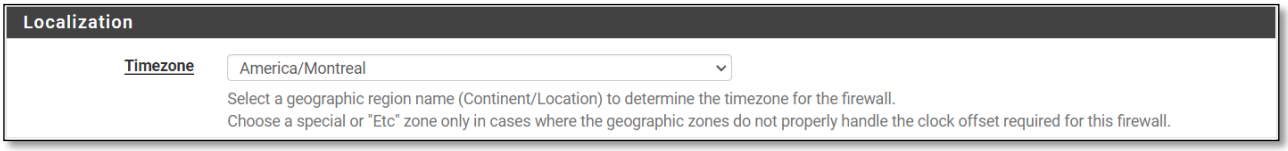
- **Création d'une règle qui accepte les PING de l'externe**
- **Création d'une règle qui autorise la gestion à distance du routeur pfSense**

Les prochaines pages affichent les paramètres déjà configurés dans le fichier "pfSense 2.5.2.vhdx".

CONFIGURATION DU PARAMÈTRE 1

Dans le menu choisir "System / General Setup"
Dans la section **Localization**
Modifier le **Timezone** pour **America/Montreal**

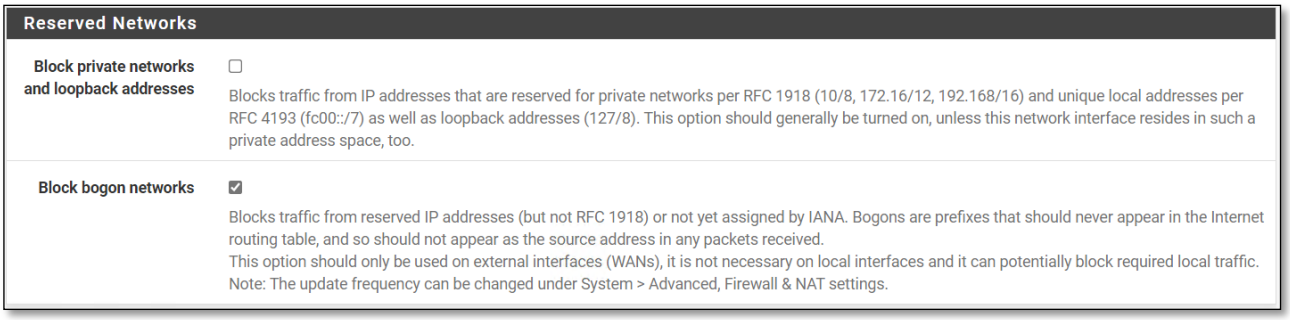
Résultat final



CONFIGURATION DU PARAMÈTRE 2

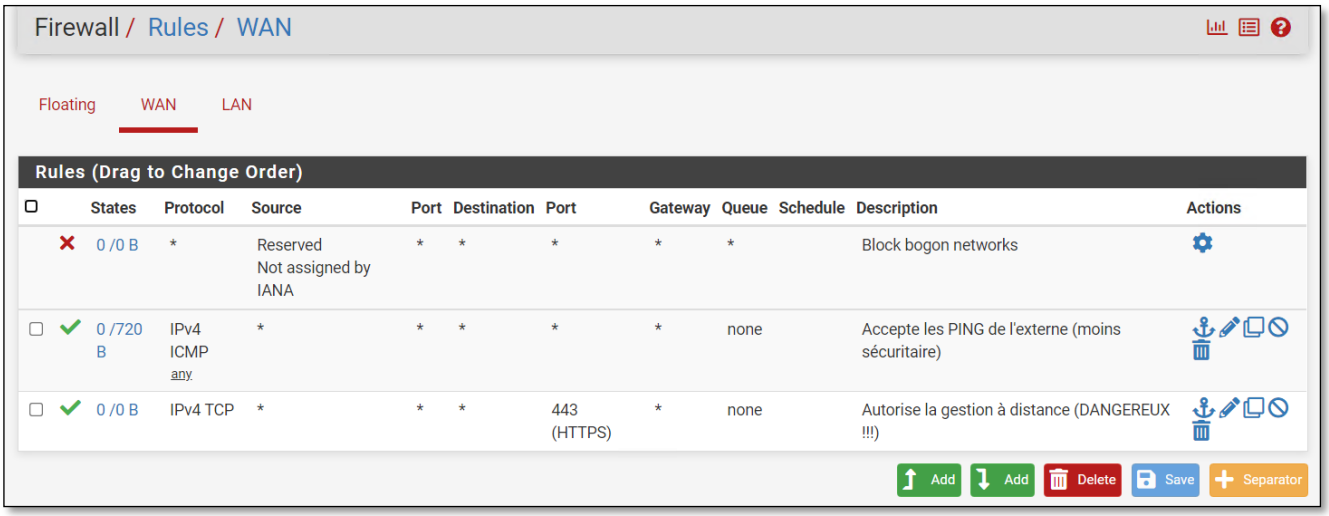
Dans le menu choisir "Interfaces / WAN"
Dans la section "**Reserved Networks**"
Désactiver l'option **"Block private networks and loopback addresses"**

Résultat final



CONFIGURATION DU PARAMÈTRE 3

VOICI LE RÉSULTAT À OBTENIR APRÈS L'AJOUT DES DEUX RÈGLES



On va ajouter deux règles en utilisant le bouton "**Add avec la flèche qui pointe vers le bas**"

1) Accepte les PING de l'externe (moins sécuritaire)
   note: les étapes sont sur la page 5
2) Autorise la gestion à distance (DANGEREUX !!!)
   note: les étapes sont sur la page 6

CONFIGURATION DE LA PREMIÈRE RÈGLE DU PARAMÈTRE 3

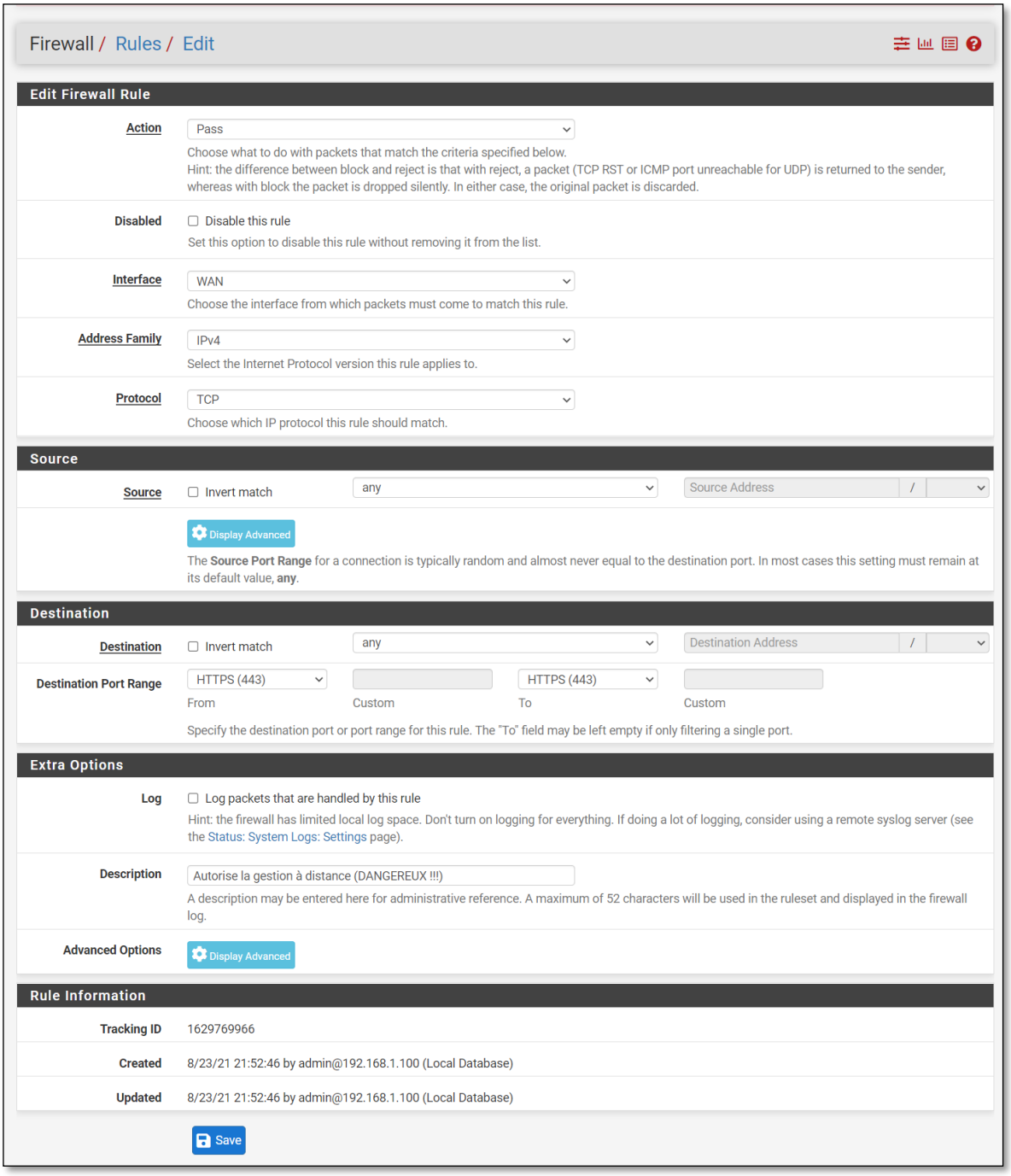
Création de la règle qui accepte les PING de l'externe

Dans le menu choisir "Firewall / Rules / WAN"

CONFIGURATION DE LA DEUXIÈME RÈGLE DU PARAMÈTRE 3

Création de la règle qui autorise la gestion à distance du routeur pfSense

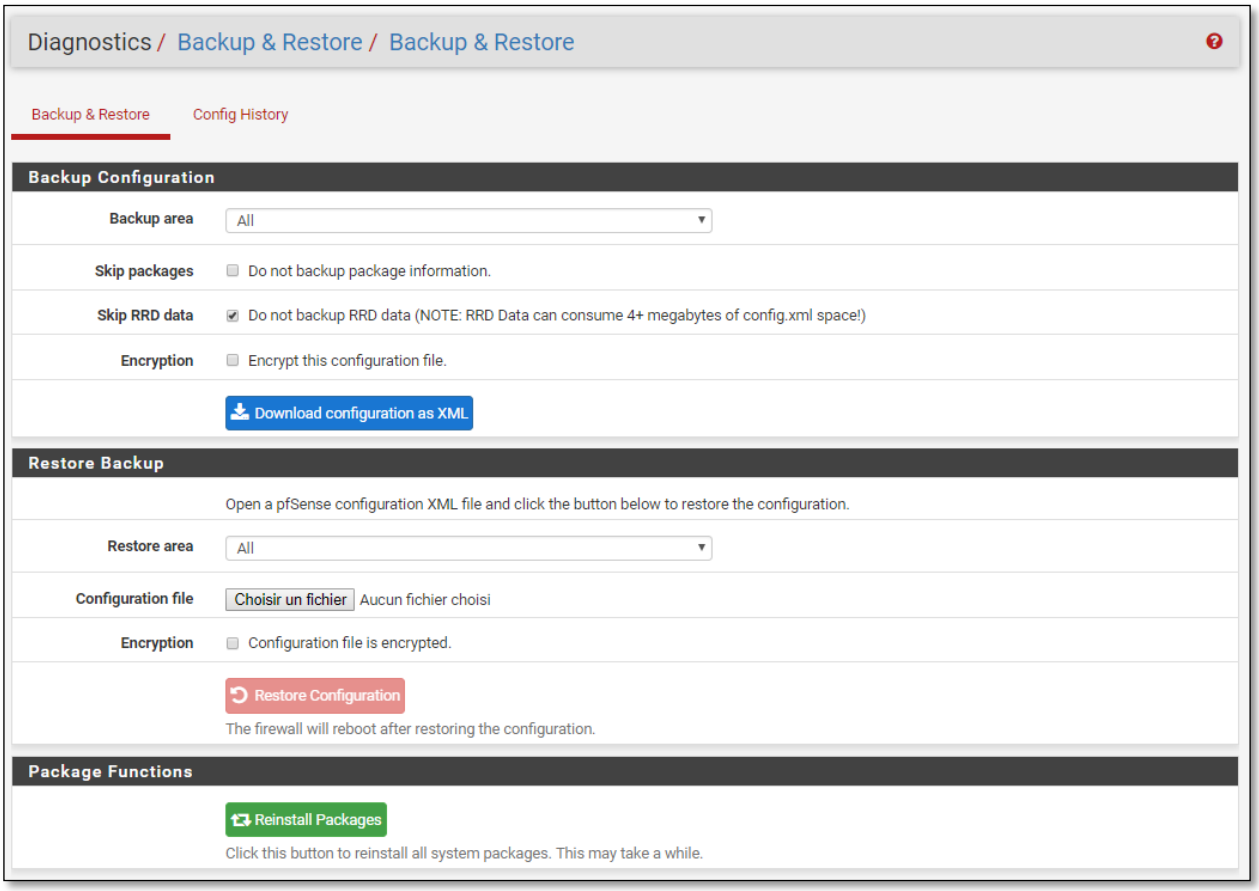
Dans le menu choisir "Firewall / Rules / WAN"



Cette règle permet de configurer le routeur à partir de votre serveur réel avec **https://10.57.x.y**

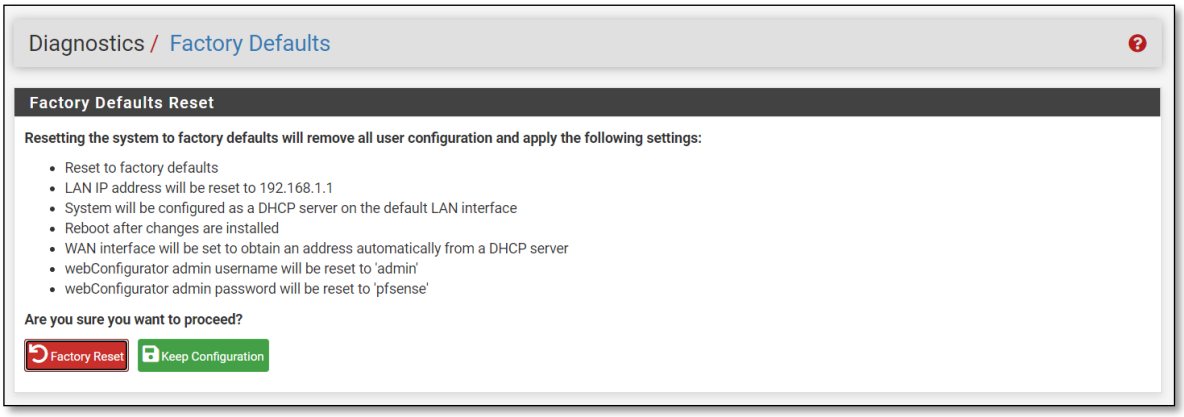
ANNEXE 3

Vous pouvez sauvegarder et restaurer la configuration du routeur pfSense.



ANNEXE 4

Vous pouvez revenir à la configuration initiale.



Création des ordinateurs virtuels dans Hyper-V

Ce laboratoire doit être fait individuellement sur votre serveur réel

Objectifs

Création des ordinateurs virtuels à partir des fichiers ISO.

- Vous devez créer le dossier **C:_ISO** sur votre serveur réel.
- Vous devez copier les fichiers qui sont dans le dossier **\\uranusprof.reseau.cvm\intranet\rjean\ISO\S2019** dans le dossier **C:_ISO** de votre serveur réel.

Vous devez vérifier l'intégrité du fichier ISO avec le script PowerShell **S2019_SHA256.ps1**

```
PS C:\_ISO> .\S2019_SHA256.ps1
Le fichier 'fr-fr_windows_server_2019_updated_aug_2021_x64_dvd_b863695e.iso' est valide.

PS C:\_ISO>
```

Étape 1 - Création de l'ordinateur virtuel "SERVEUR1"

Voici les propriétés de l'ordinateur virtuel "**SERVEUR1**"

- Nom: **SERVEUR1**
  L'emplacement devrait être "C:_VIRTUEL\ORDI"
- Génération 2
- Mémoire de démarrage: 4096 Mo
  décocher "Utiliser la mémoire dynamique pour cet ordinateur virtuel."
- Connexion: **PRIVE**
- Cocher "Créer un disque dur virtuel"
  - Nom: **serveur1.vhdx**
  - Emplacement: C:_VIRTUEL\DISQUE
  - Taille 60 GO
- Cocher "Installer un système d'exploitation à partir d'un fichier image de démarrage"
  Fichier image (.iso): **C:_ISO\fr-fr_windows_server_2019_updated_aug_2021_x64_dvd_b863695e.iso**

Modifier les paramètres de l'ordinateur virtuel pour utiliser quatre processeurs virtuels.

Étape 2 - Création de l'ordinateur virtuel "SERVEUR2"

Voici les propriétés de l'ordinateur virtuel "**SERVEUR2**"

- Nom: **SERVEUR2**
  L'emplacement devrait être "C:_VIRTUEL\ORDI"
- Génération 2
- Mémoire de démarrage: 4096 Mo
  décocher "Utiliser la mémoire dynamique pour cet ordinateur virtuel."
- Connexion: **PRIVE**
- Cocher "Créer un disque dur virtuel"
  - Nom: **serveur2.vhdx**
  - Emplacement: C:_VIRTUEL\DISQUE
  - Taille 60 GO
- Cocher "Installer un système d'exploitation à partir d'un fichier image de démarrage"
  Fichier image (.iso): **C:_ISO\fr-fr_windows_server_2019_updated_aug_2021_x64_dvd_b863695e.iso**

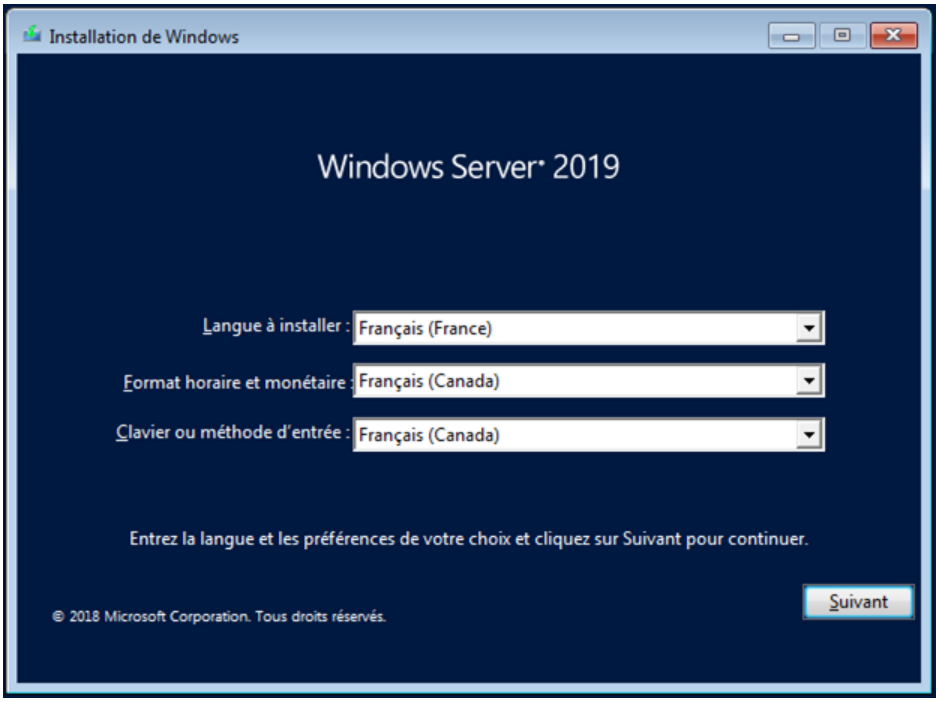
Modifier les paramètres de l'ordinateur virtuel pour utiliser quatre processeurs virtuels.

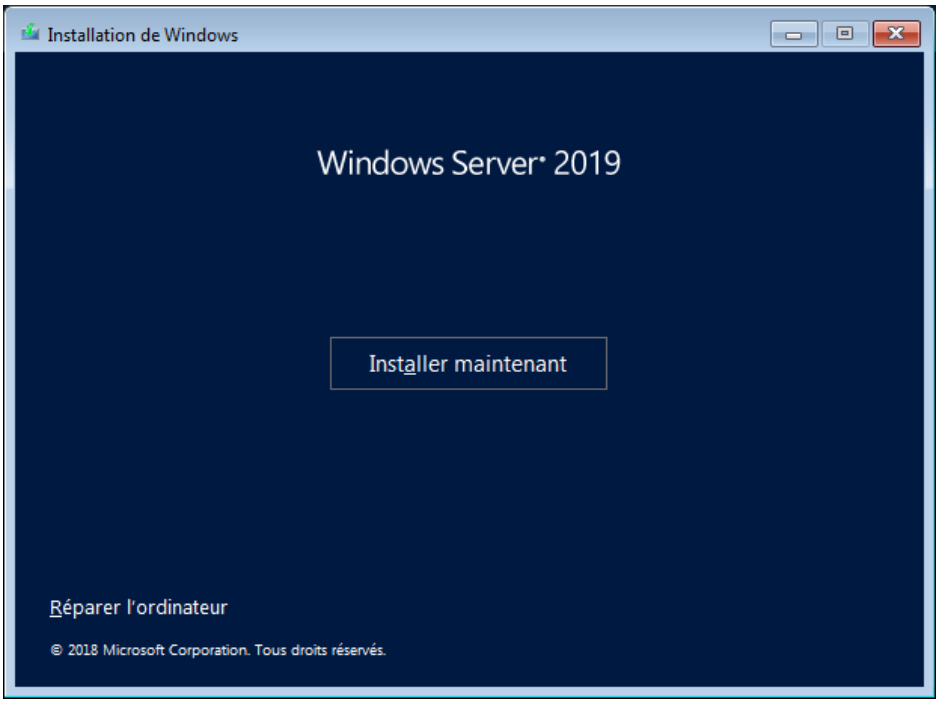
ANNEXE - Installation de Windows Server 2019

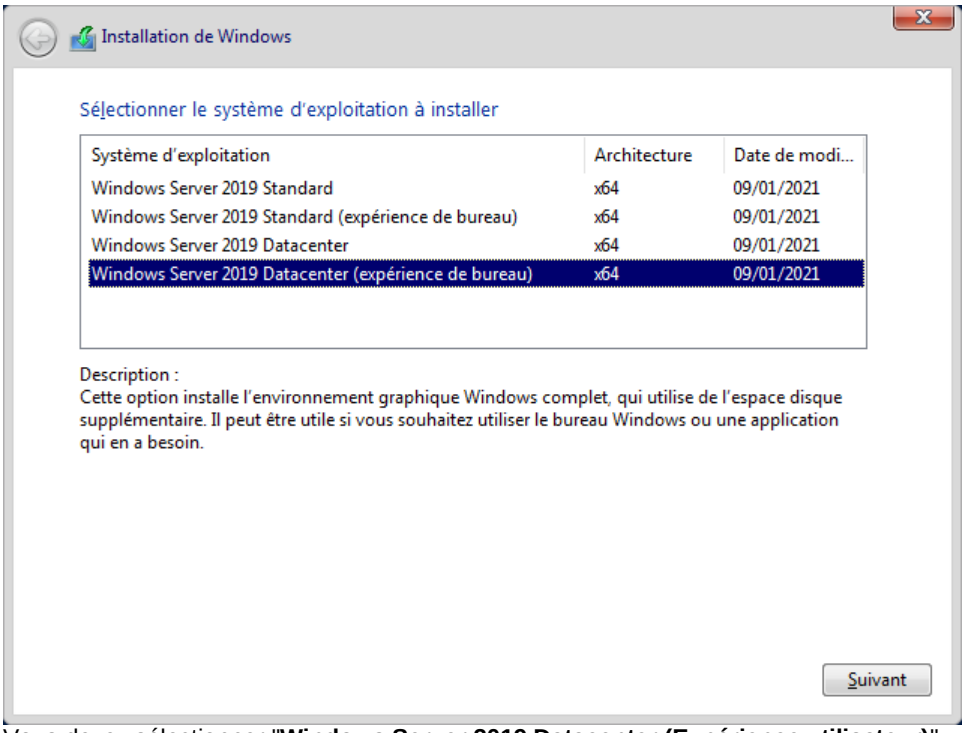
Démarrer l'ordinateur virtuel pour débuter l'installation de "Windows Server 2019".

Vous avez environ trois secondes pour démarrer avec le fichier ISO.

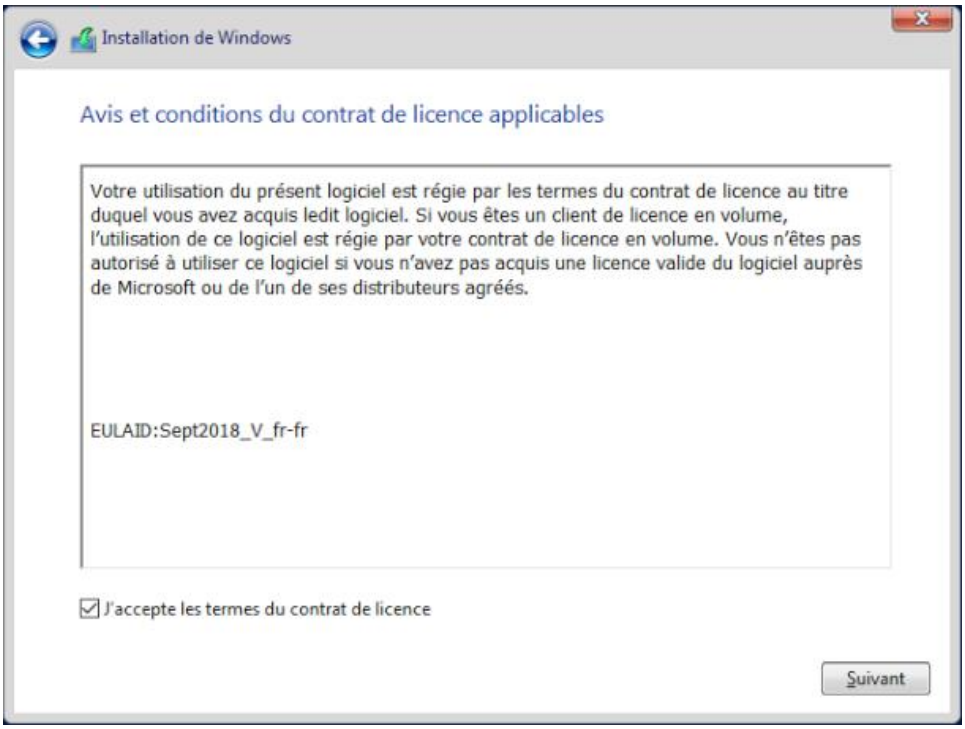
```
Press any key to boot from CD or DVD.
```



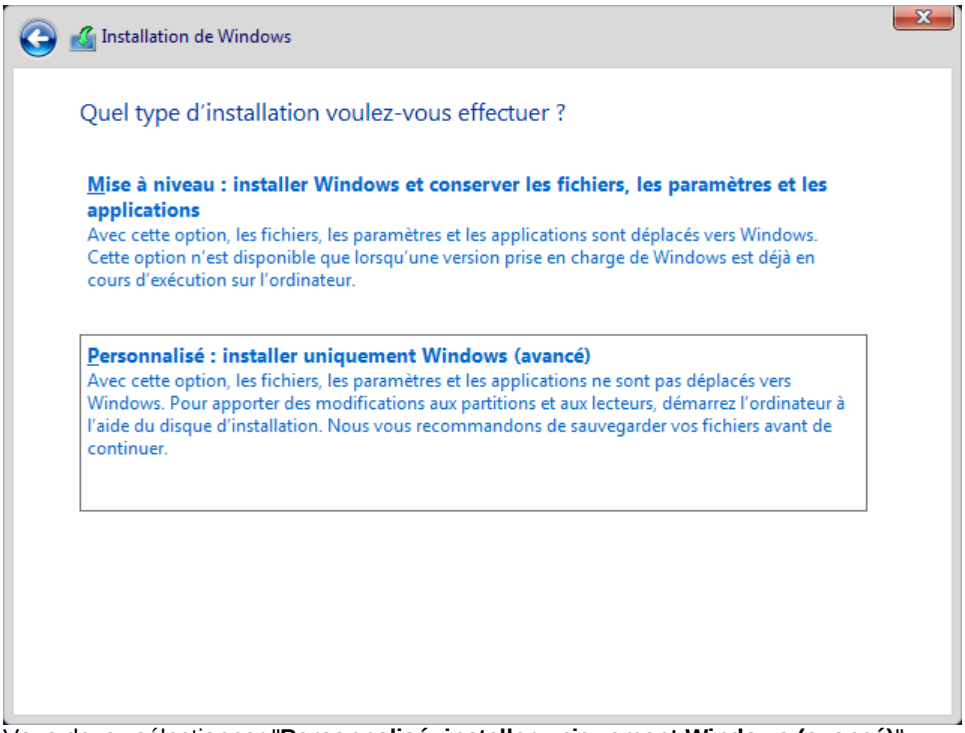




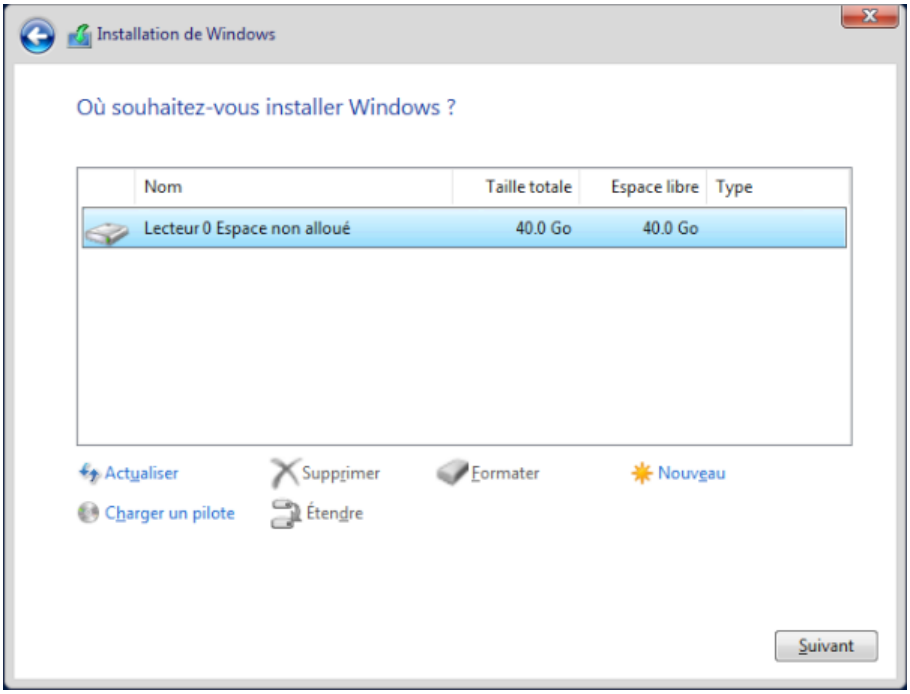
Vous devez sélectionner "**Windows Server 2019 Datacenter (Expérience utilisateur)**"



Vous devez cocher "**J'accepte les termes du contrat de licence**"

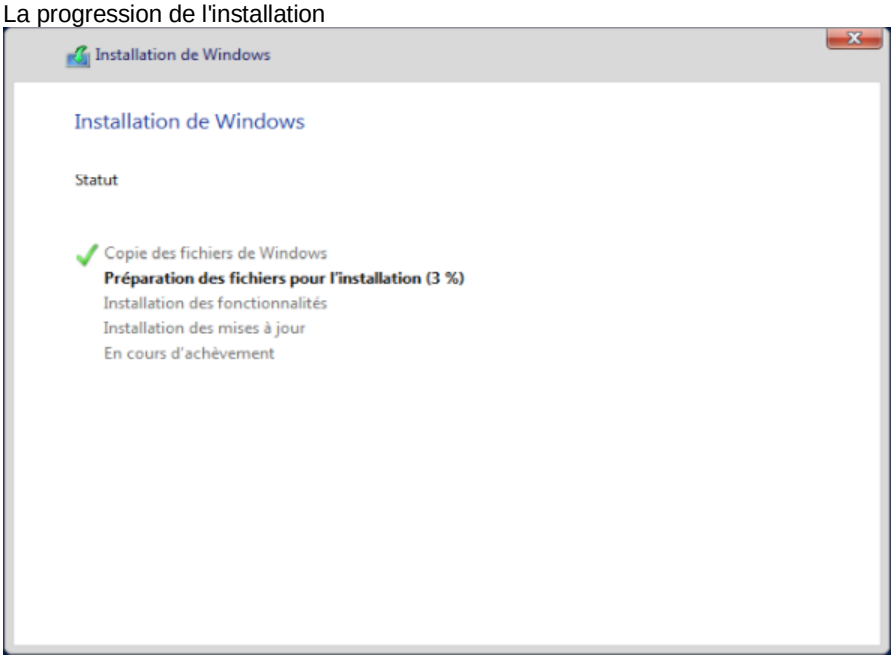


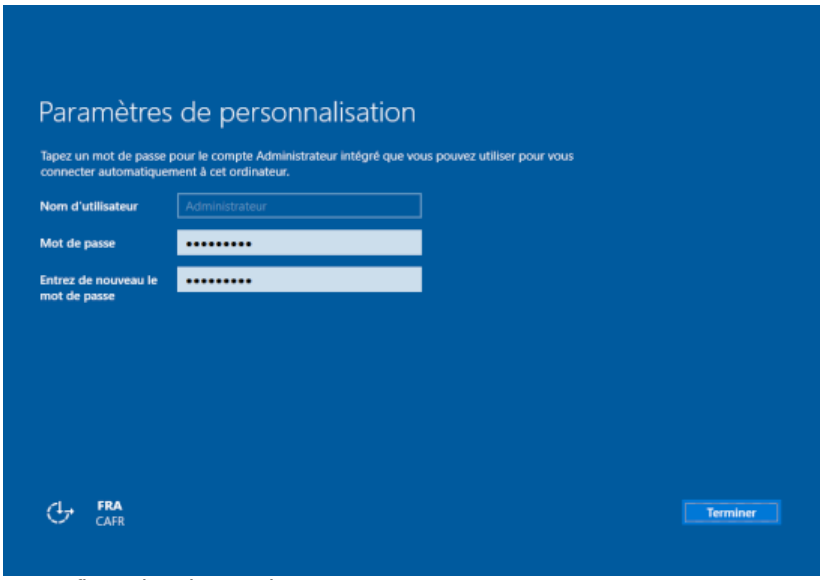
Vous devez sélectionner "**Personnalisé: installer uniquement Windows (avancé)**"



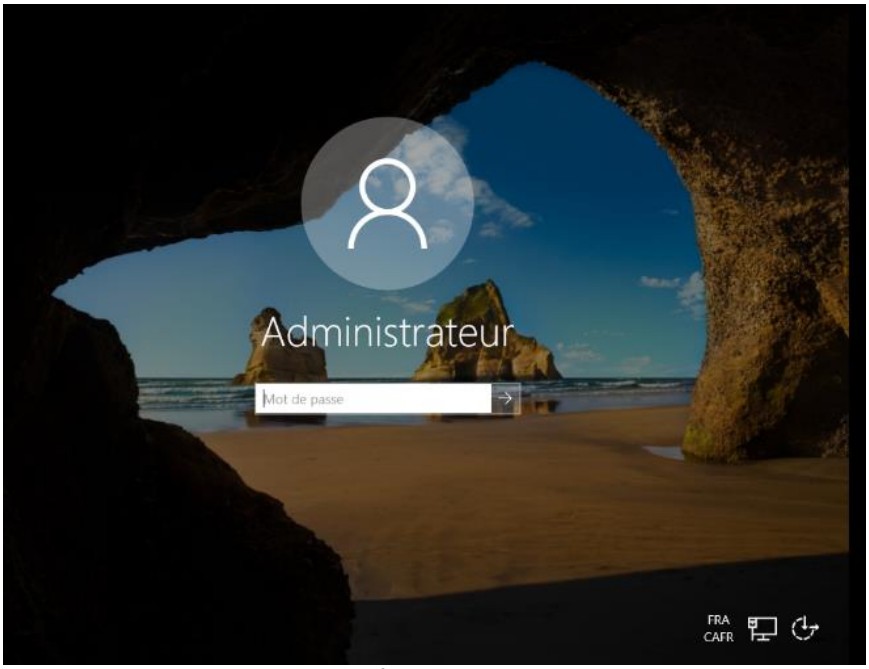
Vous devez cliquer sur le bouton "**Suivant**"

La progression de l'installation





La configuration du mot de passe



"**Windows Server 2019**" est installé.

Configurations de vos ordinateurs virtuels

Ce laboratoire doit être fait individuellement sur vos ordinateurs virtuels

Objectifs

- Configurer les ordinateurs virtuels "**SERVEUR1**" et "**SERVEUR2**"

Aucune installation de logiciel supplémentaire ne peut être effectuée sans la permission du professeur. Les systèmes non conformes seront formatés sans préavis.

Avant de commencer, vous devez renommer vos serveurs virtuels:

- SERVEUR1 le nom de l'ordinateur dans Windows
- SERVEUR2 le nom de l'ordinateur dans Windows

VOICI LA LISTE DES MODIFICATIONS

Gestionnaire de serveur - Serveur local

Dans les **Propriétés** de votre serveur

- Désactiver la **Configuration de sécurité renforcée d'Internet Explorer** pour les administrateurs
- Désactiver la **Configuration de sécurité renforcée d'Internet Explorer** pour les utilisateurs

Vous devez installer au moins un des navigateurs web

Voici les liens pour télécharger les versions complètes de trois navigateurs

Chrome
https://chromeenterprise.google/browser/download/#windows-tab

Edge Chromium
https://www.microsoft.com/en-us/edge/business/download

Firefox
https://www.mozilla.org/fr/firefox/all

Explorateur de fichiers - Affichage - Options des dossiers

Onglet **Affichage**

- Sélectionner **Afficher les fichiers, dossiers et lecteurs cachés**
- Décocher **Masquer les extensions des fichiers dont le type est connu**
- Décocher **Masquer les fichiers protégés du système d'exploitation (recommandé)**
  - Il faut confirmer votre choix
- Décocher **Utiliser l'Assistant Partage (recommandé)**

Panneau de configuration - Outils d'administration - Services

Sélectionner le service "Windows Update"

- Type de démarrage = Désactivé
- État du service = Arrêté

Panneau de configuration

Changer le mode d'affichage pour **Grandes icônes**

Gestionnaire de serveur

Dans l'option **Gérer**

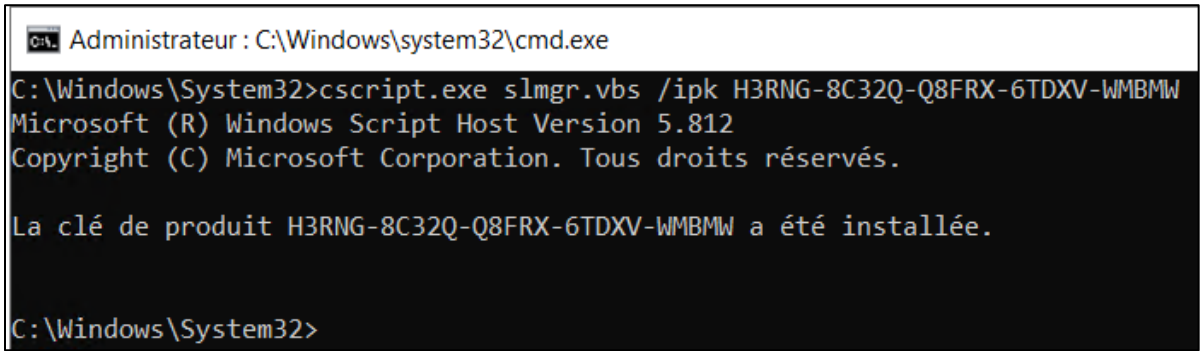
- Sélectionner l'option **Propriétés du Gestionnaire de serveur**
- Cocher l'option **Ne pas démarrer automatiquement le Gestionnaire de serveur…**

Configurer le numéro de produit AVMA "Automatic Virtual Machine Activation"

Un numéro de produit AVMA "Automatic Virtual Machine Activation" permet d'activer une machine virtuelle avec ou sans internet. Pour réussir à activer une machine virtuelle avec un numéro de produit AVMA, il faut que le serveur réel soit activé.

(**Pour votre information :**
Le site de Microsoft permet de sélectionner le bon numéro de produit AVMA à utiliser.
Automatic Virtual Machine Activation
https://docs.microsoft.com/en-us/windows-server/get-started/automatic-vm-activation)

Il faut exécuter cette commande dans une fenêtre CMD.EXE en tant qu'administrateur.
`cscript.exe slmgr.vbs /ipk H3RNG-8C32Q-Q8FRX-6TDXV-WMBMW`



Panneau de configuration - Centre Réseau et partage

Dans le menu à gauche: "**Modifier les paramètres de la carte**"

- Cliquer sur la carte réseau et la renommer **EXTERNE**

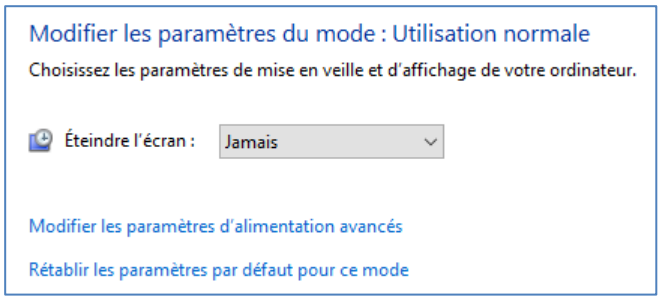
Afficher les propriétés de la carte réseau **EXTERNE**

- Modifier les propriétés de l'item **Protocole Internet version 4 (TCP/IPv4)**
  - Sélectionner **Utiliser l'adresse IP suivante:**
    Adresse IP (**SERVEUR1**): **192.168.1.10**
    Adresse IP (**SERVEUR2**): **192.168.1.20**
    Masque de sous-réseau: **255.255.255.0**
    Passerelle par défaut: **192.168.1.1**
  - Sélectionner **Utiliser l'adresse de serveur DNS suivante:**
    Serveur DNS préféré: **10.19.3.70**
    Serveur DNS auxiliaire: **10.19.3.71**

Panneau de configuration - Options d'alimentation

Vous devez cliquer sur "**Choisir quand éteindre l'écran**"

- modifier le paramètre "**Éteindre l'écran**" à "**Jamais**"



Permettre l'arrêt de l'ordinateur sans avoir à se connecter

Dans le menu Démarrer taper **gpedit.msc**

- Dans la console qui s'est ouverte
  - Sélectionner **Stratégie Ordinateur local**
  - Ouvrir la section **Configuration ordinateur**
  - Ouvrir la section **Paramètres Windows**
  - Ouvrir **Paramètres de sécurité / Stratégies locales / Options de sécurité**
  - Dans le panneau de droite activer la stratégie
    Arrêt: permet au système d'être arrêté sans avoir à se connecter
- Fermer la console

Ajouter un disque virtuel VHDX au SERVEUR1 et au SERVEUR2

Ajouter un disque virtuel VHDX à votre serveur virtuel 1

Dans le gestionnaire Hyper-V, dans le menu contextuel de votre serveur virtuel 1, sélectionner "Paramètres…"

- Sélectionner "Contrôleur SCSI"
  Dans la fenêtre de droite, sélectionner "Disque dur" et cliquer sur le bouton "Ajouter"
- Cliquer sur le bouton "Nouveau" et cliquer sur le bouton "Suivant"
  "Taille dynamique"

  Nom: **SERVEUR1_disque1.vhdx**
  Emplacement: C:_VIRTUEL\DISQUE

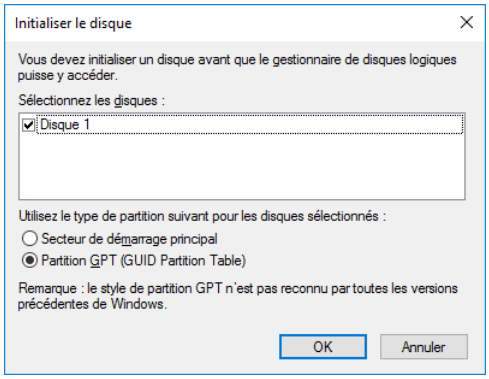
  Taille: 20 Go
- Cliquer sur le bouton "Terminer", Appliquer, Ok

À cette étape, le disque virtuel n'est pas présent dans l'Explorateur de fichiers.

Dans la console "**Gestion du disque**" du Serveur 1 (Accessible via Win-X)

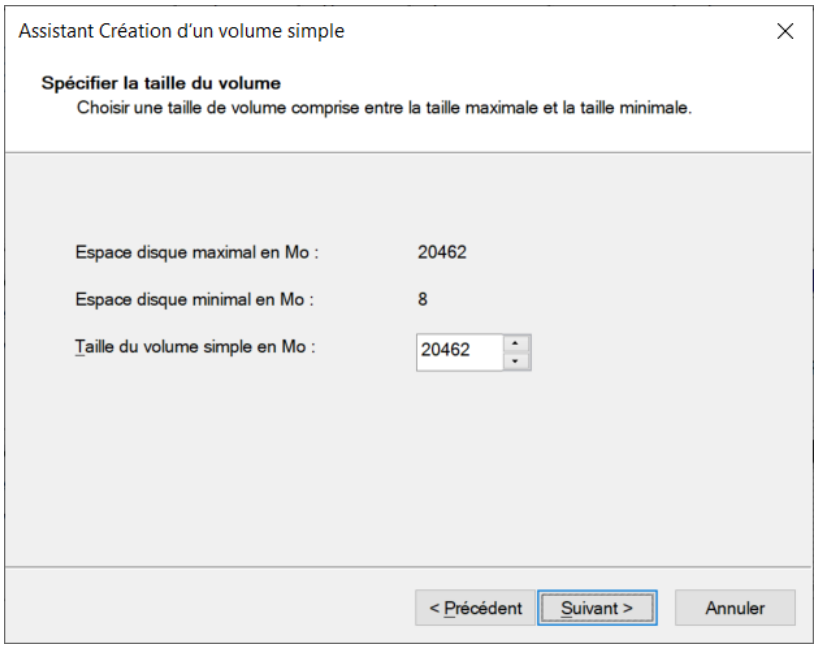
Le disque est hors connexion.
Dans la section "**Disque 1**", vous devez utiliser le bouton droit de la souris pour sélectionner l'option "**En ligne**".

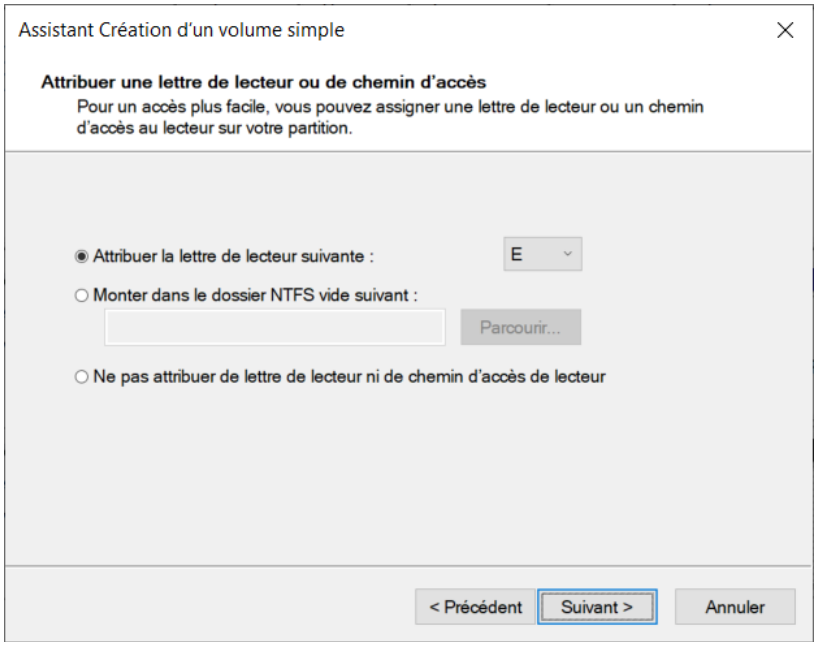
Dans la section "**Disque 1**", vous devez utiliser le bouton droit de la souris pour sélectionner l'option "**Initialiser le disque**".

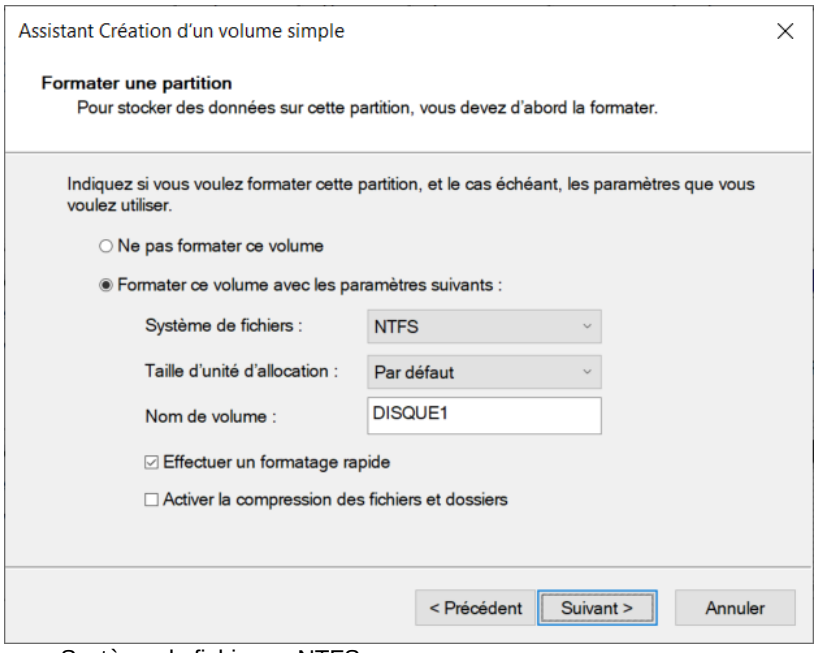


Vous devez sélectionner "**Partition GPT (GUID Partition Table)**".

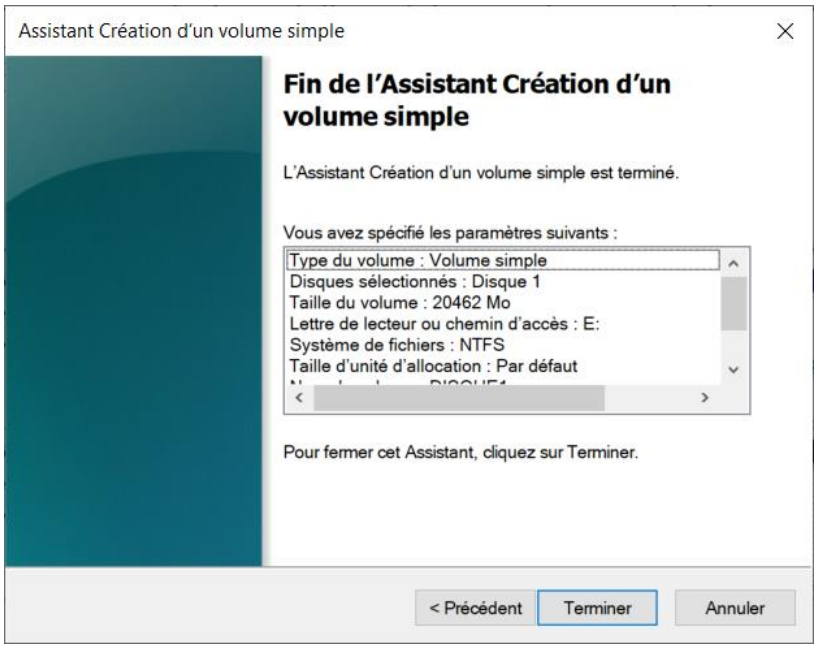
Dans la section de droite, vous devez utiliser le bouton droit de la souris pour sélectionner l'option "**Nouveau volume simple…**"

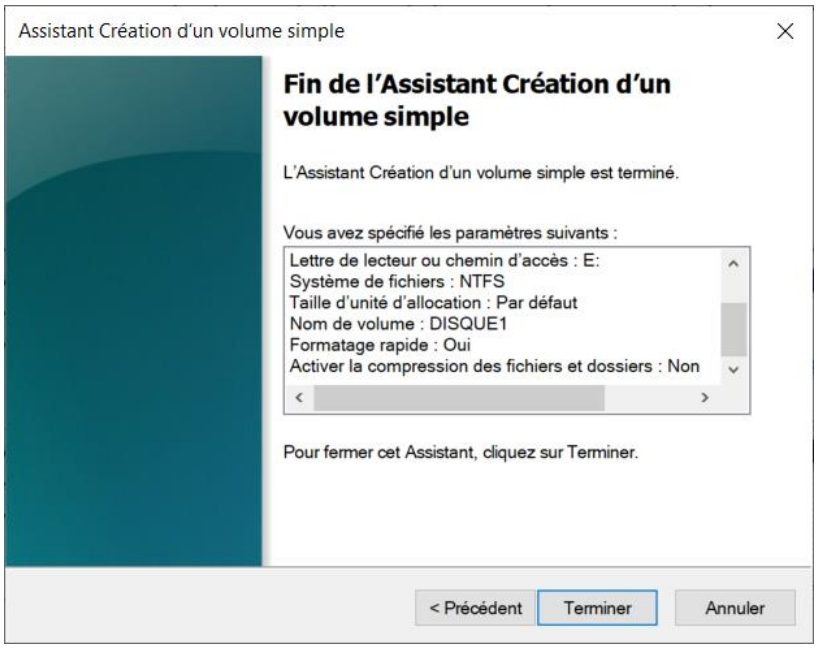




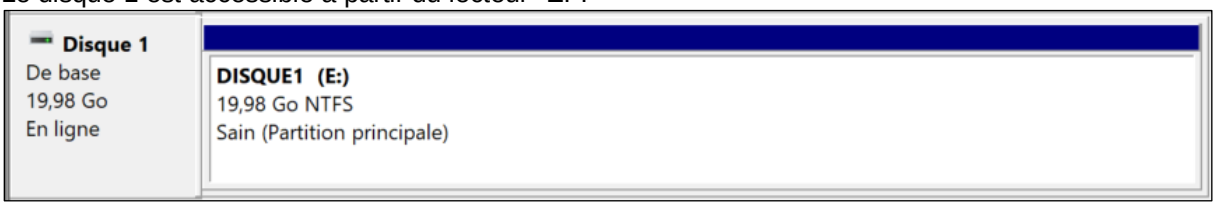


- Système de fichiers = NTFS
- Nom du volume = **DISQUE1**
- Effectuer un formatage rapide





Le disque 1 est accessible à partir du lecteur "E:".



À cette étape, le disque virtuel est présent dans l'Explorateur de fichiers.

Ajouter un disque virtuel VHDX à votre serveur virtuel 2

La seule différence est le nom du fichier VHDX qui sera "**SERVEUR2_disque1.vhdx**".

Activation et configuration du "Mode de session étendu"

Ce laboratoire doit être fait individuellement sur votre serveur réel

Objectifs

- Activer et configurer le "Mode de session étendu" dans Hyper-V

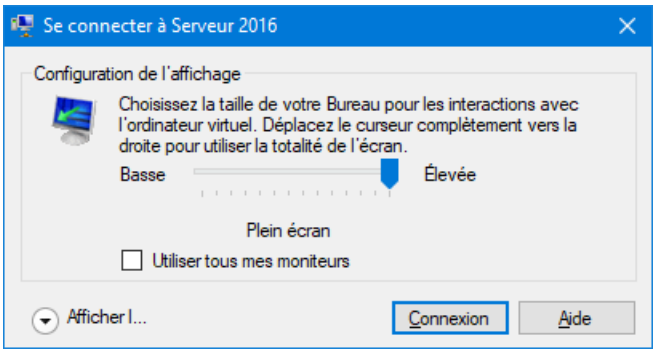
Étape 1 - Paramétrer le service Hyper-V

Ouvrir la console "Gestionnaire Hyper-V"

- Sélectionner votre ordinateur et dans le menu de droite choisir "Paramètres Hyper-V…"
  Dans la section "Stratégie de mode de session étendu" cocher "Autoriser le mode de session étendu"
  Dans la section "Mode de session étendu" cocher "Utiliser le mode de session étendu"

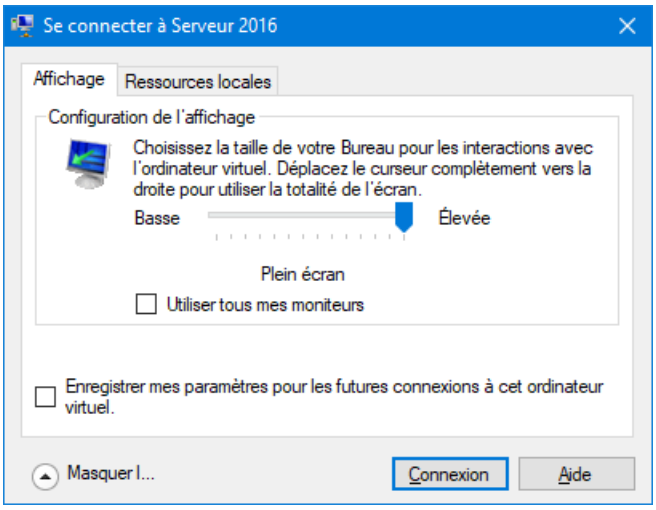
Étape 2 - Utilisation du "Mode de session étendu"

Étant donné que le "Mode session étendu" est actif, lors du démarrage de notre machine virtuelle on pourra configurer la résolution de l'affichage et sélectionner les lecteurs que nous voulons utiliser.

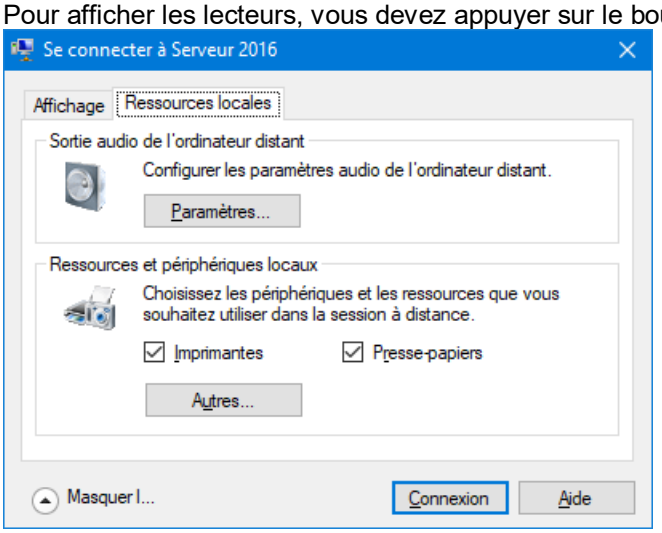


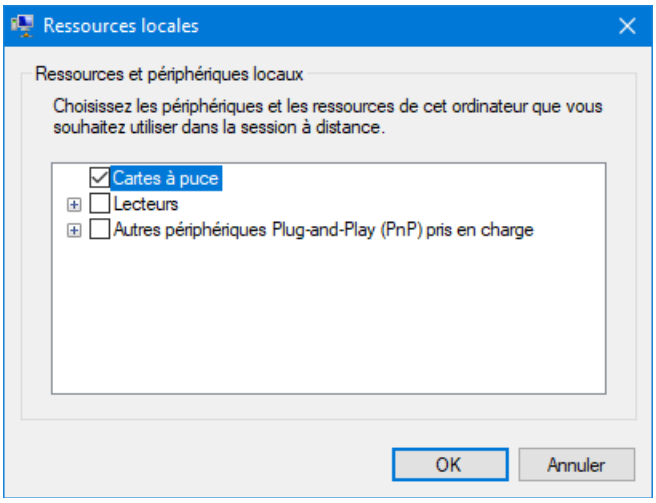
suggestion: choisir la taille de votre Bureau pour "Plein écran"

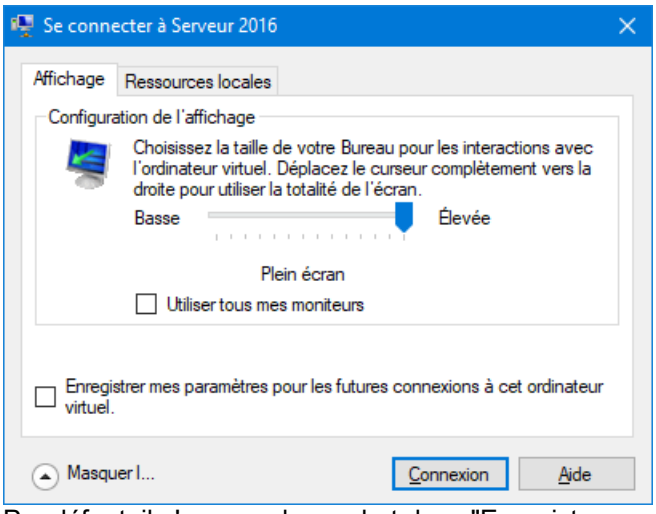
On a des options supplémentaires si vous appuyez sur la flèche "Afficher les"



Pour afficher les lecteurs, vous devez appuyer sur le bouton "Autres…"







Par défaut, il n'y a pas de crochet dans "Enregistrer mes paramètres pour les futures …"

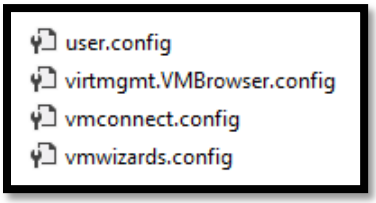
IMPORTANT: Si on enregistre les paramètres du mode de session étendu, la fenêtre ne s'affichera plus.

Voici la commande qui permet de forcer l'affichage de la fenêtre du mode de session étendu.
vmconnect.exe 127.0.0.1 "Serveur 2016" /edit

127.0.0.1	Nom_serveur
"Serveur 2016"	Nom_ordinateur_virtuel
/edit	permet d'accéder aux paramètres de connexion

<u>Pour information seulement</u>
Si on enregistre les paramètres, le fichier **vmconnect.rdp.{virtual machine GUID}.config** sera créé dans le dossier **C:\Users\%USERNAME%\AppData\Roaming\Microsoft\Windows\Hyper-V\Client\1.0**
NOTE: %USERNAME% est une variable d'environnement qui contient votre nom d'utilisateur

Avant d'enregistrer la configuration

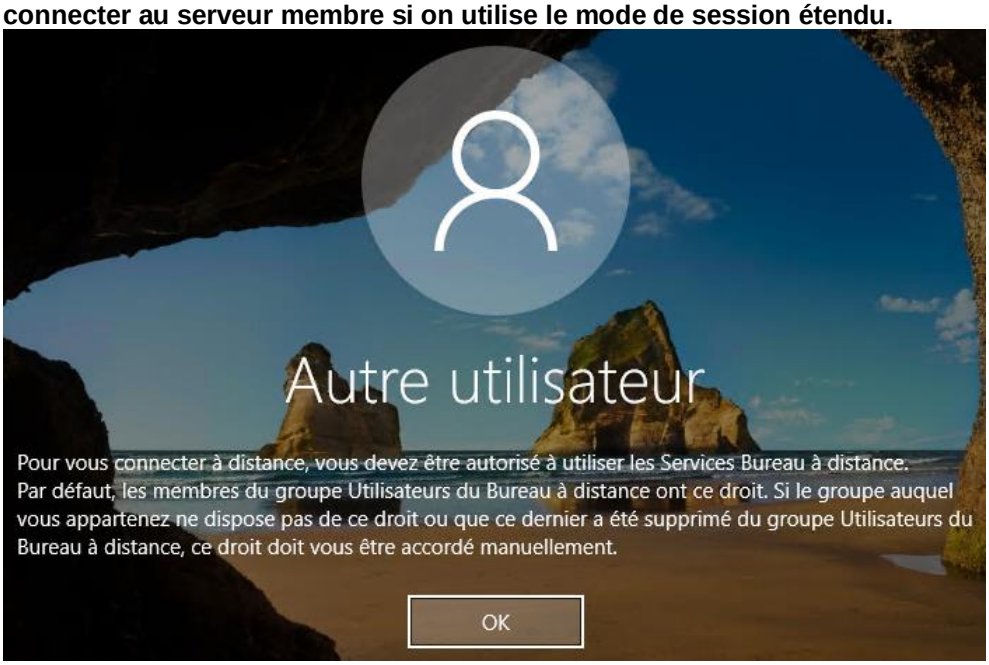


Après l'enregistrement de la configuration

Le fichier **vmconnect.rdp.{virtual machine GUID}.config** contient les paramètres de la configuration du mode de session étendu d'un ordinateur virtuel.

PROBLÈME

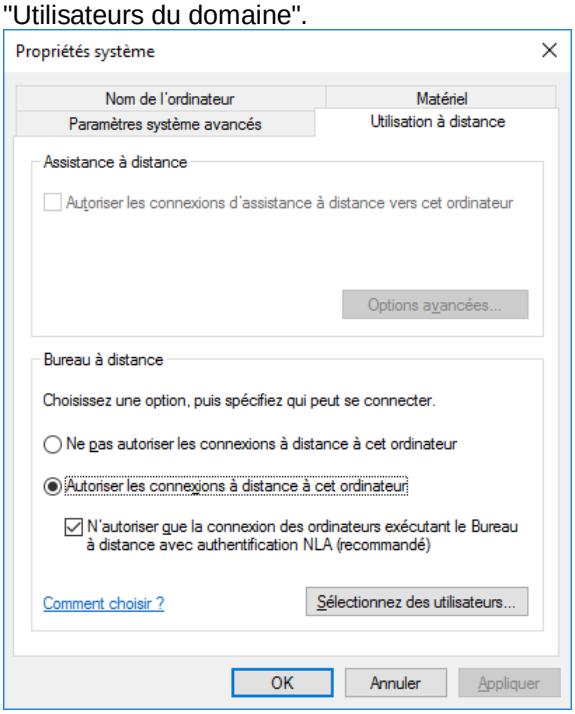
Les utilisateurs qui ne sont pas membre du groupe "Administrateurs" ne peuvent pas se connecter au serveur membre si on utilise le mode de session étendu.

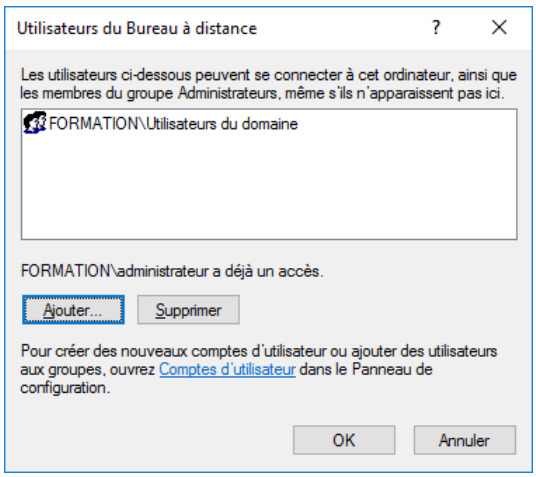


SOLUTION SIMPLE À CE PROBLÈME

Modification à effectuer sur le serveur membre

Dans "Propriétés système" cliquer sur le bouton "Sélectionnez des utilisateurs…" et ajouter le groupe "Utilisateurs du domaine".





Introduction à Active Directory

Réseau centralisé (Laboratoire 3b)

Le type de réseau avec lequel nous allons travailler dans le cours de Réseau 2 est un réseau dit centralisé.

Réseau centralisé : une seule base de données (Active Directory AD) contenant les objets (ordinateurs, usagers….) du réseau. Cette base de données est installée sur un serveur du réseau. Le serveur surlequel est installé Active Directory s'appelle le contrôleur de Domaine.

Un réseau centralisé permet facilement le partage des espaces disques (X, T…), des applications, des imprimantes ET la gestion des objets (ordinateurs, usagers….) du réseau.

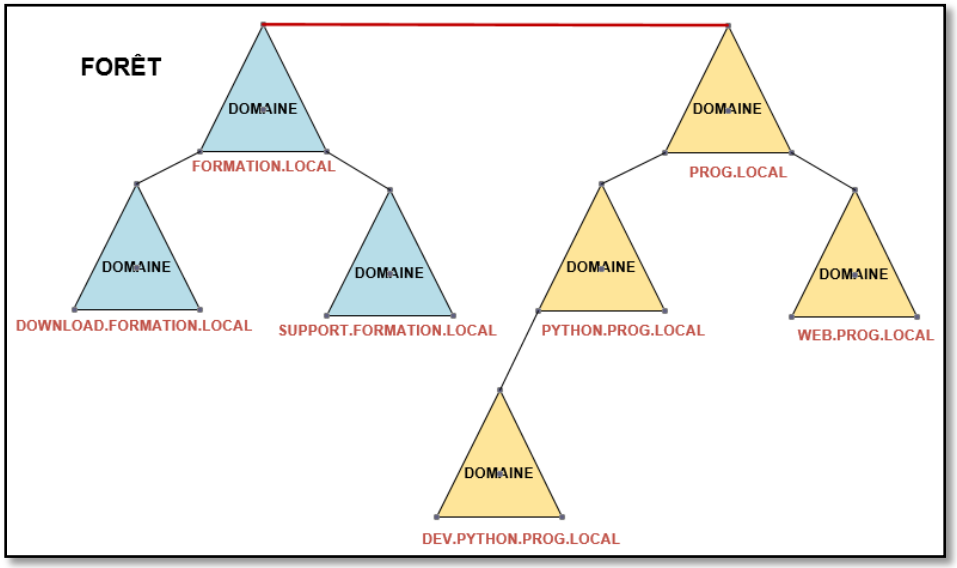
En d'autres termes, Active Directory facilite la centralisation des ressources et de la gestion, ainsi que l'authentification et l'autorisation des utilisateurs.

Installation d'Active Directory : Il faut choisir un nom de domaine pour le réseau.

Les réseaux Microsoft utilisent des notions similaires à celles du DNS. Avec un DNS, on crée de zone qui correspond à un domaine (Web) et on met des enregistrements A dans la zone. Les enregistrements A font le lien entre un hôte (www) et l'adresse IP d'un site Web.

Un réseau Microsoft a un nom de domaine et utilise un DNS dans lequel il y a des enregistrements A. Les enregistrements A font le lien entre un hôte (un ordinateur) et l'adresse IP de cet ordinateur.

On a même la notion de sous-domaine dans un réseau Microsoft.



Cette "Forêt" est constituée de deux "Arbres".

Joindre les ordinateurs au domaine :

Dans le cours de Réseau 2, c'est le serveur1 qui sera le contrôleur de domaine.

Même si le serveur1 et le serveur2 sont reliés par un commutateur (switch), le fait d'installer un AD sur le serveur1 ne lie pas automatiquement le serveur2 au domaine. Pour ce faire, il faut JOINDRE le serveur2 au domaine (vous aurez les étapes dans le laboratoire).

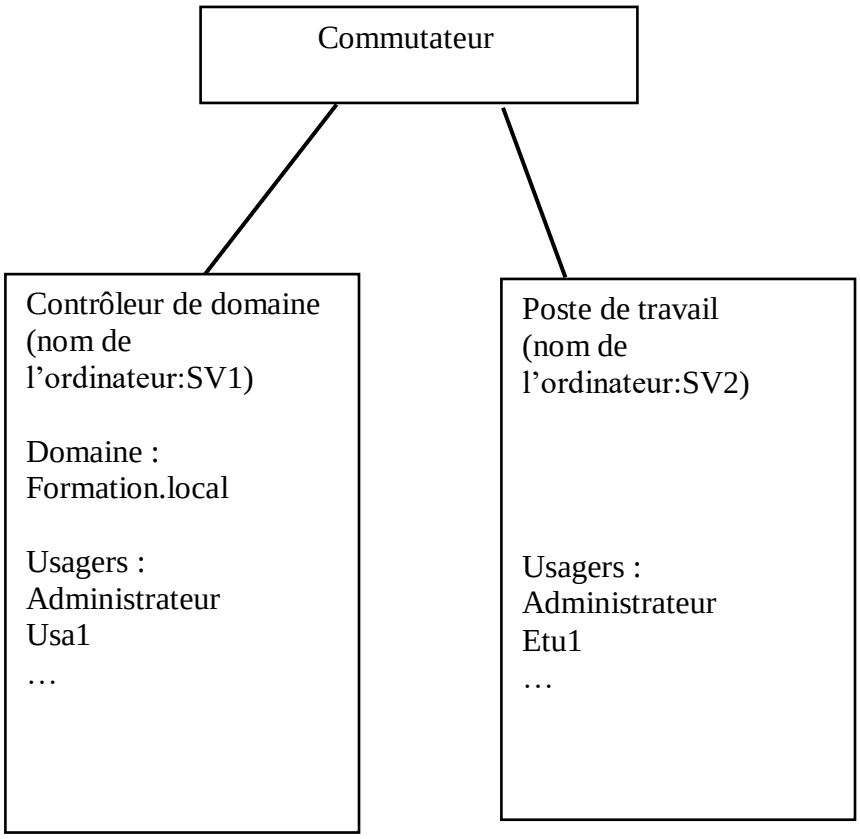
On peut joindre au domaine un poste client (ex : Win 10) ou un serveur (ex : serveur Web). Ce serveur devient alors un serveur membre (et non un contrôleur de domaine).

Sur chaque ordinateur du domaine (sauf le contrôleur de domaine), il y a une double connexion possible (double login), c'est -à-dire qu'on peut se connecter localement ou au réseau sur chaque ordinateur (sauf le contrôleur de domaine).

Connexion locale : je n'ai pas accès aux lecteurs réseau (c,est comme si j'avais débranché la câble réseau). Je me connecte localement sur le poste.
Connexion locale : nom de l'ordinateur\usager ou .\usager

Connexion réseau : je me connecte avec un utilisateur réseau dont les informations sont sur le contrôleur de domaine. Connexion : nom NETBIOS du domaine\usager

Exemple :



Si je veux me connecter au réseau comme Usa1 (de n'importe quel ordinateur du domaine) :

Formation\Usa1

Si je veux me connecter au réseau comme Administrateur (de n'importe quel ordinateur du domaine) :

Formation\Administrateur

Si je veux me connecter localement sur SV2 comme Etu1 (je ne peux pas me connecter sur SV2 localement comme Usa1, ce compte n'existe pas localement sur SV2)

SV2\ Etu1

Si je veux me connecter localement sur SV2 comme Administrateur (qui existe aussi dans le domaine)

SV2\ Administrateur

Quand aurait-on besoin de se connecter localement ?
Si on est Administrateur de domaine, on a tous les droits sur l'ordinateur local, donc pas besoin de se connecter localement pour effectuer une tâche.

Cas particulier : au CVM, je peux me connecter au réseau du collège (au domaine reseau.cvm), mais je ne suis pas administrateur de domaine sur le serveur du collège. Je ne suis pas administrateur local sur mon poste de travail à mon bureau. Si je voulais installer une application localement, il faudrait que je connaisse le mot de passe administrateur local et que je me connecte .\administrateur pour pouvoir faire mon installation localement.

Installation de l'Active Directory en mode graphique

Dans la console "**Gestionnaire de serveur**" installer le rôle "**Services AD DS**"

- L'installation de l'Active Directory nécessite l'installation du rôle DNS
- L'installation de l'Active Directory exige un redémarrage.
- Après le redémarrage, nous devons compléter la post-installation afin de "promouvoir ce serveur en contrôleur de domaine".
- On doit choisir "Ajouter une nouvelle forêt" si c'est un nouveau domaine

---

Le répertoire **C:\Windows\NTDS** contient les journaux de transaction, les logs et les fichiers temporaires utiles au fonctionnement de l'Active Directory.

Le fichier **ntds.dit** constitue la base de données de l'Active Directory.

- Ce fichier est présent sur chaque contrôleur de domaine.
- DIT signifie (Directory Information Tree)

Les partages administratifs (NETLOGON, SYSVOL)

Après l'installation de l'Active Directory nous avons accès à deux partages

- \\FORMATION.LOCAL\NETLOGON C:\Windows\SYSVOL\sysvol\FORMATION.LOCAL\scripts
- \\FORMATION.LOCAL\SYSVOL C:\Windows\SYSVOL\sysvol

On remarque que le partage NETLOGON est un sous-dossier du partage SYSVOL.

Le partage SYSVOL est répliqué entre les contrôleurs de domaine.

Le partage SYSVOL contient deux dossiers

- \\FORMATION.LOCAL\SYSVOL\FORMATION.LOCAL**Policies**
  Ce dossier contient les GPO.
- \\FORMATION.LOCAL\SYSVOL\FORMATION.LOCAL**scripts**
  Ce dossier contient les les scripts de logon.

Gestion des utilisateurs, des groupes, des ordinateurs et des unités d'organisation (Laboratoire 3c)

La console "**Utilisateurs et ordinateurs Active Directory**" permet la gestion des utilisateurs, des groupes, des ordinateurs et des unités d'organisation.

La console "**Centre d'administration Active Directory**" permet la gestion des utilisateurs, des groupes, des ordinateurs et des unités d'organisation. Cette console est plus récente que la console UOAD (Utilisateurs et ordinateurs Active Directory).

Structure de l'AD (Laboratoires 3d et 3e)

Active Directory (AD) est une organisation hiérarchisée d'objets. La racine est le domaine.

Les unités d'organisations
La console "Utilisateurs et ordinateurs Active Directory" contient les objets de l'Active Directory.
Par défaut, il existe seulement une "Unité d'organisation"

- Domain controllers contient les contrôleurs de domaine

Les autres objets de la console sont des "Conteneurs" et voici une courte liste

- Builtin contient que des groupes
- Computers contient des ordinateurs
- Users contient des utilisateurs et des groupes

IMPORTANT: un administrateur peut ajouter des "Unité d'organisation" mais il ne peut pas ajouter des "Conteneurs".

Installation de l'Active Directory

Ce laboratoire doit être fait individuellement sur les ordinateurs virtuels 1 et 2

Objectifs

- Installer le rôle AD DS (Active Directory Domain Services)
- Joindre un ordinateur au domaine

Matériels

- L'ordinateur réel avec deux ordinateurs virtuels

Étape 1 - Vérification du contrôleur de domaine

Démarrer l'ordinateur virtuel "SERVEUR1" et connectez-vous avec l'usager Administrateur.

Vérifier la configuration du serveur virtuel "SERVEUR1"

- Le nom du serveur est **SERVEUR1**
- Modifier la configuration DNS de la carte réseau
  DNS: **127.0.0.1**

Étape 2 - Création d'un domaine

Démarrer la console "Gestionnaire de serveur", dans le menu "Gérer"

Installer le rôle "Services AD DS" avec les fonctionnalités

- Lire les informations qui vous sont données
- Cocher "Redémarrer automatiquement le serveur de destination, si nécessaire"
- Quand l'installation sera terminée, fermer la fenêtre et cliquez sur le triangle jaune qui est en haut et à droite, à côté du drapeau
  - Il faut effectuer la configuration post-déploiement (choisir "Promouvoir ce serveur en contrôleur de domaine")

Répondre aux questions de la post-configuration du domaine

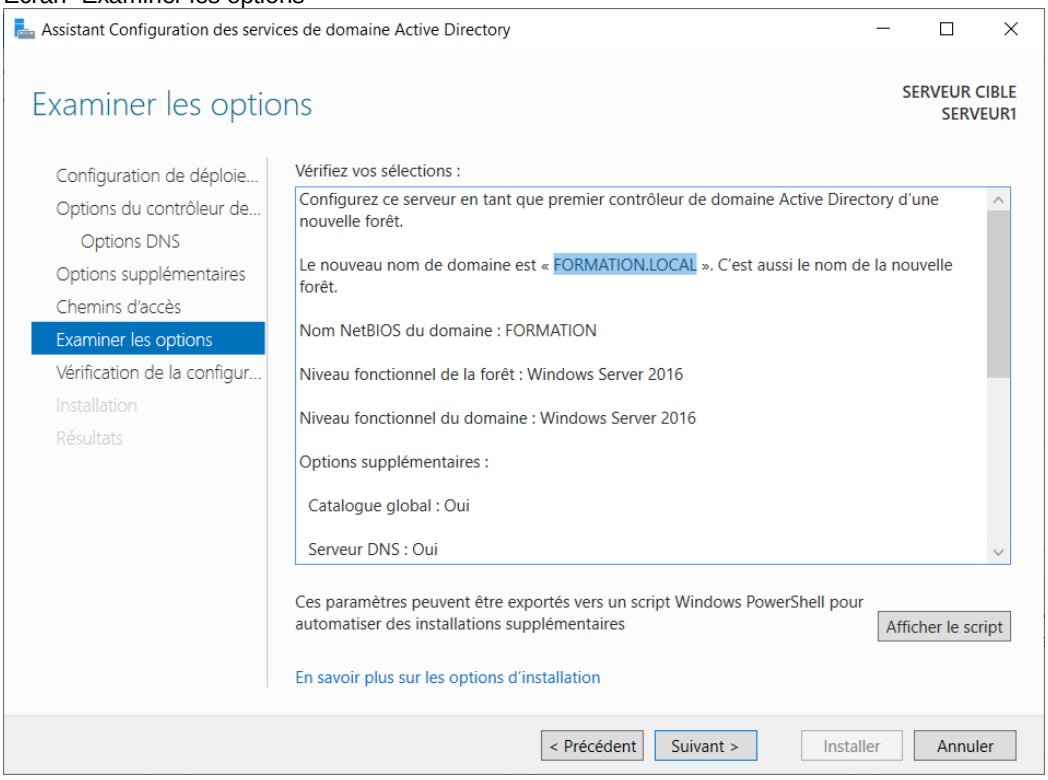
- Écran "Configuration de déploiement"
  - Choisir "Ajouter une nouvelle forêt"
  - Le nom de domaine racine est **FORMATION.LOCAL**

- Écran "Options du contrôleur de domaine"
  - Taper le mot de passe du mode de restauration des services d'annuaire (DSRM)
    Mot de passe: AAAaaa111
    Confirmer le mot de passe: AAAaaa111 et Suivant
  - Vous aurez un avertissement "Il est impossible de créer une délégation pour ce serveur DNS …"
    Ne vous en occupez pas et cliquer sur le bouton "Suivant".

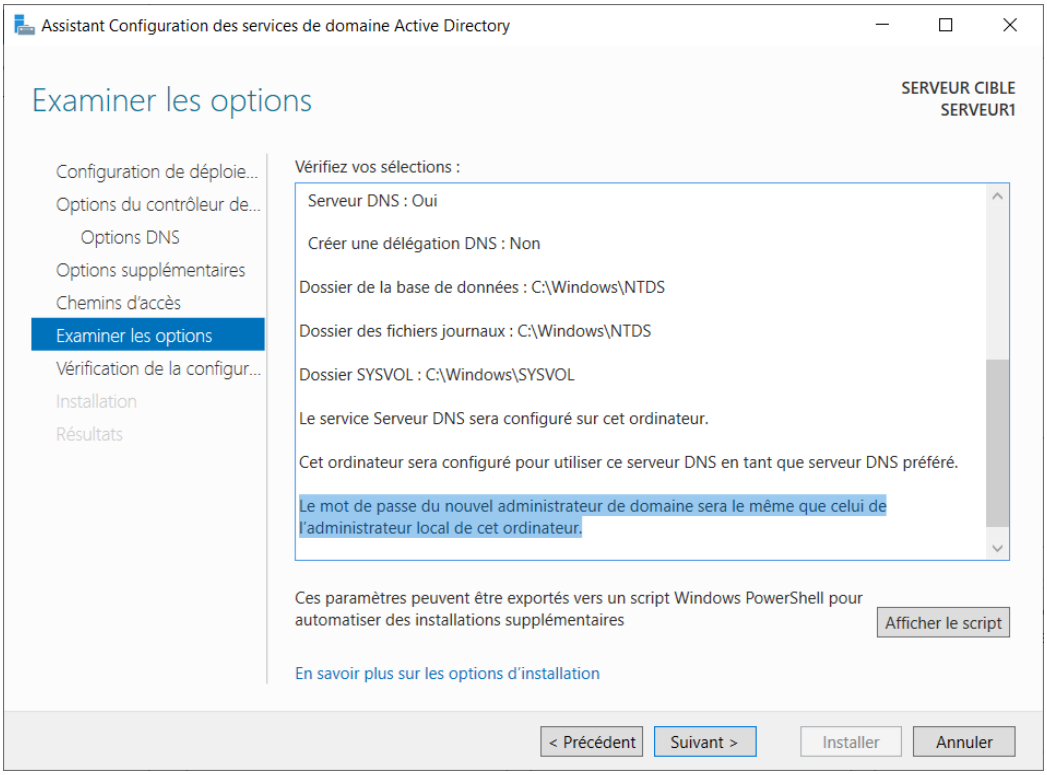
- Écran "Options supplémentaires"
  - Le nom de domaine NetBIOS: **FORMATION**
    Vous aurez besoin de cette information pour joindre le SERVEUR2 au domaine.

- Écran "Chemins d'accès"
  - Ne pas changer l'emplacement des dossiers proposés par défaut

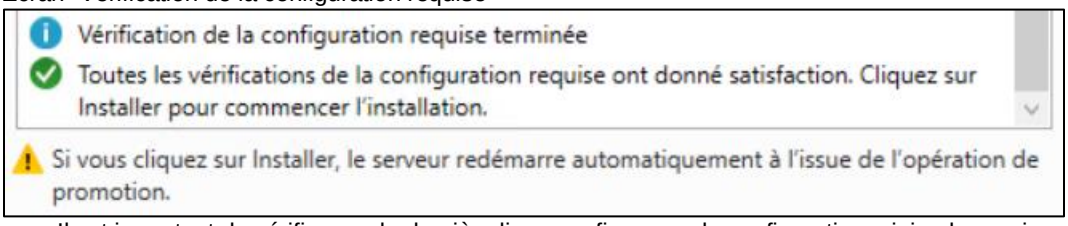
- Écran "Examiner les options"



Il est important de vérifier que le nom du domaine est **FORMATION.LOCAL**



- Écran "Vérification de la configuration requise"



  - Il est important de vérifier que la dernière ligne confirme que la configuration minimale requise est satisfaite.
  - Démarrer l'installation
  - Le serveur va redémarrer à la fin de l'installation.

Étape 3 - Validation du domaine

Connectez-vous sur votre contrôleur de domaine (Administrateur et même mot de passe que vous avez donné à votre VM)

Dans la console DNS (Gestionnaire de serveur / Outils / DNS)

- Sélectionner "SERVEUR1 / Zones de recherche directes" / FORMATION.LOCAL"
  pour vérifier la présence d'un enregistrement de type "A" au nom de votre serveur SERVEUR1

Dans une invite de commandes exécuter la commande suivante

```
ping SERVEUR1.FORMATION.LOCAL -4
```

Vérifier la présence des partages: NETLOGON et SYSVOL
Rappel du cours de C22 : 2 façons de faire :

- exécuter "net share" dans une "Invite de commandes"
  ou bien
- Win-r
  \\FORMATION.LOCAL\NETLOGON
  \\FORMATION.LOCAL\SYSVOL

Étape 4 - Joindre un ordinateur au domaine

Connectez-vous sur l'ordinateur virtuel "SERVEUR2" en utilisant le compte Administrateur.

Vérifier la configuration du serveur virtuel "SERVEUR2"

- Le nom du serveur 2 est **SERVEUR2**
- Modifier la configuration DNS de la carte réseau
  DNS: 192.168.1.10 (adresse IP du "Contrôleur de Domaine")

Joindre l'ordinateur à votre domaine

- Win-x, Système
- Informations système
- Dans la section « Nom de l'ordinateur », cliquer sur « Modifier les paramètres »
- Dans l'onglet « Nom de l'ordinateur », cliquer sur Modifier
- Choisir l'option "Membre d'un Domaine:"
- Taper le nom de domaine NetBIOS qui est **FORMATION**

Authentifiez-vous avec le compte Administrateur du domaine: FORMATION\administrateur

- Après le message de bienvenue faites redémarrer votre ordinateur client. (Ce message de bienvenue peut prendre plusieurs secondes à apparaître…)

Toujours sur votre serveur virtuel "SERVEUR2", connectez-vous au domaine, de la façon suivante:

Par défaut la connexion sur un ordinateur se fait toujours localement. Nous voulons nous connecter à notre nouveau domaine avec le compte administrateur. Comme ce compte existe aussi localement le processus de login va toujours privilégier le compte local au compte réseau. Il faut toujours prêter attention lors du login et bien vérifier les informations inscrites devant le nom de l'utilisateur. En cas de doute la commande "whoami" nous permet de savoir exactement notre identification.

Exemples d'identification si l'ordinateur se nomme SERVEUR2 et membre du domaine FORMATION.LOCAL

- SERVEUR2\Administrateur utilise le compte Administrateur local
- .\Administrateur utilise le compte Administrateur local
- FORMATION\Administrateur utilise le compte Administrateur du domaine

Note : la commande WHOAMI vous permet toujours de savoir si vous êtes connecté localement ou au domaine.

.

Fin du laboratoire….Les annexes qui suivent sont pour information seulement

ANNEXE 1

Comment sortir un ordinateur du domaine

Vous devez utiliser ces commandes en cas de besoin seulement.

Cette commande permet de sortir un ordinateur du domaine si le contrôleur de domaine existe.
Cette commande demande le mot de passe du compte " **NomDuDomaine\Administrateur**".

```
Remove-Computer -ComputerName NomDeLordinateur `
                -UnjoinDomainCredential NomDuDomaine\Administrateur `
                -WorkgroupName NomDuGroupeDeTravail `
                -Force `
                -Restart
```

Cette commande permet de sortir un ordinateur du domaine même si le contrôleur de domaine n'existe plus.
Après l'exécution de la commande, le groupe de travail portera le nom du domaine.

```
netdom.exe remove NomDeLordinateur /Domain:NomDuDomaine /Force /Reboot
```

ANNEXE 2

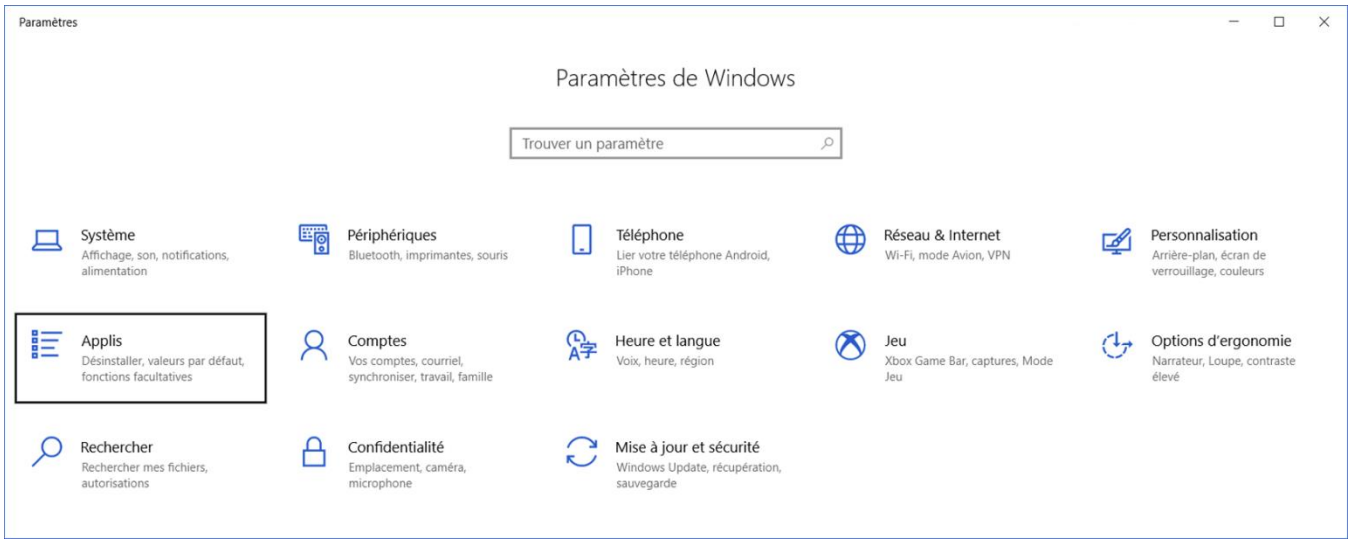
RSAT (Remote Server Administration Tools)

Si un ordinateur qui est membre du domaine utilise "**Windows 10**", il est possible d'installer les différentes consoles de gestion d'un serveur Windows.

À partir de "Windows 10 version 1809" RSAT est inclus dans les "Fonctionnalités facultatives".

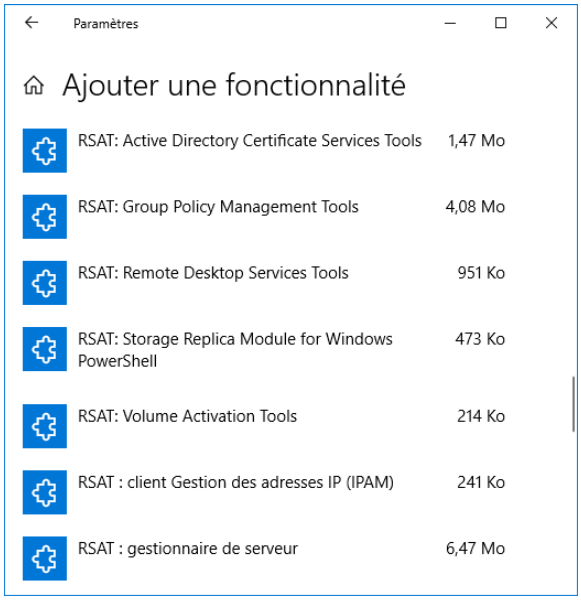
Installation de RSAT en utilisant l'environnement graphique

Dans la console "**Paramètres de Windows**"



On doit ouvrir "**Applis**"

- cliquer sur "**Fonctionnalités facultatives**"
  - cliquer sur "**Ajouter une fonctionnalité**"



Création d'un utilisateur dans l'Active Directory

Ce laboratoire doit être fait individuellement sur l'ordinateur virtuel 2

Objectifs

- Introduction à la console "Utilisateurs et Ordinateurs Active Directory (UOAD)"
- Création d'un utilisateur avec les mêmes caractéristiques que l'utilisateur "Administrateur" du domaine

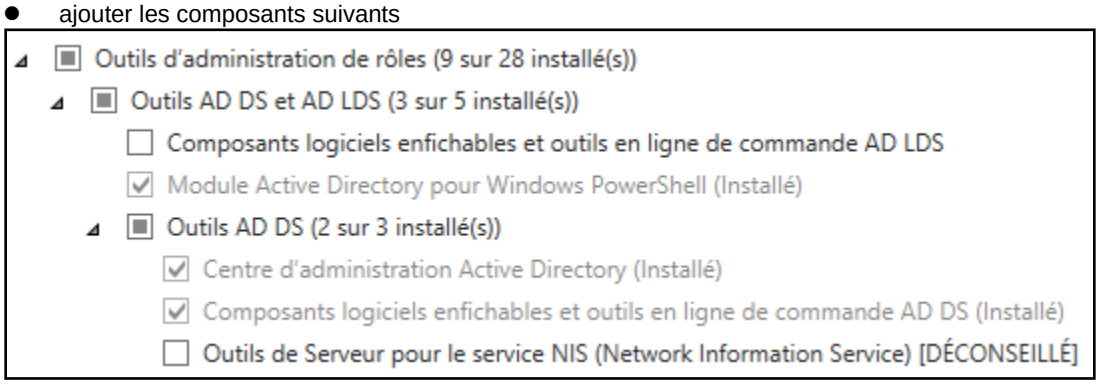
Étape 1 – Installer les outils d'administration sur le serveur 2

De préférence, chaque étudiant doit travailler sur un serveur membre.
Je vous suggère de ne pas travailler directement sur le contrôleur de domaine.

Les étapes suivantes vont vous indiquer comment installer la console UOAD sur le serveur2.

Dans la fonctionnalité "Outils d'administration de serveur distant", développer "Outils d'administration de rôles".

- ajouter les composants suivants



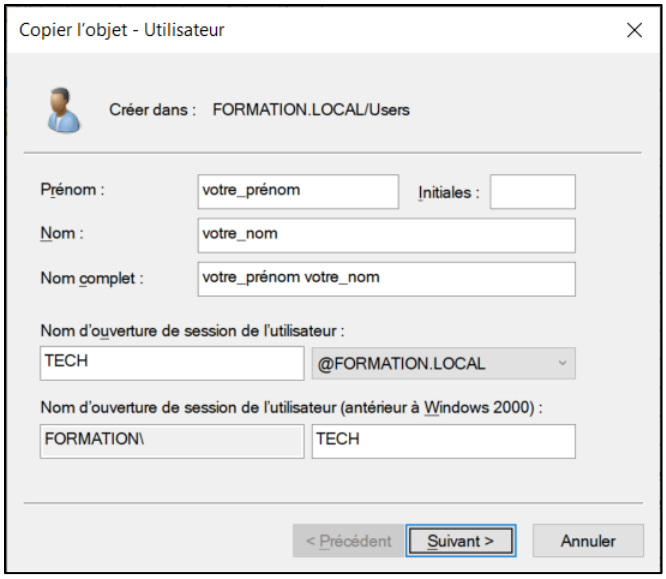
Vérification: si tout s'est bien passé, la console UOAD (Utilisateurs et Ordinateurs Active Directory) sera installée sur le serveur2. Vous la trouverez en regardant dans les Outils d'administration.

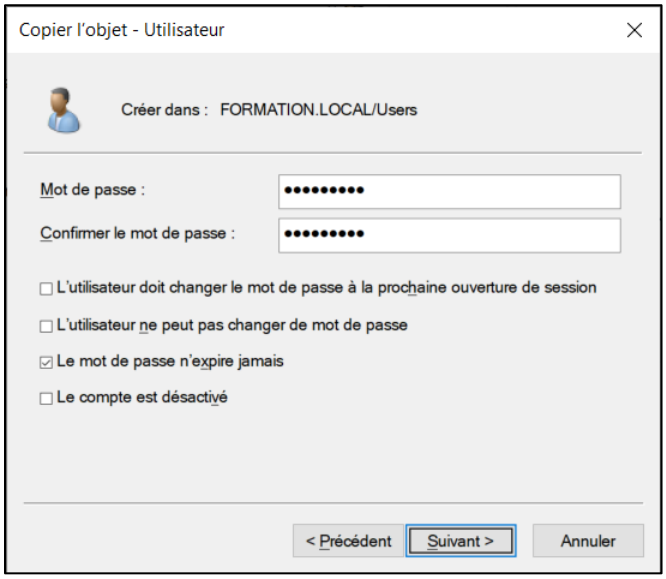
Étape 2 - Création de l'utilisateur TECH

Toujours en étant connecté sur l'ordinateur virtuel "SERVEUR2" avec l'utilisateur "FORMATION\Administrateur", démarrer la console "Utilisateurs et Ordinateurs Active Directory"

Le but est de créer un nouvel utilisateur FORMATION\TECH qui aura les mêmes caractéristiques que le compte "Administrateur" du domaine.

- À l'intérieur du conteneur "**Users**" sélectionner l'utilisateur "**Administrateur**"
  Dans le menu contextuel, il faut sélectionner "**Copier...**"



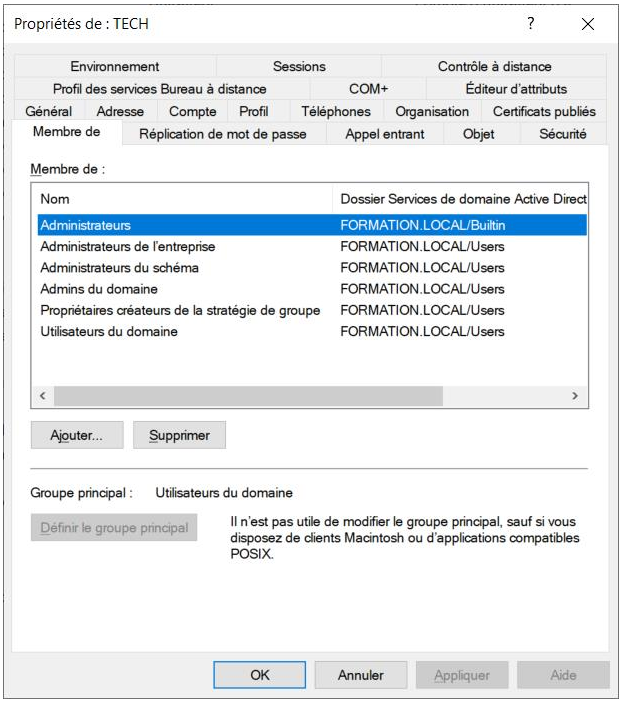


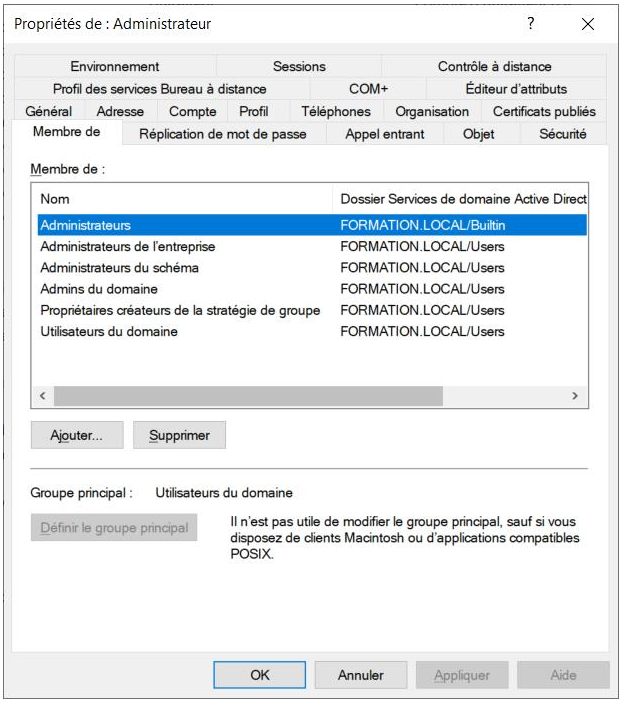
Mot de passe

- **Le mot de passe doit avoir au moins 6 caractères et être complexe.**

Cocher "**Le mot de passe n'expire jamais**"

- Vérification des deux utilisateurs





Les deux utilisateurs sont membres des mêmes groupes.

À PARTIR DE MAINTENANT VOUS NE DEVEZ PLUS UTILISER LE COMPTE "ADMINISTRATEUR" DU DOMAINE.

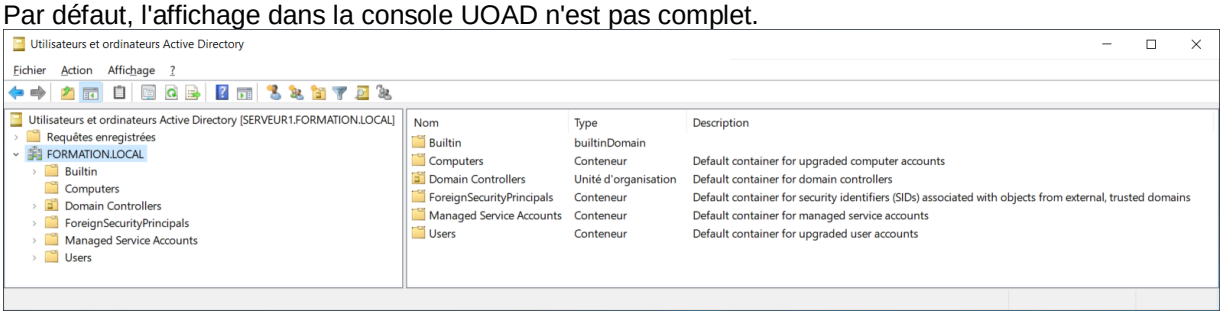
Étape 3 – Renommer le compte "Administrateur" local

Pour éviter de se connecter par erreur avec le compte "Administrateur" local sur le SERVEUR2, je vous conseille de renommer le compte "Administrateur" local.

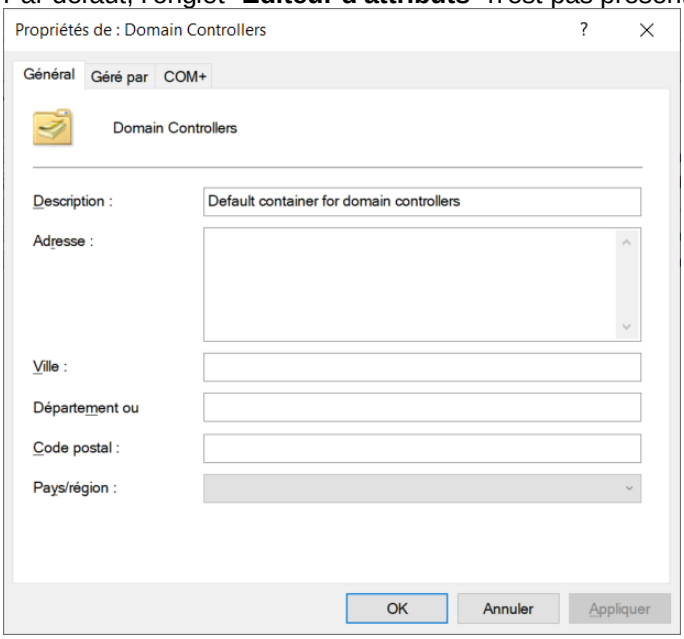
```
# Le code doit s'exécuter sur le SERVEUR2
$NewAdminName = "AdminLocal"

$Objet = Get-LocalUser | Where-Object { $PSItem.SID -like "S-1-5-21-*-500" }
Rename-LocalUser -InputObject $Objet -NewName $NewAdminName
```

La console "Utilisateurs et ordinateurs Active Directory" (UOAD)

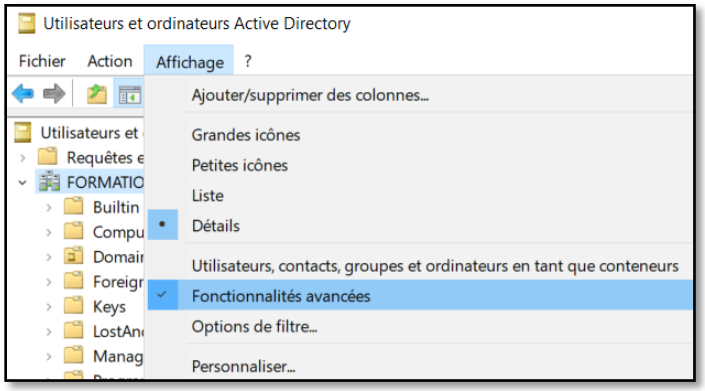
Par défaut, l'affichage dans la console UOAD n'est pas complet.

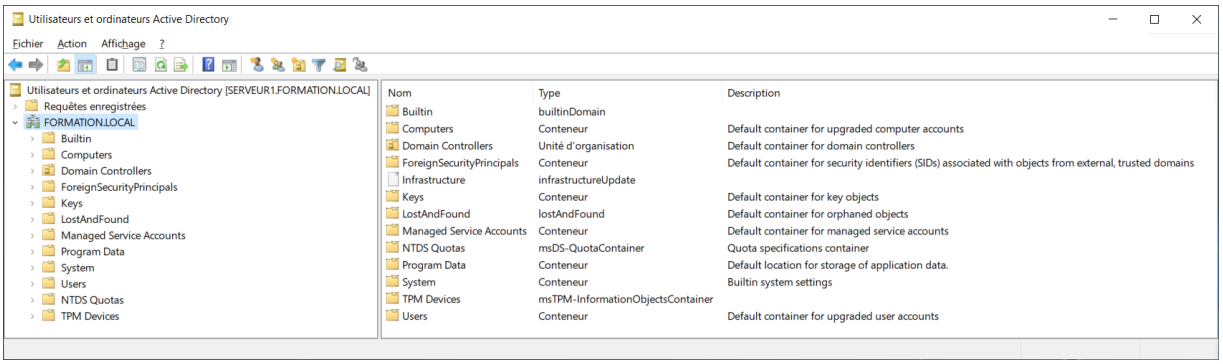


Chaque objet de l'Active Directory a des attributs.
Par défaut, l'onglet "**Éditeur d'attributs**" n'est pas présent dans les propriétés d'un objet.

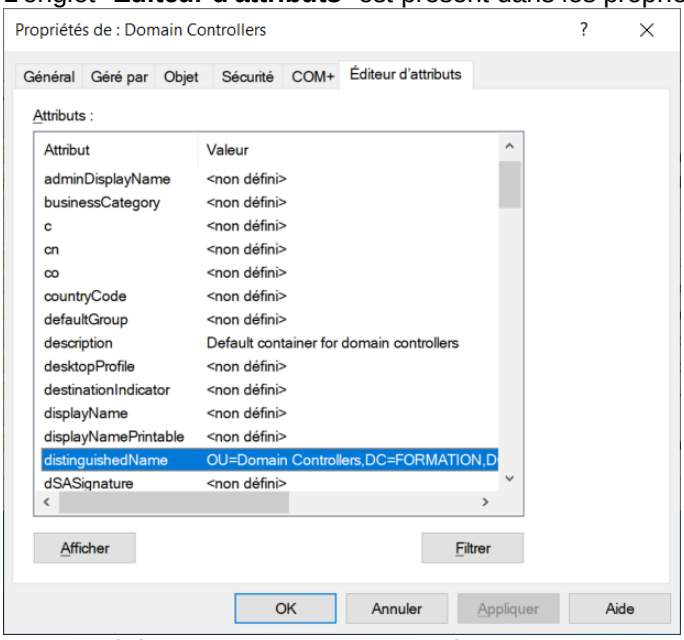


Dans le menu "**Affichage**" de la console UOAD, il faut activer "**Fonctionnalités avancées**"





L'onglet "**Éditeur d'attributs**" est présent dans les propriétés d'un objet.



La propriété DistinguishedName est très importante.
La propriété DistinguishedName est unique pour chaque objet dans l'Active Directory.

L'attribut DistinguishedName

La valeur de l'attribut **DistinguishedName** est utilisée dans plusieurs commandes PowerShell.

Attribut	Description
DC	Domain Component
CN	Common Name
OU	Organizational Unit

Exemple de valeur pour l'attribut "distinguishedName" pour une "Unité d'organisation"

L'unité d'organisation "Domain Controllers" existe par défaut.
OU=Domain Controllers,DC=FORMATION,DC=LOCAL

NOTE: Un Administrateur peut créer des unités d'organisation.
OU=UTILISATEURS,DC= FORMATION,DC= LOCAL

Exemple de valeur pour l'attribut "distinguishedName" pour un "Conteneur"

Voici les "Conteneurs" les plus utilisés dans le cours:
CN=Builtin,DC=FORMATION,DC=LOCAL
CN=Computers,DC= FORMATION,DC= LOCAL
CN=Users,DC= FORMATION,DC= LOCAL

NOTE: Un Administrateur ne peut pas créer des objets "Conteneur".

Exemple de valeur pour l'attribut "distinguishedName" pour un "Utilisateur"

Si l'utilisateur Richard est dans l'unité d'organisation UTILISATEURS.
CN=Richard,OU=UTILISATEURS,DC= FORMATION,DC= LOCAL

Exemple de valeur pour l'attribut "distinguishedName" pour un "Groupe"

Si le groupe PROFESSEURS est dans l'unité d'organisation UTILISATEURS.
CN=PROFESSEURS,OU=UTILISATEURS,DC= FORMATION,DC= LOCAL

Exemple de valeur pour l'attribut "distinguishedName" pour un "Ordinateur"

L'ordinateur SERVEUR1 est dans l'unité d'organisation "Domain Controllers".
CN=SERVEUR1,OU=Domain Controllers,DC= FORMATION,DC= LOCAL

L'ordinateur SERVEUR2 est dans le conteneur "Computers".
CN=SERVEUR2,CN=Computers,DC= FORMATION,DC= LOCAL

Les propriétés des unités d'organisation

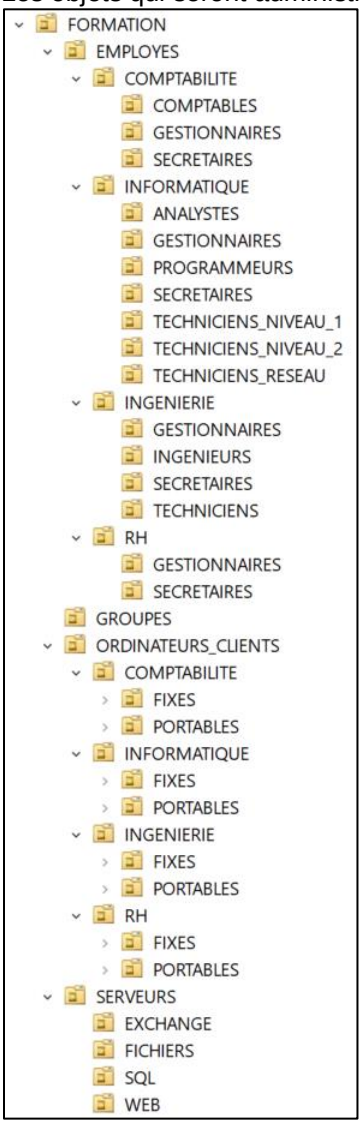
Ce laboratoire doit être fait individuellement sur le "SERVEUR VIRTUEL 2"

Objectifs

- Utiliser l'onglet "Éditeur d'attribut"
- Utiliser plusieurs attributs de l'objet "Unité d'organisation"

Le but d'une Unité d'Organisation

Les objets qui seront administrés de la même manière devront être placés dans la même unité d'organisation.



- La conception d'UO aura une incidence sur le déploiement des stratégies de groupe.
- Il est important de ne pas mélanger les comptes utilisateurs et ordinateurs dans une même UO.
- Ne gardez pas les utilisateurs et les ordinateurs dans les conteneurs par défaut.
- Il est important d'activer le paramètre "Protéger le conteneur contre une suppression accidentelle".

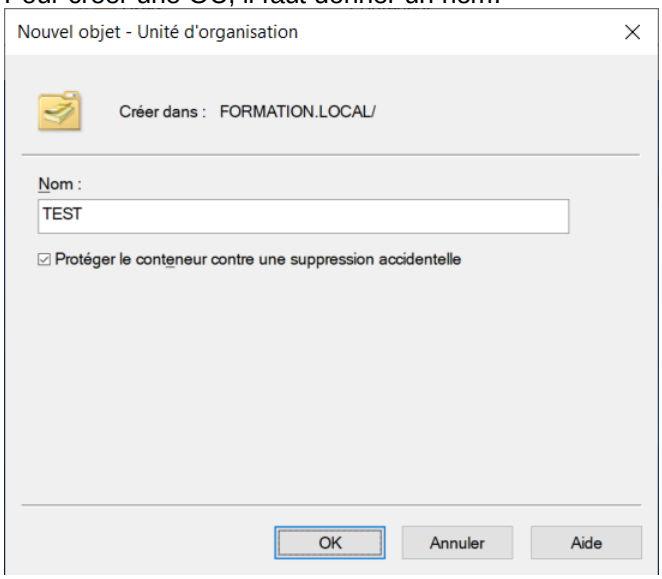
Les attributs d'une unité d'organisation

Ouvrir la console "Utilisateurs et ordinateurs Active Directory" (UOAD)

- Vérifier que votre affichage est en "Fonctionnalités Avancées"

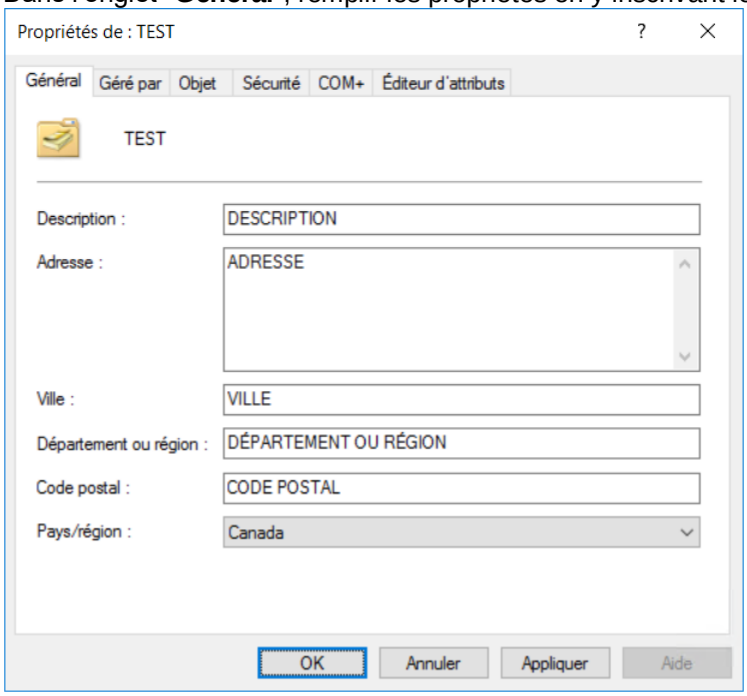
Sélectionner le domaine "FORMATION\LOCAL" et créer l'unité d'organisation "TEST".

Pour créer une OU, il faut donner un nom.

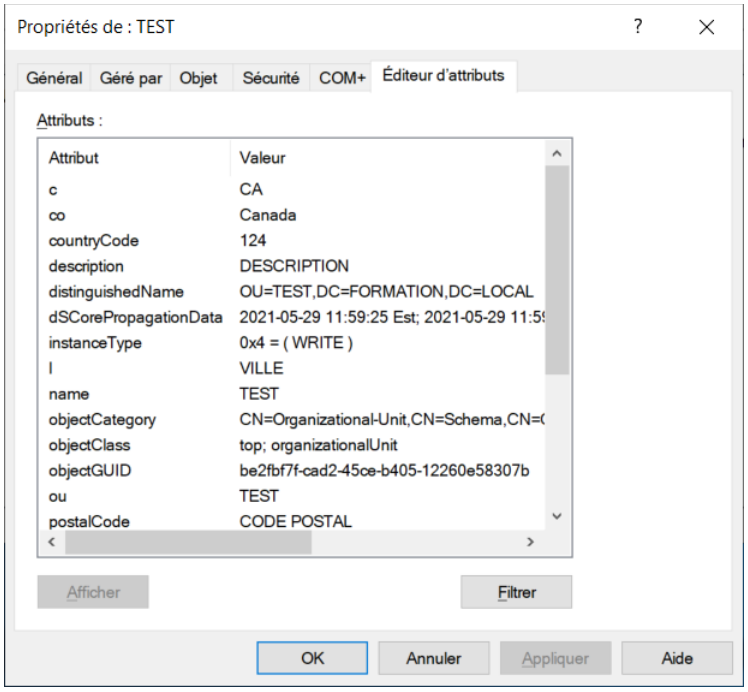


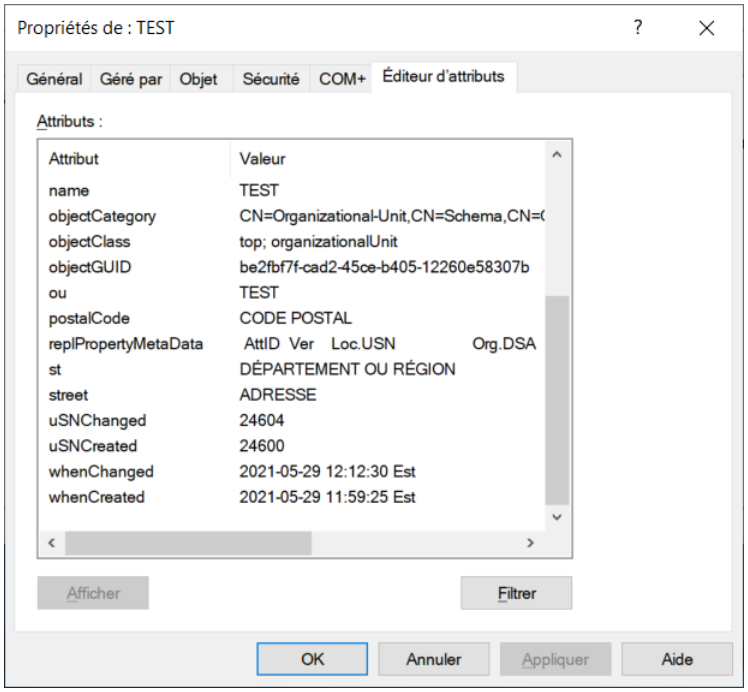
Cliquer sur le bouton "**OK**"

Après la création de l'unité d'organisation "TEST", nous allons modifier les propriétés.
Dans l'onglet "**Général**", remplir les propriétés en y inscrivant les valeurs suivantes.



Cliquer sur le bouton "**Filtrer**" et cocher "**Afficher uniquement les attributs ayant des valeurs**"
En utilisant l'onglet "**Éditeur d'attributs**", trouvez le nom des attributs qui contiennent vos valeurs.





Nom du champ dans l'onglet "Général"	Nom de l'attribut
Description	**description**
Adresse	**street**
Ville	**l**
Département ou région	**st**
Code postal	**postalCode**
Pays/région	**note: il y a trois attributs par pays** **c=CA** **co=Canada** **countryCode=124**

Trouvez la valeur pour les attributs suivants:

Nom de l'attribut	Valeur de l'attribut
DistinguishedName	**OU=TEST,DC=FORMATION,DC=LOCAL**
name	**test**
ou	**test**
WhenChanged	**2019-05-29 12:12:30 Est**
WhenCreated	**2019-05-29 11:59:25 Est**

Étape 2 - Programmation d'une UO avec PowerShell ISE

Il existe 4 cmdlet spécifiques pour la gestion des unités d'organisation.

- Get-ADOrganizationalUnit
- New-ADOrganizationalUnit
- Remove-ADOrganizationalUnit
- Set-ADOrganizationalUnit

Avant de créer une unité d'organisation par programmation PowerShell, il faut faire le lien entre le nom des attributs dans l'Active Directory et le nom des propriétés dans PowerShell.

Nom du champ dans l'onglet "Général"	Nom de la propriété dans PowerShell pour le cmdlet New-ADOrganizationalUnit
Description	**-Description**
Adresse	**-Street**
Ville	**-City**
Département ou région	**-State**
Code postal	**-PostalCode**
Pays/région	**-Country** **Il est préférable d'utiliser le paramètre -OtherAttributes avec les trois attributs c, co et countryCode.**

Exemple de création d'une unité d'organisation avec PowerShell.

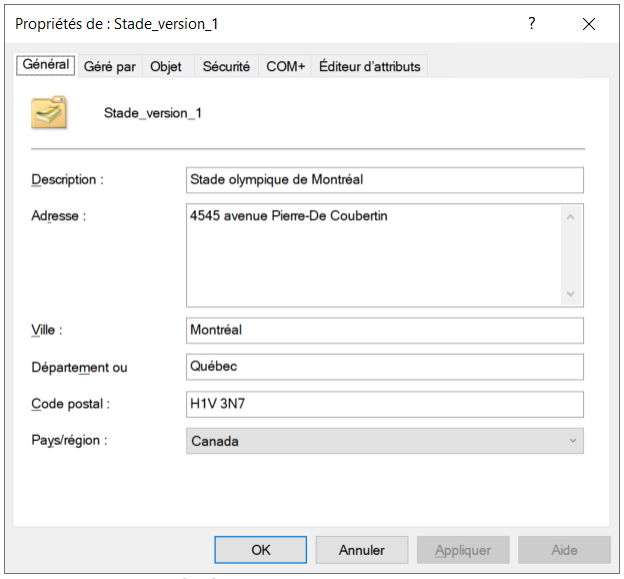
Méthode avec le paramètre -Country

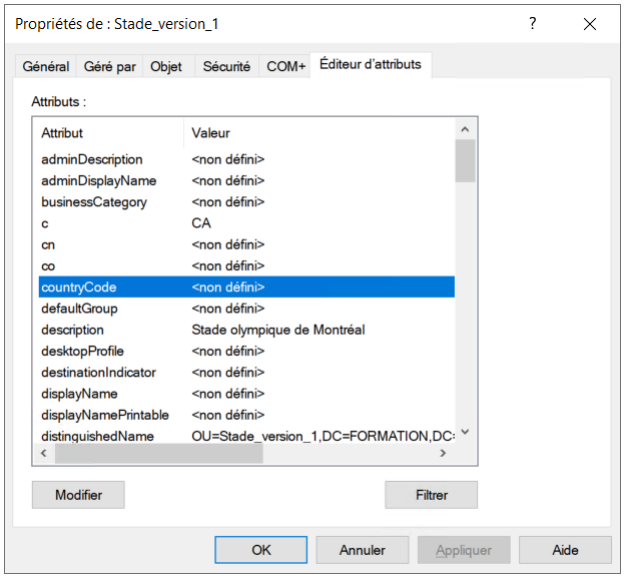
On veut créer l'unité organisationnelle "Stade_version_1" qui sera directement sous le domaine.

Le paramètre **-Path** utilise la valeur de l'attribut **DistinguishedName**

Pour le pays, nous utiliserons le paramètre **-Country**

```
New-ADOrganizationalUnit -Name "Stade_version_1" `
                         -Path "DC=formation,DC=local" `
                         -Description "Stade olympique de Montréal" `
                         -street "4545 avenue Pierre-De Coubertin" `
                         -City "Montréal" `
                         -PostalCode "H1V 3N7" `
                         -State "Québec" `
                         -Country "CA" `
                         -ProtectedFromAccidentalDeletion $false
```





Sur l'onglet "Général" le pays est Canada.
Sur l'onglet "Éditeur d'attributs" les attributs co et countryCode sont vides.
Nous avons un comportement qui est différent que celui de l'environnement graphique.

Malheureusement, le paramètre -Country configure seulement l'attribut c.

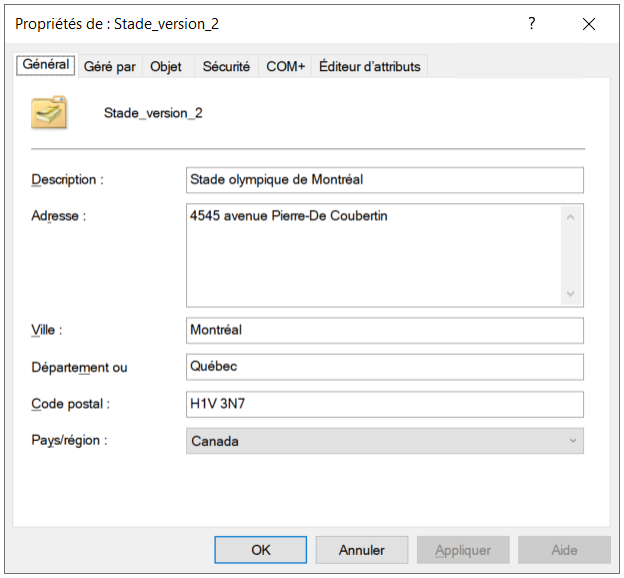
Méthode avec le paramètre -OtherAttributes

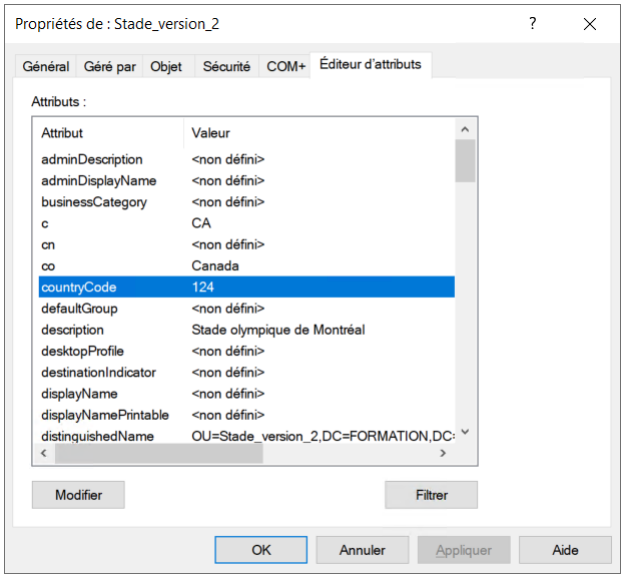
On veut créer l'unité organisationnelle "Stade_version_2" qui sera directement sous le domaine.

Le paramètre **-Path** utilise la valeur de l'attribut **DistinguishedName**

Pour le pays, nous utiliserons le paramètre **-OtherAttributes** avec les trois attributs **c**, **co** et **countryCode**

```
New-ADOrganizationalUnit -Name "Stade_version_2" `
                         -Path "DC=formation,DC=local" `
                         -Description "Stade olympique de Montréal" `
                         -street "4545 avenue Pierre-De Coubertin" `
                         -City "Montréal" `
                         -PostalCode "H1V 3N7" `
                         -State "Québec" `
                         -OtherAttributes @{'c'="CA";
                                            'co'="Canada";
                                            'countryCode'=124} `
                         -ProtectedFromAccidentalDeletion $false
```



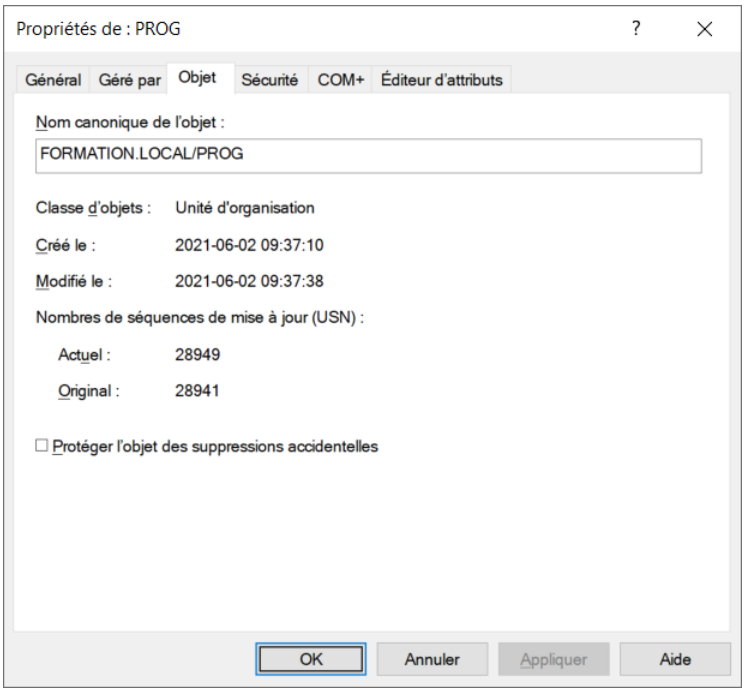


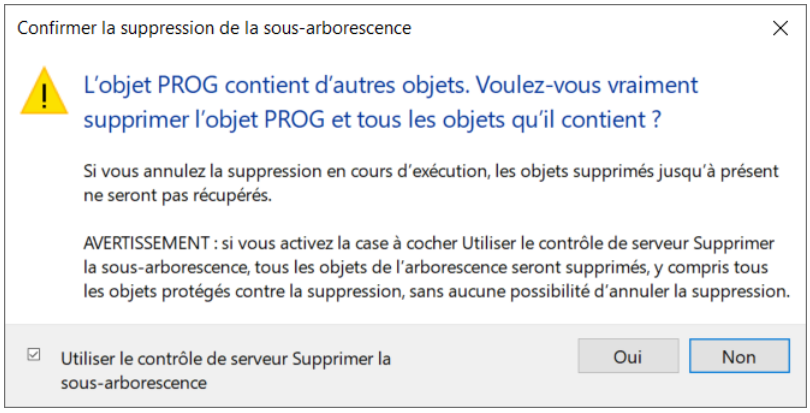
Sur l'onglet "Général" le pays est Canada.
Sur l'onglet "Éditeur d'attributs" les attributs co et countryCode ont des valeurs.
Nous avons le même comportement que celui de l'environnement graphique.

Conclusion: il est préférable d'utiliser le paramètre -OtherAttributes avec les trois attributs c, co et countryCode.

Supprimer une unité d'organisation par l'environnement graphique

Pour supprimer une OU qui a une arborescence, il faut enlever le crochet "**Protéger l'objet des suppressions accidentelles**" sur la OU à supprimer.





Pour supprimer l'arborescence de la OU,
il faut cocher "**Utiliser le contrôle de serveur Supprimer la sous-arborescence**".

Supprimer une unité d'organisation par programmation PowerShell

Pour supprimer une OU qui a une arborescence par programmation PowerShell.

Le paramètre **-Identity** utilise la valeur de l'attribut **DistinguishedName**

 # Enlève la protection contre la suppression accidentelle

```
Set-ADOrganizationalUnit -Identity "OU=prog,DC=formation,DC=local" `
                         -ProtectedFromAccidentalDeletion $false
```

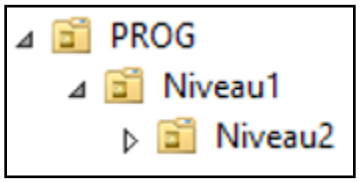
Pour supprimer l'unité d'organisation "**PROG**"

```
Remove-ADOrganizationalUnit -Identity "OU=prog,DC=formation,DC=local" `
                            -Confirm:$false `
                            -Recursive
```

Le paramètre **-Recursive** est obligatoire si l'unité d'organisation n'est pas vide.

Exercice

En utilisant la console "PowerShell ISE", écrire un script pour créer la structure suivante directement sous le domaine. Inclure le code qui permet de détruire la structure si elle existe déjà.



Votre code doit utiliser **Remove-ADOrganizationalUnit** et **New-ADOrganizationalUnit** qui sont dans le module ActiveDirectory.

```
#=====================================================
# Solution programmation des unitÃ©s d'organisation
# 1 septembre 2023
#=====================================================
Clear-Host

#=====================================================
# Code pour dÃ©truire la UO "PROG"
#EnlÃ¨ve la protection contre la suppression accidentelle
#Si la OU n'existe pas il y aura un message d'erreur
Set-ADOrganizationalUnit -Identity "OU=prog,DC=formation,DC=local" `
                         -ProtectedFromAccidentalDeletion $false

#Suppression de la OU "PROG" qui est sous le domaine FORMATION.LOCAL
#Si la OU n'existe pas il y aura un message d'erreur
Remove-ADOrganizationalUnit -Identity "OU=prog,DC=formation,DC=local" `
                            -Confirm:$false `
                            -Recursive

#=====================================================
#CrÃ©ation de la OU "PROG" sous le domaine FORMATION.LOCAL
New-ADOrganizationalUnit -Name "PROG" `
                         -Path "DC=formation,DC=local" `
                         -Description "PROG est sous le domaine" `
                         -ProtectedFromAccidentalDeletion $false
```

```
#CrÃ©ation de la OU "Niveau1" sous "PROG"
New-ADOrganizationalUnit -Name "Niveau1" `
                         -Path "OU=PROG,DC=formation,DC=local" `
                         -Description "Niveau1 est sous PROG" `
                         -ProtectedFromAccidentalDeletion $false

#CrÃ©ation de la OU "Niveau2" sous "Niveau1"
New-ADOrganizationalUnit -Name "Niveau2" `
                         -Path "OU=Niveau1,OU=PROG,DC=formation,DC=local" `
                         -Description "Niveau2 est sous Niveau1" `
                         -ProtectedFromAccidentalDeletion $false
```

Théorie Scripts PowerShell

Gestion d'un système d'exploitation

Définition
Shell: couche logicielle qui sert d'interface au système d'exploitation. S'applique pour tous les systèmes d'exploitation.

Cette couche forme la coquille du OS, d'où son nom.

Cmd et Powershell sont des shells. Linux a... Shell (ou bash). Les interfaces graphiques le sont aussi, mais on les appelle rarement comment tel.

Powershell: interface shell sous forme de lignes de commandes et un langage de script (possibilité d'intégrer des variables, des boucles, des tests, …aux lignes de commandes)

Powershell vise à remplacer cmd. Il vise l'administration de système et fait très bien son travail dans cette optique.

1. Les variables en Powershell

a) Les noms de variables sont précédés d'un signe de dollar. Ex: $Variable1, $Nombre…

b) Affectation: $MaVariable = 5

c) Affichage: $MaVariable

d) Écrire du texte: Write-Host "Ma variable vaut " $MaVariable

e) Les minuscules et les majuscules n'importent pas (insensibilité à la case).
$mavariable = $MaVariable = $MAVARIABLE

f) Les variables dans Powershell sont faiblement typées. Cela signifie que contrairement au C++, l'information que la variable contient définit son type (pas de int, float, char, etc..)

```
$Var1 = "Marie"
$Var1
$Var1 =  $Var1 + " Pierre"
$Var1
$Var1 = 12
$Var1
```

Résultat :

```
Marie
Marie Pierre
12
```

2. Les expressions conditionnelles

Tableau des équivalences PowerShell avec C++

-lt <

-le <=

-gt >

-ge >=

-eq ==

-ne !=

-not !

-and &&

-or ||

On peut mettre le résultat d'une expression conditionnelle (vrai ou faux) dans une variable :

```
$Comparaison = 5 -eq 5

Write-Host "Est-ce que 5 est égal à 5? Réponse:" $Comparaison
```

Résultat :

```
Est-ce que 5 est égal à 5? Réponse: True
```

---

3. Boucle for

Syntaxe : get-help about_for

```
Syntax
      The following shows the For statement syntax.


          for (<init>; <condition>; <repeat>)
          {<statement list>}
```

Exemple1 :

```
for ($i=1; $i -le 10; $i=$i+2)
{
       $i
}
```

Résultat:

```
1
3
5
7
9
```

Exemple2 : (on peut concaténer une variable dans une chaine de caractères)

```
for ($i=1; $i -le 5; $i++)
{
Write-Host "Tour de boucle $i"
}
```

Résultat:

```
Tour de boucle 1
Tour de boucle 2
Tour de boucle 3
Tour de boucle 4
Tour de boucle 5
```

4. Boucle while

Syntaxe : get-help about_while

```
  The following shows the while statement syntax:

        while (<condition>){<statement list>}
```

Exemple :

```
$i = 1
while ($i -le 10)
{
        $i
        $i = $i+2
}
```

Résultat:

```
1
3
5
7
9
```

5. Instruction conditionnelle if

```
  Syntax
    The following example shows the If statement syntax:


        if (<test1>)
            {<statement list 1>}
        [elseif (<test2>)
            {<statement list 2>}]
        [else
            {<statement list 3>}]
```

Exemple 1:

```
for ($i=0; $i -le 10; $i++)
{
        $Parite = $i%2 -eq 0

        if($Parite)
        {
            Write-Host "$i est pair"
        }
        else
        {
            Write-Host "$i est impair"
        }
}
```

Résultat :

```
0 est pair
1 est impair
2 est pair
3 est impair
4 est pair
5 est impair
6 est pair
7 est impair
8 est pair
9 est impair
10 est pair
```

Exemple 2:

```
for ($i=10; $i -ge 0; $i=$i-1)
{
        $Parite = $i%2 -eq 0
        if($i -eq 0)
        {
            Write-Host "On est arrivé à zéro"
        }
        elseif($Parite)
        {
            Write-Host "$i est pair"
        }
        else
        {
            Write-Host "$i est impair"
        }
}
```

Résultat :

```
10 est pair
9 est impair
8 est pair
7 est impair
6 est pair
5 est impair
4 est pair
3 est impair
2 est pair
1 est impair
On est arrivé à zéro
```

6. Branchement switch

```
A basic Switch statement has the following format:

      Switch (<test-value>)
      {
          <condition> {<action>}
          <condition> {<action>}
     }
```

```
# selon la valeur de la variable $test
clear-host

switch ($test)
{
    {$test -eq 1}
    {write-host "la valeur est 1"}

    {$test -eq 2}
    {write-host "la valeur est 2"}

    {$test -eq 3}
    {write-host "la valeur est 3"}
}
```

7. Boucle foreach

La boucle foreach est utilisée pour traiter les collections.

On peut affecter un ensemble d'items à une variable. C'est ce qu'on appelle une collection.

Exemples :

$Nombres = 2,4,6,8 (Ensemble d'entiers)

$Cara = `"a","b","c","d","e"` (Ensemble de caractères)

$Mots = `"Émilie","Marie","Gabrielle"` (Ensemble de caractères)

$Fichier = `Get-Content C:\Temp\FichierPSH.txt` (Ensemble d'enregistrements, (de lignes) du fichier)

La boucle foreach parcourt l'ensemble (la collection) et traite chacun de ses éléments. Elle gère le début et la fin de la collection.

Exemple 1 :

```
# Exemple avec une collection qui contient des entiers
Clear-Host

$Nombres = 4,7,8,2,3,6,5
$i = 0

foreach ($Nombre in $Nombres)
{
$i++
Write-host "Le nombre en position $i est $Nombre"
}
```

Résultat :

```
Le nombre en position 1 est 4
Le nombre en position 2 est 7
Le nombre en position 3 est 8
Le nombre en position 4 est 2
Le nombre en position 5 est 3
Le nombre en position 6 est 6
Le nombre en position 7 est 5
```

Exemple 2 :

```
Clear-Host

# Exemple avec une collection qui contient des caractères

$Mots = "Émilie", "Marie", "Gabrielle"
$i = 0

foreach ($Mot in $Mots)
{
	$i++
	Write-host "Le mot en position $i est $Mot"
}
```

Résultat :

```
Le mot en position 1 est Émilie
Le mot en position 2 est Marie
Le mot en position 3 est Gabrielle
```

Exemple 3 :

```
Clear-Host
# Exemple avec un fichier (collection de lignes)

$fichier = Get-Content C:\Temp\FichierPSH.txt
$i = 0

foreach ($ligne in $fichier)
{
	$i++
	Write-host "La ligne numéro $i du fichier est: $ligne"
}
```

Résultat :

```
La ligne numéro 1 du fichier est: Germinal  Émile Zola
La ligne numéro 2 du fichier est: Cent ans de solitude   Gabriel Garcia Marquez
La ligne numéro 3 du fichier est: Les cerfs-volants   Romain Gary
La ligne numéro 4 du fichier est: L'exil et le royaume   Albert Camus
```

8. Try and Catch

Syntaxe :

```
Try {commande qui ne fonctionne pas}
Catch {Write-Host……}
```

Plus précisément, exemple avec Objet non trouvé :

```
Try {Objet non trouvé}
Catch [Microsoft.ActiveDirectory.Management.AdIdentityNotFoundException]
{
	$message = “objet inexistant”
	Write-Host $message -ForegroundColor Yellow
}
Catch
{
	Write-Warning “ERREUR INATTENDUE”
	Exit
}
```

Notes :

1. pour pouvoir utiliser la commande ICACLS dans une fenêtre PSH, il faut encadrer chaque attribution de droits par des guillemets :
   ex : ICACLS C:\Rep1 /grant "System:(oi)(ci)(F)" "Administrateurs:(oi)(ci)(F)"

2. Commentaire : # au début d'une ligne
   <# et #> pour le début et la fin d'un paragraphe

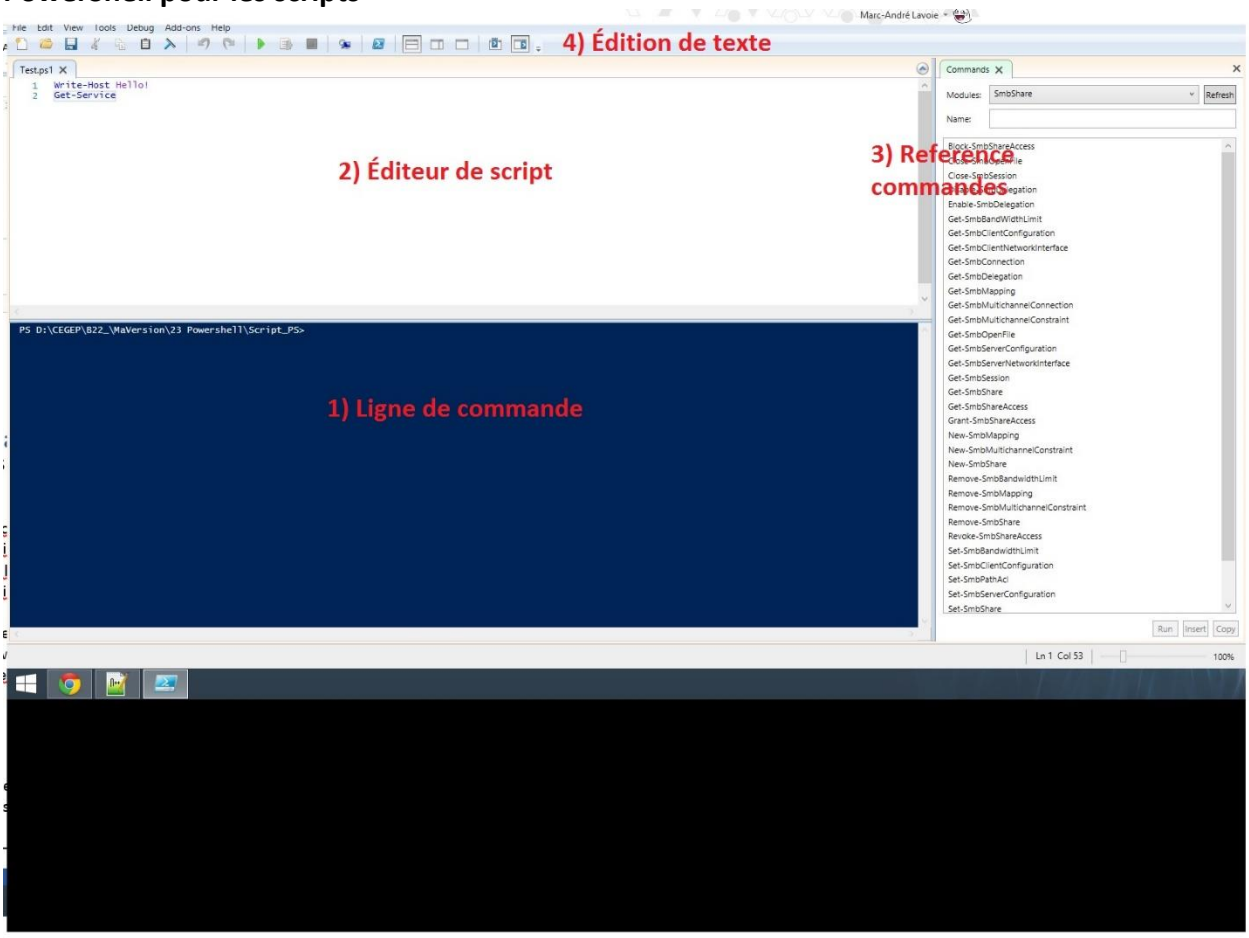
3. Accent grave à la fin de la ligne pour changer de ligne

4. Clear-host au début pour ne pas avoir le code dans la fenêtre de résultats

5. Création de partages en PowerShell : New-SMBShare

6. Aide : get-help, get-help about, documentation, Internet

PowerShell pour les scripts



Programmation d'une structure d'unité d'organisation

Ce laboratoire doit être fait individuellement sur l'ordinateur virtuel 2

Objectifs

- Utiliser un fichier CSV pour créer des unités d'organisation en utilisant PowerShell

Description du travail

Écrire un programme en PowerShell qui créera une structure complexe de UO.
La structure sera créée directement sous le domaine.

Notes techniques

Pour créer les unités d'organisation, vous devez lire les données à partir du fichier "UO_FORMATION.CSV".

Vous devez ajouter des commentaires pertinents dans votre code.

Vous devez utiliser des variables.

Au début de votre programme, vous devez ajouter du code pour supprimer les unités d'organisation déjà existantes avant de les recréer.

Votre code doit utiliser un "Try and Catch" pour ne pas afficher les messages d'erreurs de PowerShell.
Dans la section "Catch" vous devez utiliser le nom complet de l'erreur lorsqu'un objet de l'Active Directory n'existe pas.

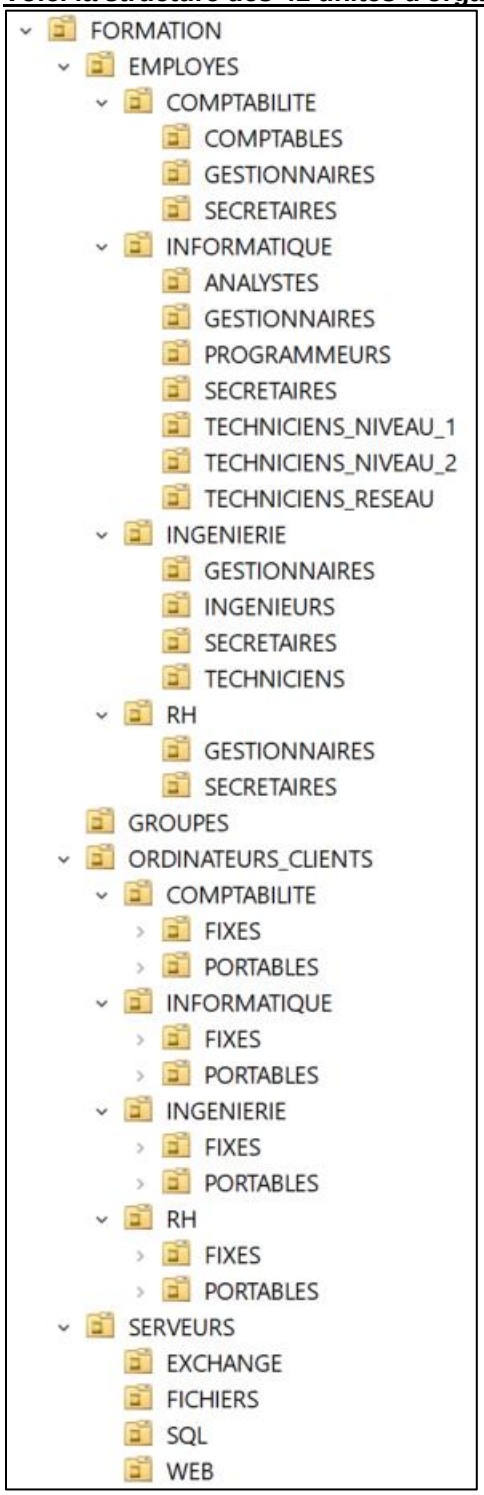
À la fin du traitement, vous devez afficher le nombre d'unités d'organisation créées.

```
OU=FORMATION,DC=formation,DC=local n'existe pas.
Création de 41 unités d'organisation.
```

```
OU=FORMATION,DC=formation,DC=local existe, donc on l'efface.
Création de 41 unités d'organisation.
```

Le module ActiveDirectory de PowerShell contient quatre cmdlets pour gérer les UO.
Get-ADOrganizationalUnit
New-ADOrganizationalUnit
Remove-ADOrganizationalUnit
Set-ADOrganizationalUnit

Voici la structure des 41 unités d'organisation




```
# ==============================================================================
# Richard Jean
# 12 septembre 2021
# NOTE 1: Le fichier CSV doit exister
# NOTE 2: Le script affiche l'adresse MAC en utilisant des "-"
# ==============================================================================
Clear-Host
```

```
# Fichier CSV
$fCsv = "ORDI_MAC.csv"

# Importation du fichier CSV
$oCsv = Import-Csv -path $fCsv -Delimiter ";"

Foreach ($Ligne in $oCsv)
{
 $NomOrdinateur = $Ligne.NomOrdi
 $AdresseMAC    = $Ligne.AdresseMAC

 $AdresseMAC = $AdresseMAC.Replace(":","-")

 Write-Host "$AdresseMAC de l'ordinateur $NomOrdinateur" -ForegroundColor Yellow
}



#==============================================================================
# Technique: Fichier CSV avec PowerShell
#
# Création de la structure des unités d'organisation
# Cette version utilise "Try and Catch"
#
# AUTEUR: Richard Jean
# DATE:   14 septembre 2022
#
# COMMENTAIRES:
#   Le fichier csv contient
#   - le nom de la UO
#   - le DistinguishedName de son parent
#==============================================================================
Clear-Host

#------------------------------------------------------------------------------
$OU = "OU=FORMATION,DC=formation,DC=local"

try
{
  Remove-ADOrganizationalUnit -Identity $OU `
                              -Recursive `
                              -Confirm:$false
}
catch [Microsoft.ActiveDirectory.Management.ADIdentityNotFoundException]
{
  $message = "$OU n'existe pas."
  Write-Host $message -ForegroundColor Yellow
}
catch
{
  Write-Warning "ERREUR INATTENDUE !!!"
  Exit
}

#------------------------------------------------------------------------------
# Création de la structure des UO en utilisant le fichier CSV
```

```
$FichierCSV = Import-Csv -Path "UO_FORMATION.csv" `
                    -Delimiter ";"

$Compte = 0

Foreach ($Ligne in $FichierCSV)
{
  $Nom    = $Ligne.Nom
  $Parent = $Ligne.Parent

  New-ADOrganizationalUnit -Name $Nom `
                      -Path $Parent `
                      -ProtectedFromAccidentalDeletion $false

  $Compte++
}

$message = "Création de $Compte unités d'organisation."
Write-Host $message
```

Propriété des utilisateurs dans l'Active Directory

Ce laboratoire doit être fait individuellement sur l'ordinateur virtuel 2

Objectifs

- Utiliser l'onglet "Éditeur d'attribut"
- Comprendre la différence entre le nom des attributs et les paramètres des cmdlets

Site qui affiche la liste complète des attributs de l'Active Directory avec des explications pour chaque attribut.
Active Directory Schema
https://docs.microsoft.com/en-us/windows/win32/adschema/attributes-all
https://docs.microsoft.com/fr-fr/windows/win32/adschema/attributes-all

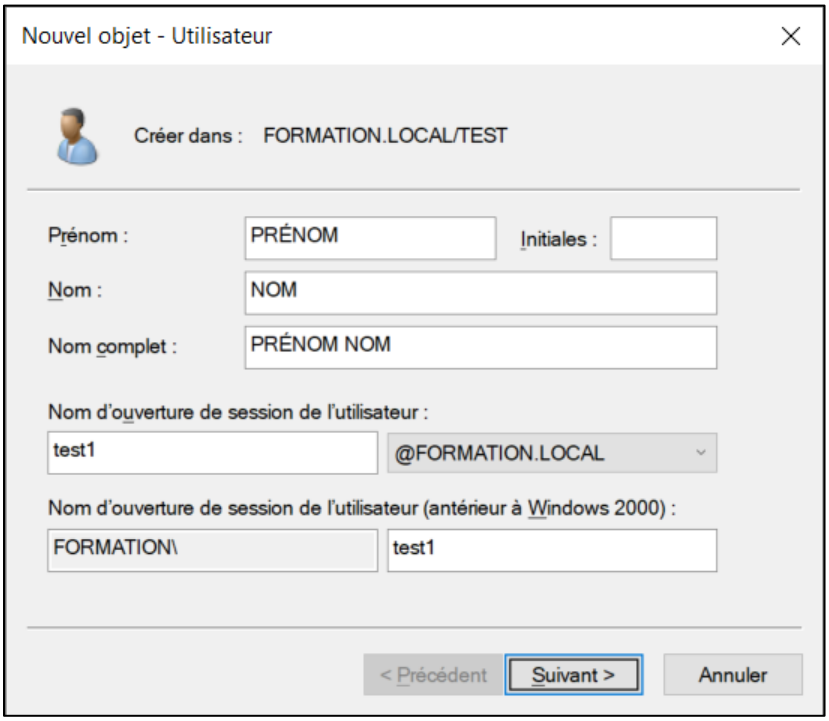
Étape 1.1 - Les propriétés de bases d'un utilisateur

Ouvrir la console UOAD et vérifier que votre affichage est en "Fonctionnalités Avancées"

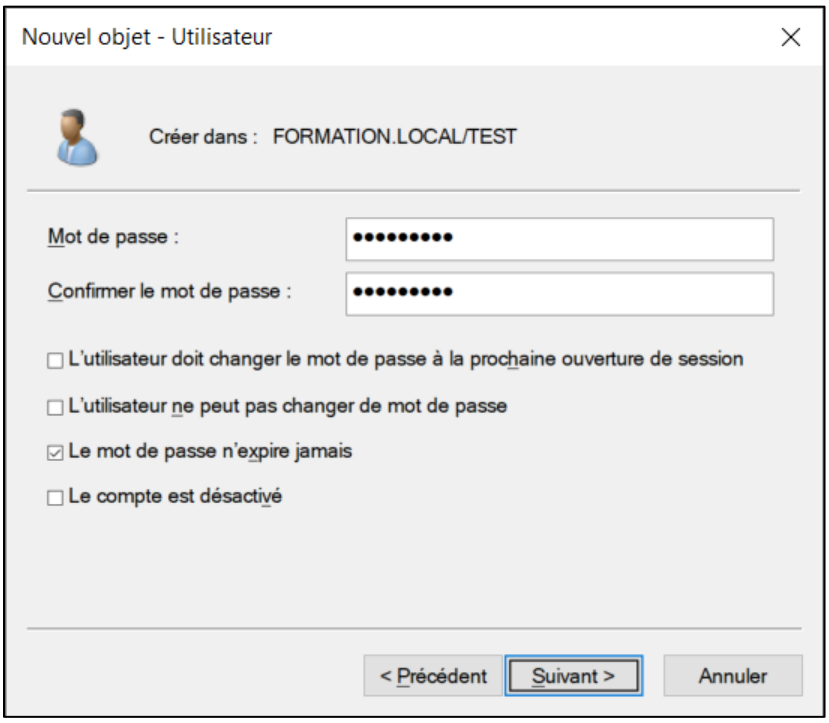
Dans l'unité d'organisation **TEST** qui est directement sous le domaine FORMATION.LOCAL

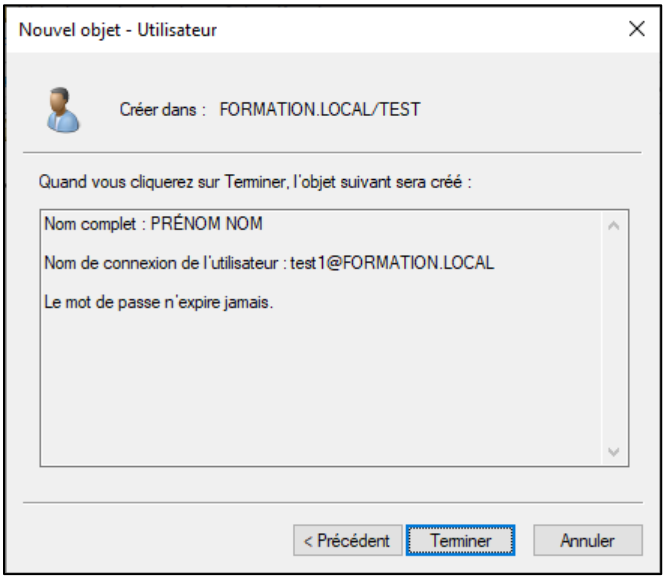
- Créer un utilisateur en utilisant les paramètres suivants :
  - Prénom: **PRÉNOM**
  - Nom: **NOM**
  - Nom complet: **PRÉNOM NOM**
  - Nom d'ouverture de session de l'utilisateur: **test1@FORMATION.LOCAL**
  - Nom d'ouverture de session de l'utilisateur (antérieur à Windows 2000): **FORMATION\test1**

  - Mot de passe: AAAaaa111
  - Le mot de passe n'expire jamais

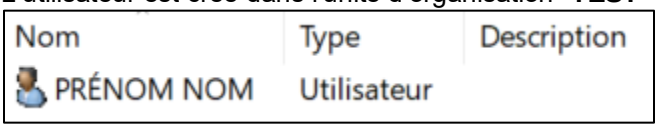


Le nom de connexion de l'utilisateur (nom d'ouverture de session) doit être unique dans tout le domaine.



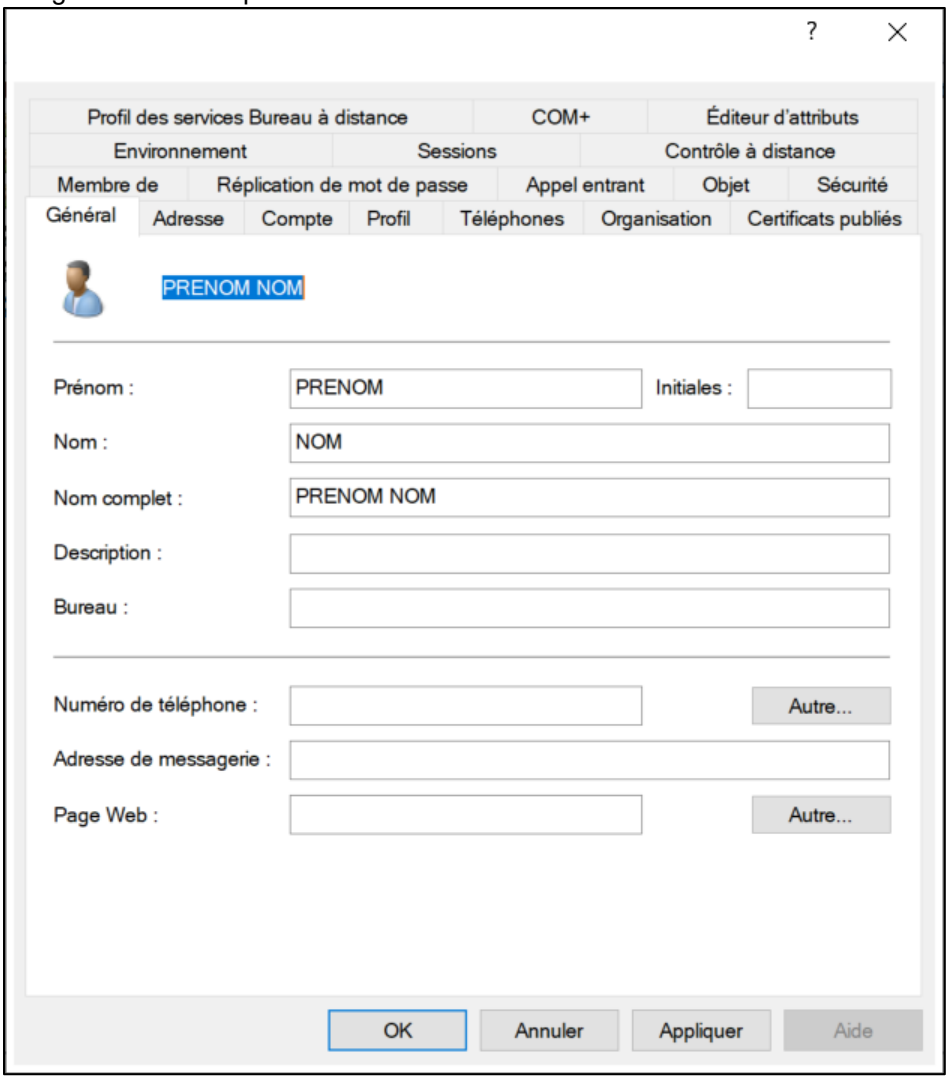


L'utilisateur est créé dans l'unité d'organisation "**TEST**"

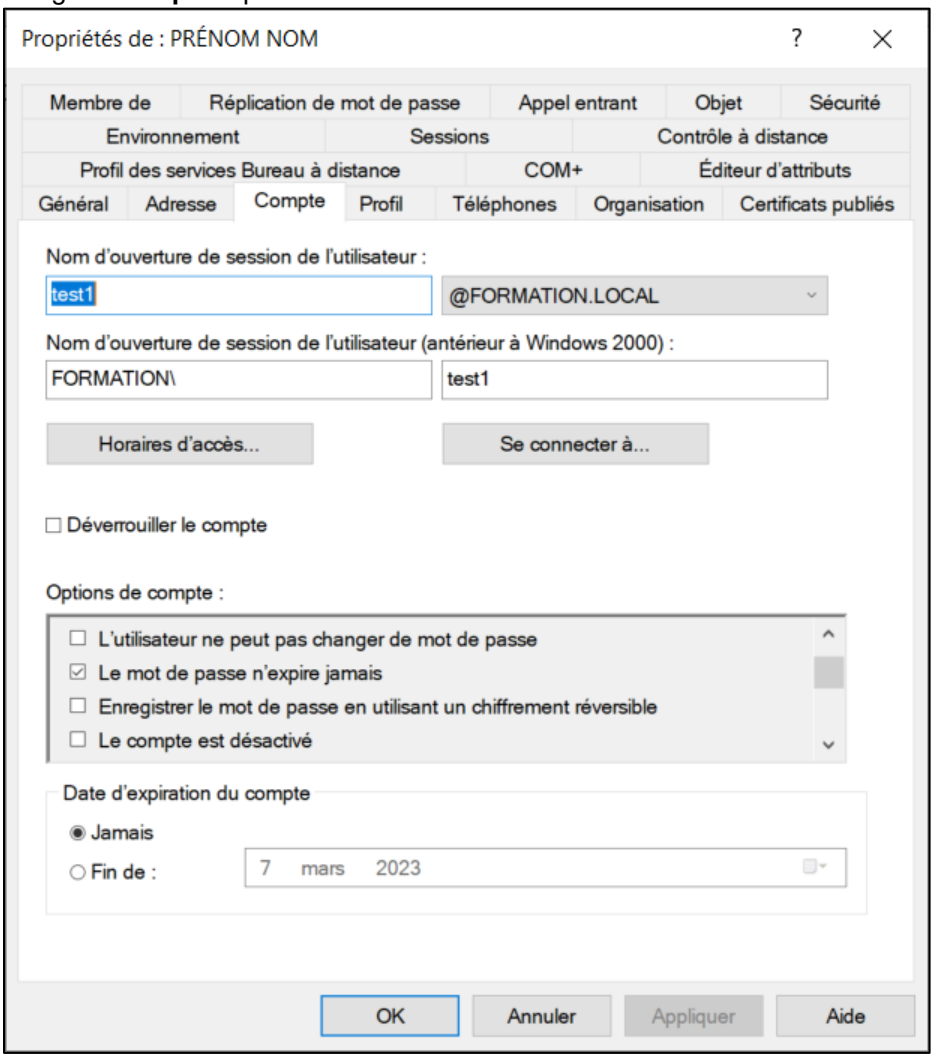


Vérification de la configuration de l'utilisateur après sa création

L'onglet "**Général**" après la création de l'utilisateur

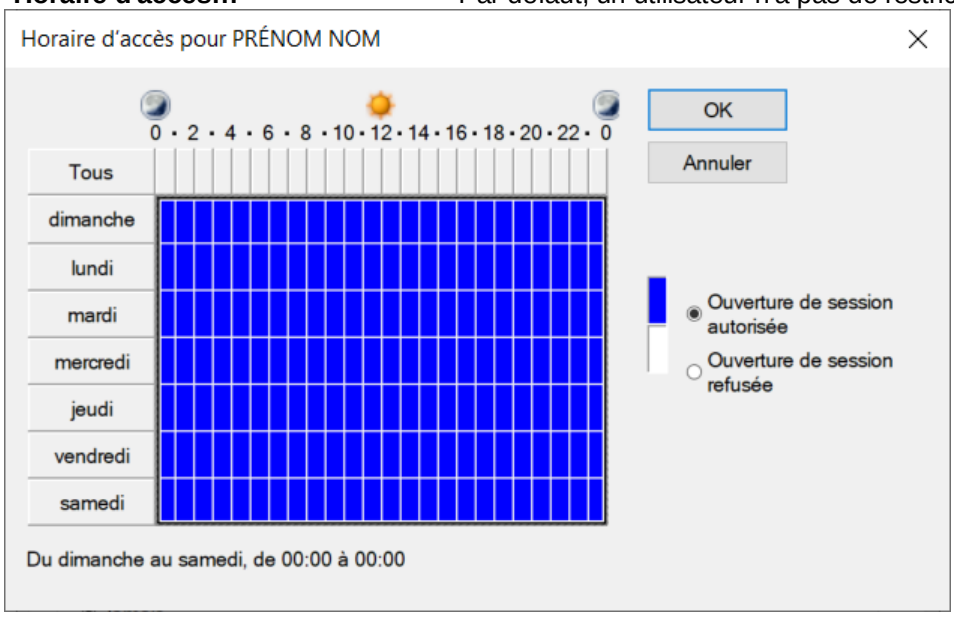


L'onglet "**Compte**" après la création de l'utilisateur

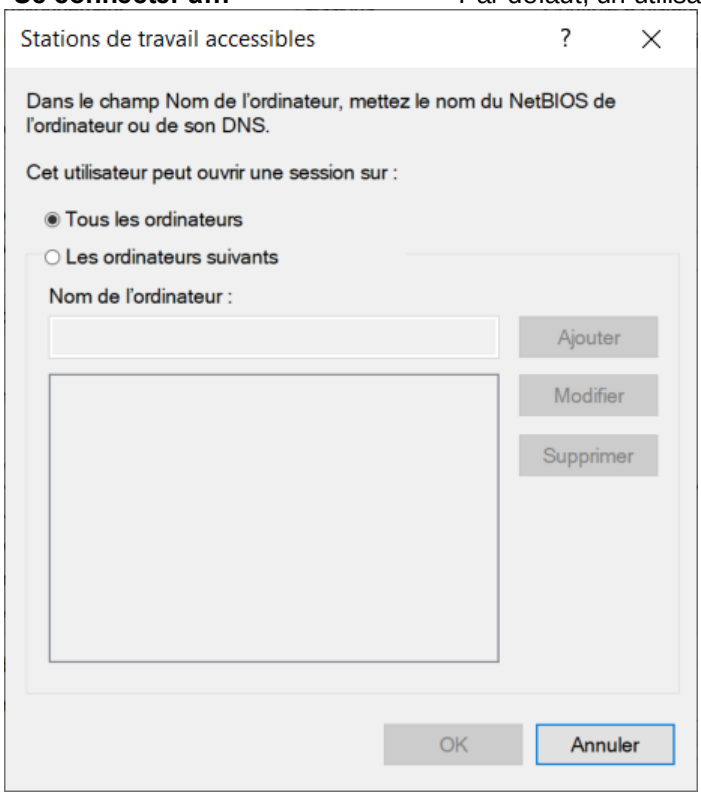


"**Date d'expiration du compte**" Par défaut, le compte d'un utilisateur n'expire jamais.

"**Horaire d'accès…**" Par défaut, un utilisateur n'a pas de restriction pour les heures d'accès.

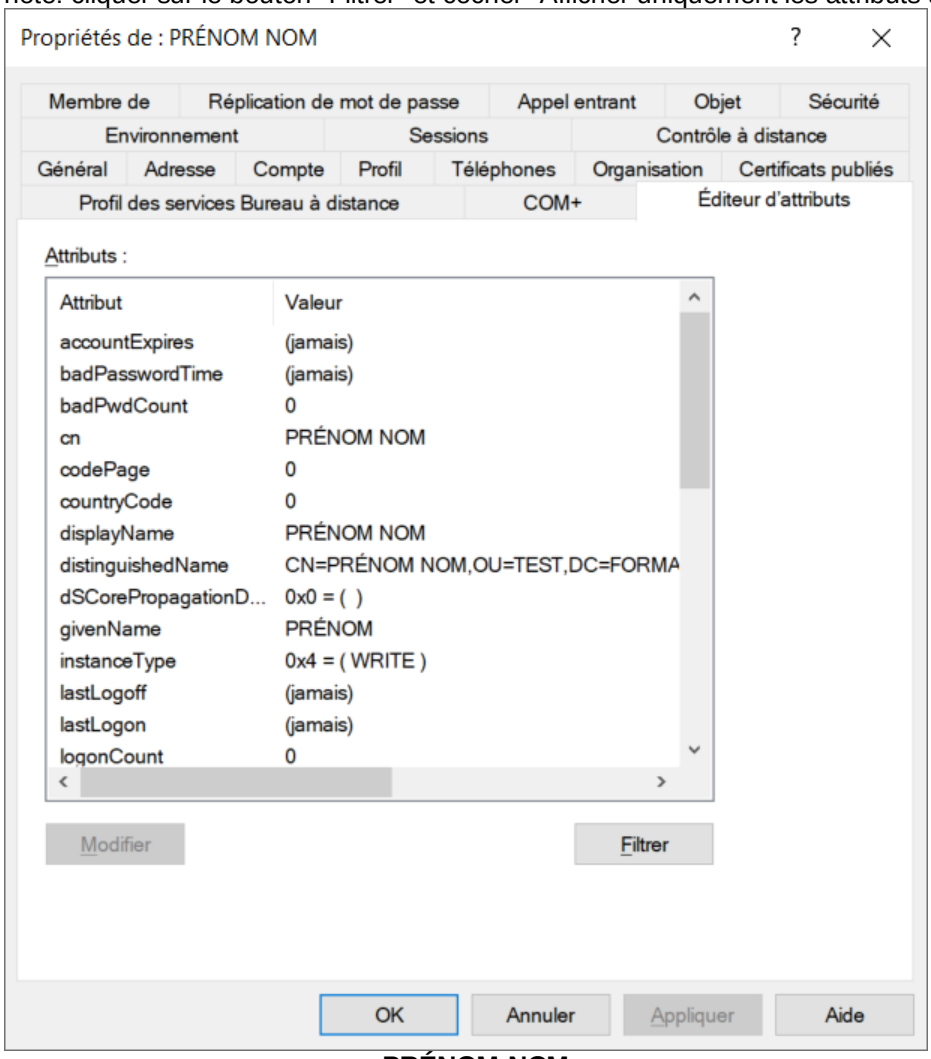


"**Se connecter à…**" Par défaut, un utilisateur peut ouvrir une session sur tous les ordinateurs.



Vérification des attributs lorsqu'un utilisateur est créé avec la console UOAD.

En utilisant l'onglet "Éditeur d'attributs", trouvez le nom des attributs qui contiennent vos valeurs.
note: cliquer sur le bouton "Filtrer" et cocher "Afficher uniquement les attributs ayant des valeurs"

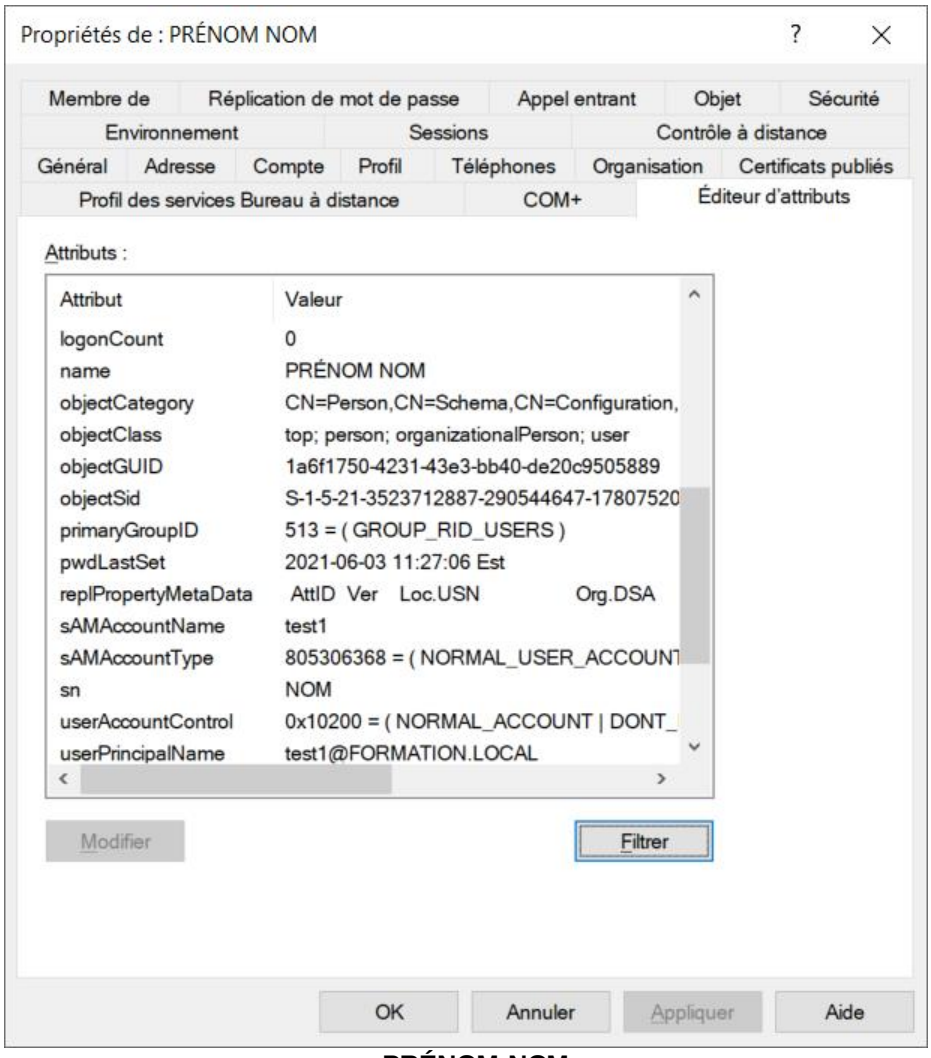


cn	**PRÉNOM NOM**
displayName	**PRÉNOM NOM**
distinguishedName	**CN=PRÉNOM NOM,OU=TEST,DC=FORMATION,DC=LOCAL**
givenName	**PRÉNOM**

Lors de la création d'un utilisateur en utilisant l'environnement graphique.
L'attribut "CN" contient le nom complet de l'utilisateur.
L'attribut "CN" est unique dans une unité d'organisation.

L'attribut "distinguishedName" est unique dans le domaine.
L'attribut "distinguishedName" débute par l'attribut "CN".

Pour être capable de créer un utilisateur qui a le même prénom et le même nom qu'un autre utilisateur qui est dans la même unité d'organisation, il faut simplement modifier le nom complet.



name **PRÉNOM NOM**
sAMAccountName **test1**
sn **NOM**
userPrincipalName **test1@FORMATION.LOCAL**

L'attribut "name" correspond au nom du compte de l'utilisateur.

Lors de la création d'un utilisateur en utilisant l'environnement graphique l'attribut "sAMAccountName" contient le nom d'ouverture de session (antérieur à Windows 2000) de l'utilisateur.

Nom du champ lors de la création de l'utilisateur	Nom de l'attribut
Prénom	**givenName**
Nom	**sn**
Nom Complet	**displayName**
Nom d'ouverture de session de l'utilisateur	**userPrincipalName**
Nom d'ouverture de session de l'utilisateur (antérieur à Windows 2000)	**sAMAccountName**

Étape 1.2 - Programmation d'un utilisateur avec PowerShell ISE

Il existe 4 cmdlet spécifiques pour la gestion des utilisateurs de l'Active Directory.

- Get-ADUser
- New-ADUser
- Remove-ADUser
- Set-ADUser

Nom du champ lors de la création de l'utilisateur	Nom de la propriété dans PowerShell pour le cmdlet New-ADUser
Prénom	**-GivenName**
Nom	**-SurName**
Nom Complet	**-DisplayName**
Nom d'ouverture de session de l'utilisateur	**-UserPrincipalName**
Nom d'ouverture de session de l'utilisateur (antérieur à Windows 2000)	**-SamAccountName**

Exemple de création d'un utilisateur avec PowerShell.

```
# Supprime l'utilisateur test1 sans aucune confirmation
Remove-ADUser -Identity test1 `
              -Confirm:$false

$mdp = ConvertTo-SecureString -AsPlainText "AAAaaa111" -Force
```

```
PS E:\> $mdp = ConvertTo-SecureString -AsPlainText "AAAaaa111" -Force

PS E:\> $mdp
System.Security.SecureString

PS E:\> $mdp | ConvertFrom-SecureString
01000000d08c9ddf0115d1118c7a00c04fc297eb0100000020b34493f2a76e478acce7dacc07087a000000000200000000001066000000010000
2000000079186dd1da767d78b0007356096ebb3f94d14c65e093db376dfcb8f215fbbc7f000000000e8000000002000020000000ba0a42e3eb04c0
4fe739f2df04d2722174cfd65d63950b07c56f3060c8264c612000000067d1e3b37876151a18967ed8de9015d894d4fae8b4e8d76da3d126a1a38
106884000000081dcedcf7b7a217cc26dca1e51a47382ce4ee2529a416ceb025ad0646c352e4dcd202621731e2ce8681f96462082f6b5d45ec752
a9a03f11dbffd9db0b64abb1

PS E:\>
```

```
# Ce code permet de recréer l'utilisateur test1 avec les mêmes propriétés qu'à l'étape 1.1.
New-ADUser -Name "PRÉNOM NOM" `
           -SamAccountName test1 `
           -UserPrincipalName "test1@formation.local" `
           -Path "OU=TEST,DC=FORMATION,DC=LOCAL" `
           -GivenName "PRÉNOM" `
           -Surname "NOM" `
           -DisplayName "PRÉNOM NOM" `
           -AccountPassword $mdp `
           -PasswordNeverExpires $true `
           -Enabled $true
```

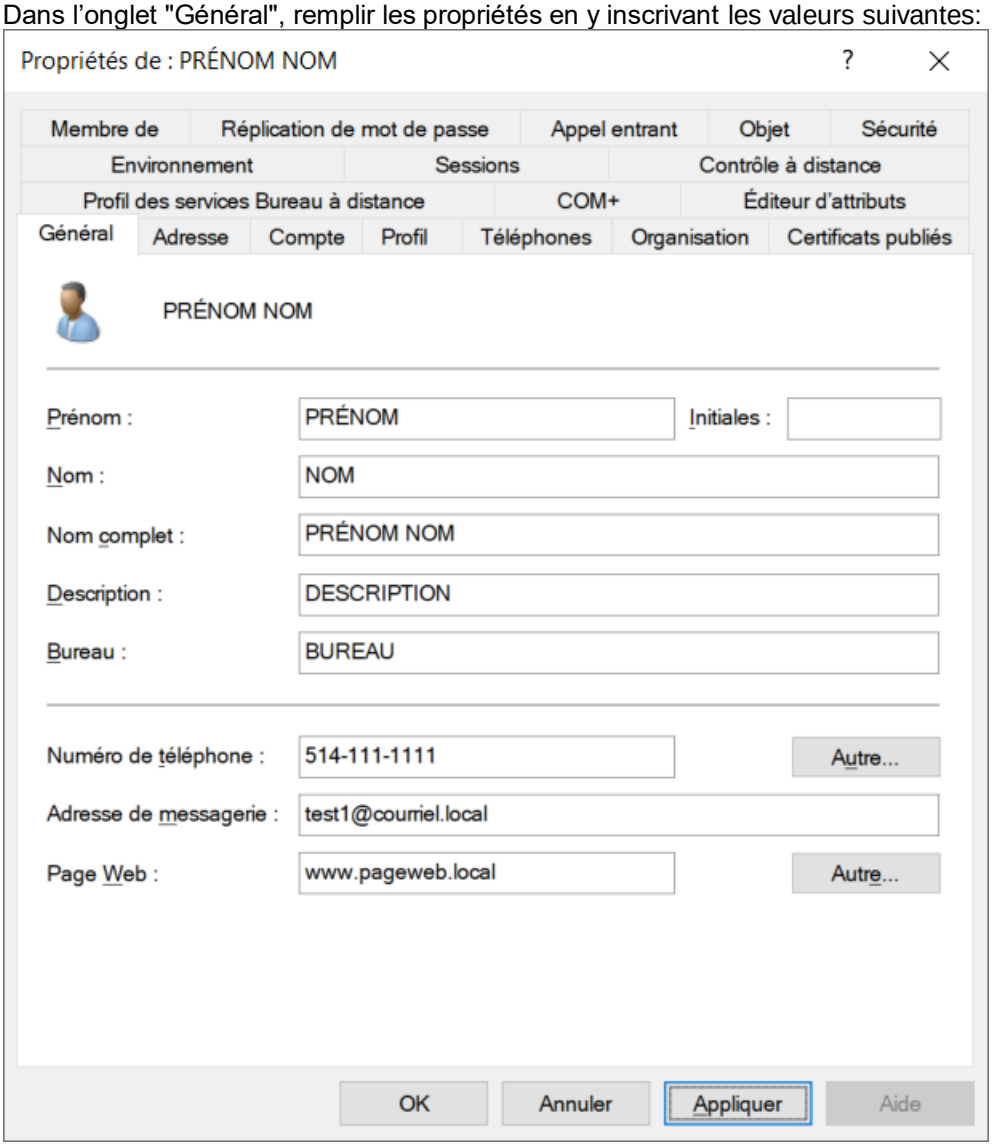
L'utilisateur est créé dans l'unité d'organisation "**TEST**"

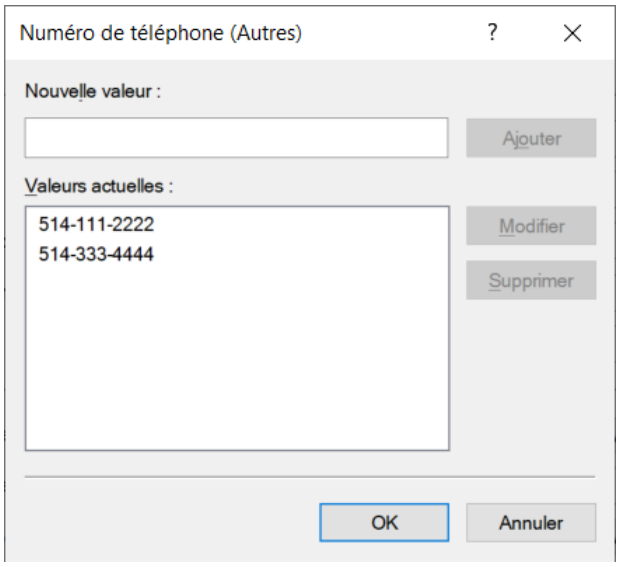
Nom	Type	Description
PRÉNOM NOM	Utilisateur	

Le résultat final est identique.

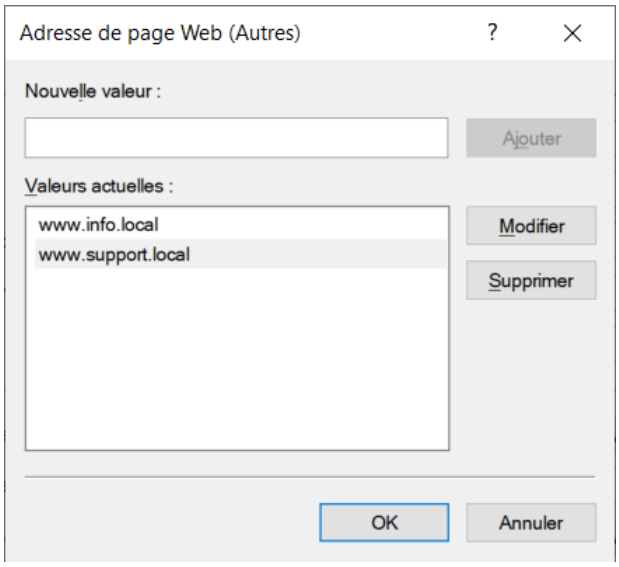
Étape 2.1 - Les attributs d'un utilisateur dans l'onglet "Général"

- Dans l'onglet "Général", remplir les propriétés en y inscrivant les valeurs suivantes:





Ajouter des numéros de téléphones



Ajouter des pages web

En utilisant l'onglet "Éditeur d'attributs", trouvez le nom des attributs qui contiennent vos valeurs.
note: cliquer sur le bouton "Filtrer" et cocher "Afficher uniquement les attributs ayant des valeurs"

Nom du champ dans l'onglet "Général"	Nom de l'attribut
Prénom	**givenName**
Nom	**sn**
Nom complet	**displayName**
Description	**description**
Bureau	**physicalDeliveryOfficeName**
Numéro de telephone Autre…	**telephoneNumber** **otherTelephone**
Adresse de messagerie	**mail**
Page Web Autre…	**wWWHomePage** **url**

Étape 2.2 - Programmation d'un utilisateur avec PowerShell ISE

Il existe 4 cmdlet spécifiques pour la gestion des utilisateurs de l'Active Directory.

- Get-ADUser
- New-ADUser
- Remove-ADUser
- Set-ADUser

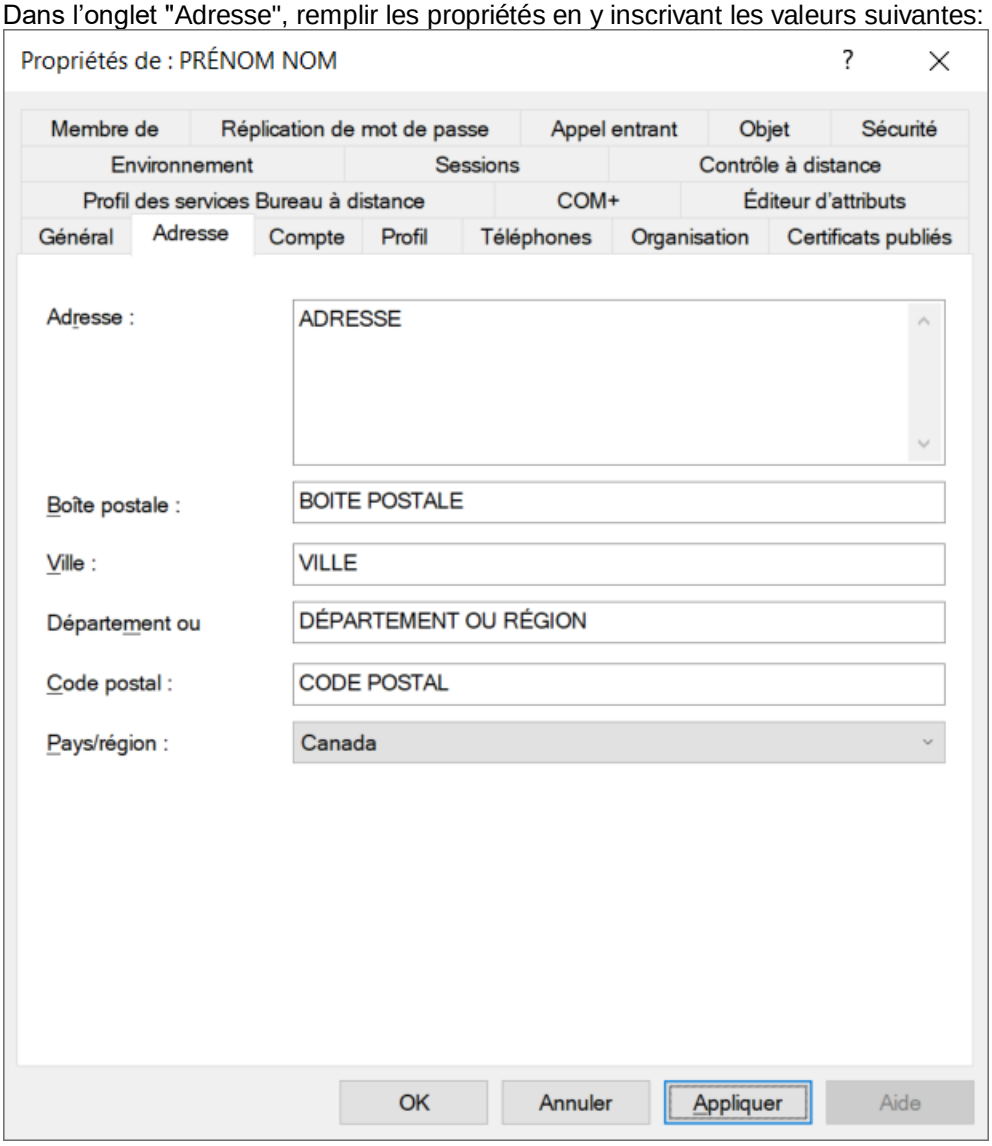
Nom du champ dans l'onglet "Général"	Nom de la propriété dans PowerShell pour le cmdlet New-ADUser
Prénom	**-GivenName**
Nom	**-SurName**
Nom complet	**-DisplayName**
Description	**-Description**
Bureau	**-Office**
Numéro de telephone Autre…	**-OfficePhone** **Il faut utiliser le paramètre -OtherAttributes avec l'attribut otherTelephone.**
Adresse de messagerie	**-EmailAddress**
Page Web Autre…	**-HomePage** **Il faut utiliser le paramètre -OtherAttributes avec l'attribut url.**

Exemple de création d'un utilisateur avec PowerShell.

```
# Permet de recréer l'utilisateur test1
# avec les mêmes propriétés qu'à l'étape 1.1 (rouge sur fond jaune)
# et avec les propriétés de l'étape 2.1 (rouge sur fond noir)
Remove-ADUser -Identity test1 `
              -Confirm:$false

$mdp = ConvertTo-SecureString -AsPlainText "AAAaaa111" -Force

New-ADUser -Name "PRÉNOM NOM" `
           -SamAccountName test1 `
           -UserPrincipalName "test1@formation.local" `
           -Path "OU=TEST,DC=FORMATION,DC=LOCAL" `
           -GivenName "PRÉNOM" `
           -Surname "NOM" `
           -DisplayName "PRÉNOM NOM" `
           -Description "DESCRIPTION" `
           -Office "BUREAU" `
           -OfficePhone "514-111-1111" `
           -EmailAddress "test1@courriel.local" `
           -HomePage "www.pageweb.local" `
           -OtherAttributes @{'otherTelephone'="514-111-2222","514-333-4444";
                              'url'="www.info.local","www.support.local"} `
           -AccountPassword $mdp `
           -PasswordNeverExpires $true `
           -Enabled $true
```

Étape 3.1 - Les attributs d'un utilisateur dans l'onglet "Adresse"

- Dans l'onglet "Adresse", remplir les propriétés en y inscrivant les valeurs suivantes:



En utilisant l'onglet "Éditeur d'attributs", trouvez le nom des attributs qui contiennent vos valeurs.
note: cliquer sur le bouton "Filtrer" et cocher "Afficher uniquement les attributs ayant des valeurs"

Nom du champ dans l'onglet "Adresse"	Nom de l'attribut
Adresse	**streetAddress**
Boîte postale	**postOfficeBox**
Ville	**l**
Département ou région	**st**
Code postal	**postalCode**
Pays/région	**note: il y a trois attributs par pays** **c=CA** **co=Canada** **countryCode=124**

Étape 3.2 - Programmation d'un utilisateur avec PowerShell ISE

Il existe 4 cmdlet spécifiques pour la gestion des utilisateurs de l'Active Directory.

- Get-ADUser
- New-ADUser
- Remove-ADUser
- Set-ADUser

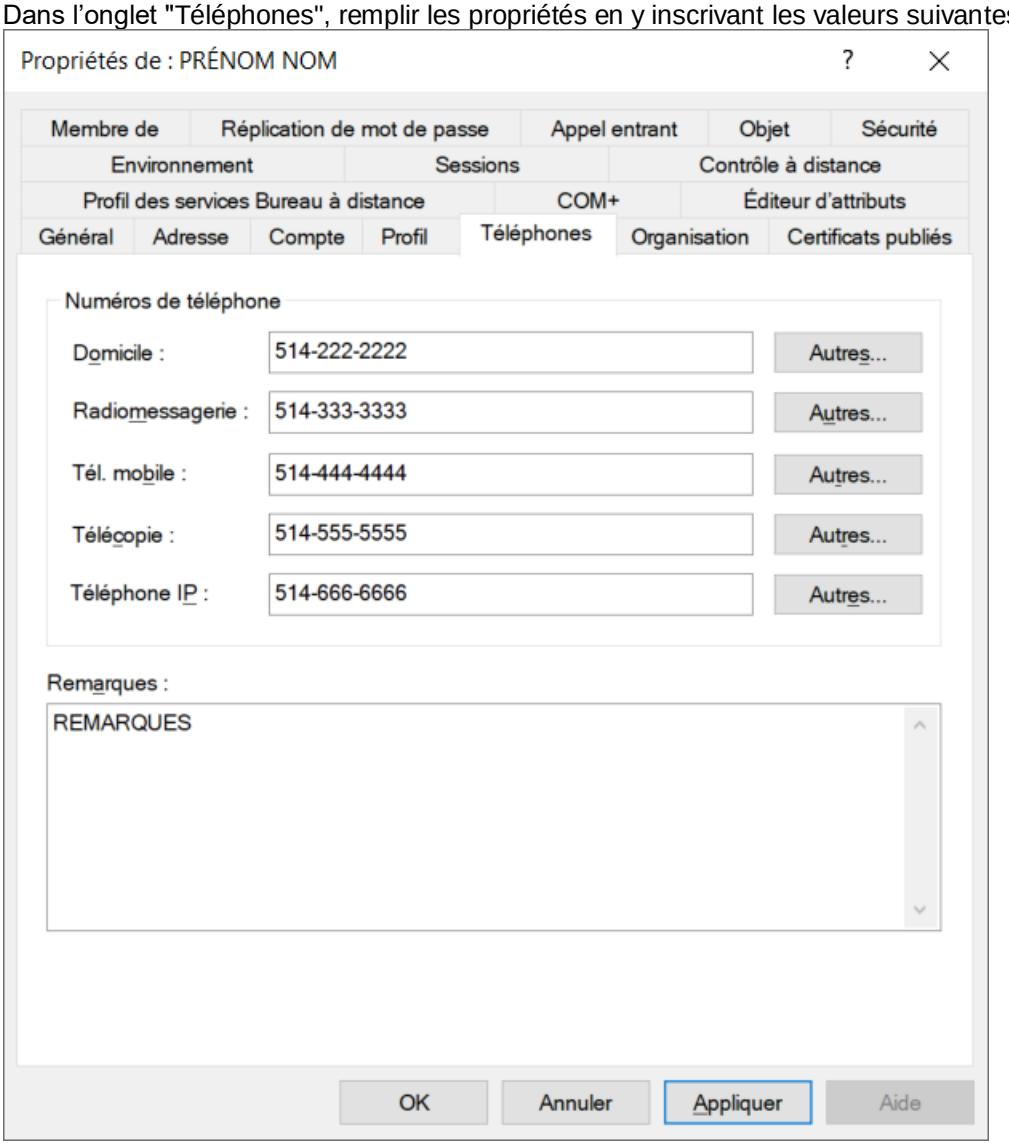
Nom du champ dans l'onglet "Adresse"	Nom de la propriété dans PowerShell pour le cmdlet New-ADUser
Adresse	**-StreetAddress**
Boîte postale	**-POBox**
Ville	**-City**
Département ou région	**-State**
Code postal	**-PostalCode**
Pays/région	**-Country** **Il est préférable d'utiliser le paramètre -OtherAttributes avec les trois attributs c, co et countryCode.**

Exemple pour configurer le pays avec le paramètre -OtherAttributes

```
-OtherAttributes @{'c'="CA";'co'="Canada";'countryCode'=124}
```

Étape 4.1 - Les attributs d'un utilisateur dans l'onglet "Téléphones"

- Dans l'onglet "Téléphones", remplir les propriétés en y inscrivant les valeurs suivantes:



En utilisant l'onglet "Éditeur d'attributs", trouvez le nom des attributs qui contiennent vos valeurs.
note: cliquer sur le bouton "Filtrer" et cocher "Afficher uniquement les attributs ayant des valeurs"

Nom du champ dans l'onglet "Téléphones"	Nom de l'attribut
Domicile Autres…	**homePhone** **otherHomePhone**
Radiomessagerie Autres…	**pager** **otherPager**
Tél. mobile Autres…	**mobile** **otherMobile**
Télécopie Autres…	**facsimileTelephoneNumber** **otherFacsimileTelephoneNumber**
Téléphone IP Autres…	**ipPhone** **otherIpPhone**
Remarques	**info**

Étape 4.2 - Programmation d'un utilisateur avec PowerShell ISE

Il existe 4 cmdlet spécifiques pour la gestion des utilisateurs de l'Active Directory.

- Get-ADUser
- New-ADUser
- Remove-ADUser
- Set-ADUser

Nom du champ dans l'onglet "Téléphones"	Nom de la propriété dans PowerShell pour le cmdlet New-ADUser
Domicile Autres…	**-HomePhone** **Il faut utiliser le paramètre -OtherAttributes avec l'attribut otherHomePhone.**
Radiomessagerie Autres…	**Il faut utiliser le paramètre -OtherAttributes avec l'attribut Pager.** **Il faut utiliser le paramètre -OtherAttributes avec l'attribut otherPager.**
Tél. mobile Autres…	**-MobilePhone** **Il faut utiliser le paramètre -OtherAttributes avec l'attribut otherMobile.**
Télécopie Autres…	**-Fax** **Il faut utiliser le paramètre -OtherAttributes avec l'attribut otherFacsimileTelephoneNumber.**
Téléphone IP Autres…	**ipPhone** **Il faut utiliser le paramètre -OtherAttributes avec l'attribut otherIpPhone.**
Remarques	**Il faut utiliser le paramètre -OtherAttributes avec l'attribut info.**

Programmation d’un utilisateur dans l'Active Directory

Ce laboratoire doit être fait individuellement sur l'ordinateur virtuel 2

Objectifs

- Utiliser PowerShell dans le cadre de la création d’objets d’un domaine
- Créer un objet utilisateur dans votre domaine

Étape 1 - Mise en place

L’arborescence des unités d’organisation du domaine Formation.Local doit être déjà créée.

Étape 2 - Introduction au cmdlet New-ADuser

Le paramètre -Name est le seul paramètre obligatoire de New-ADUser.

Voici la commande la plus simple pour créer un utilisateur: **New-ADUser -Name TOTO**

- Le compte est créé mais il est désactivé.
- L'utilisateur doit changer le mot de passe à l'ouverture de session
- L'utilisateur TOTO est créé dans le conteneur "Users".
- La valeur de l'attribut SamAccountName est le même que celui du paramètre -Name

ATTENTION: Si on ne déclare pas le paramètre -SamAccountName le contenu de l'attribut SamAccountName sera le même que celui du paramètre -Name.

Étape 3 - Création d'un utilisateur avec le cmdlet New-ADuser

Écrire un programme en PowerShell pour créer un utilisateur dans l'unité d'organisation TEST.
Votre code doit utiliser les cmdlets du module ActiveDirectory et des variables autant que possible.

Le mot de passe de l'utilisateur sera AAAaaa111

```
$mdp = ConvertTo-SecureString -AsPlainText "AAAaaa111" -Force
```

Voici les propriétés de l'onglet "**Général**" du nouvel utilisateur

- Prénom: JOHN
- Nom: DOE
- Nom complet: JOHN DOE
- Description: Mon premier utilisateur
- Bureau: Informatique
- Numéros de téléphone: 514-999-6000
  - 514-999-7000, 514-999-8000
- Adresse de messagerie: JOHN.DOE@FORMATION.LOCAL
- Page WEB: www.formation.local

Voici les propriétés de l'onglet "**Adresse**" du nouvel utilisateur

- Pays: Canada
  note: un pays est constitué de trois attributs

Voici les propriétés de l'onglet "**Compte**" du nouvel utilisateur

- Nom d'ouverture de session de l'utilisateur: JOHN.DOE@FORMATION.LOCAL
- Nom d'ouverture de session de l'utilisateur (antérieur à Windows 2000): FORMATION\JOHN.DOE
- Le mot de passe n’expire jamais: OUI
- Le compte est désactivé: NON

Voici les propriétés de l'onglet "**Téléphones**" du nouvel utilisateur

- Téléphone à domicile: 450-222-2222
- Téléphone mobile: 450-333-3333
- Télécopie: 450-444-4444

Le nom du compte de l’utilisateur sera identique au nom complet.

Au début de votre script de création de l’utilisateur ajouter une ligne de code qui va détruire votre utilisateur.

Votre code doit utiliser un "Try and Catch" pour ne pas afficher les messages d'erreurs de PowerShell.

Étape 4 - Validation

Après la création de l'utilisateur, vous devez vérifier les valeurs de chacune des propriétés dans l'onglet "Éditeur d'attribut" de ce nouvel utilisateur.

Le nom du compte de l'utilisateur est "John Doe" mais le nom d'ouverture de session est "John.Doe".

Nom	Type	Description
John Doe	Utilisateur	Mon premier utilisateur

Voici la commande PowerShell qui affiche la liste des attributs qui correspondent aux propriétés de bases lorsqu'un utilisateur est créé avec la console UOAD.

```
Get-ADuser -Identity "John.Doe" `
           -Properties * | `
               Select-Object CN,DisplayName,DistinguishedName,GivenName,
                             Name,SamAccountName,sn,UserPrincipalName
```

```
CN                : John Doe
DisplayName       : John Doe
DistinguishedName : CN=John Doe,OU=TEST,DC=FORMATION,DC=LOCAL
GivenName         : John
Name              : John Doe
SamAccountName    : John.Doe
sn                : Doe
UserPrincipalName : John.Doe@FORMATION.LOCAL
```

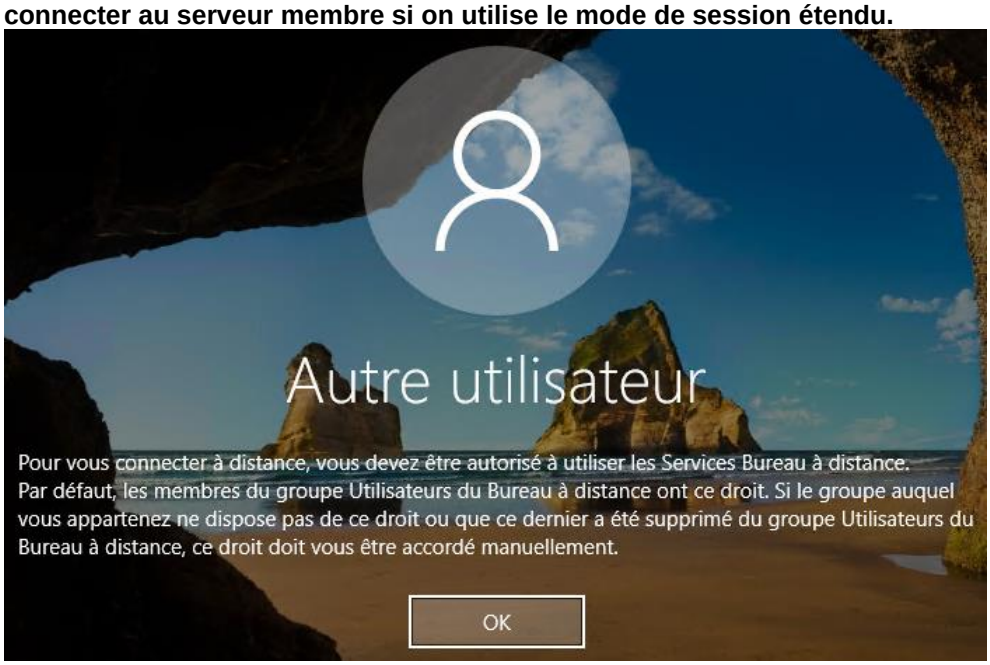
Une fois toutes les propriétés exactes, vérifier la fonctionnalité de votre utilisateur en vous connectant.
Attention : **Par défaut, les utilisateurs qui ne sont pas membre du groupe "Administrateurs" ne peuvent pas se connecter au serveur membre si on utilise le mode de session étendu.**

Pour la solution à ce problème, voir la dernière page de ce document.

Pour "changer d'utilisateur" vous pouvez exécuter TSDISCON.EXE
Il est important de fermer la session d'un utilisateur si votre intention est de modifier ses propriétés.

PROBLÈME AVEC LE MODE DE SESSION ÉTENDU

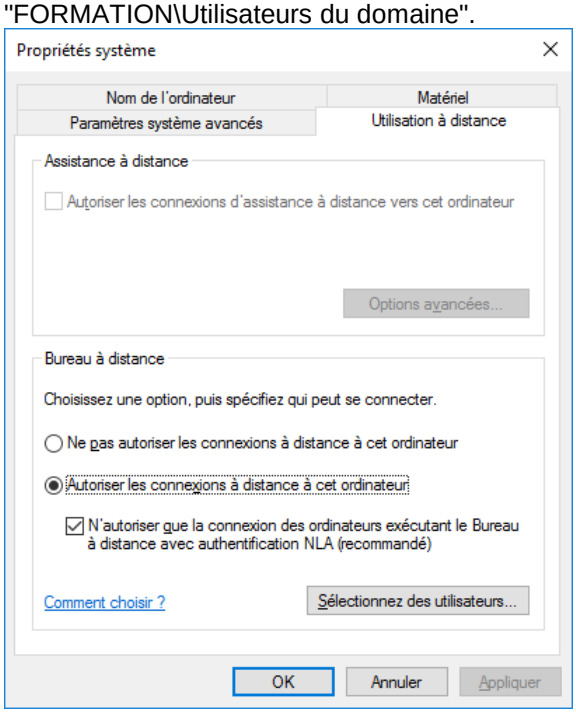
Par défaut, les utilisateurs qui ne sont pas membre du groupe "Administrateurs" ne peuvent pas se connecter au serveur membre si on utilise le mode de session étendu.

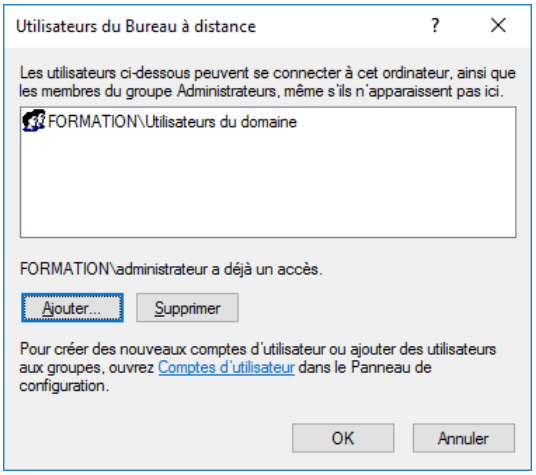


SOLUTION SIMPLE À CE PROBLÈME

Modification à effectuer sur le serveur membre

Dans "Propriétés système" cliquer sur le bouton "Sélectionnez des utilisateurs…" et ajouter le groupe "FORMATION\Utilisateurs du domaine".





#==========
AUTEUR: Richard Jean
DATE : 5 février 2023

```
#
# Création d'un utilisateur à  l'aide des cmdlets du module ActiveDirectory
# L'utilisateur sera créé directement sous l'unité d'organisation TEST
#==============================================================================
Clear-Host

# On efface le compte utilisateur avant de le créer
try
{
  Remove-ADUser -Identity "John.Doe" -Confirm:$False
}
catch [Microsoft.ActiveDirectory.Management.ADIdentityNotFoundException]
{
  Write-Host "Le compte 'John Doe' n'existe pas." -ForegroundColor Yellow
}
catch
{
  Write-Warning "ERREUR INATTENDUE !!!"
  Exit
}

$mdp = ConvertTo-SecureString -AsPlainText "AAAaaa111" -Force

New-ADUser -Name "John Doe" `
        -SamAccountName "John.Doe" `
        -UserPrincipalName "John.Doe@FORMATION.LOCAL" `
        -Path "OU=test,DC=formation,DC=local" `
        -GivenName "John" `
        -Surname "Doe" `
        -DisplayName "John Doe"  `
        -Description "Mon premier utilisateur" `
        -Office "INFORMATIQUE" `
        -OfficePhone "514-999-6000" `
        -HomePhone "450-222-2222" `
        -MobilePhone "450-333-3333" `
        -Fax "450-444-4444" `
        -EmailAddress "John.Doe@FORMATION.LOCAL" `
        -HomePage "WWW.FORMATION.LOCAL" `
        -OtherAttributes @{'c'="CA";'co'="CANADA";'countryCode'=124;
                'otherTelephone'="514-999-7000","514-999-8000"} `
        -AccountPassword $mdp `
        -PasswordNeverExpires $true `
        -Enabled $true

Write-Host "Fin de la création du compte utilisateur 'John Doe'"
```

Création de plusieurs utilisateurs

Ce laboratoire doit être fait individuellement sur l'ordinateur virtuel 2

Objectifs

- Création de plusieurs utilisateurs en utilisant un fichier CSV

Informations générales

Vous devez écrire un script PowerShell utilisant les cmdlets du module ActiveDirectory permettant de créer plusieurs utilisateurs à l'intérieur de la structure des unités d'organisation de FORMATION.

Les données nécessaires à la création de chaque utilisateur sont incluses dans le fichier "FORMATION_donnees.CSV" disponible sur LÉA.

Description du fichier CSV
Le fichier CSV est constitué de cinq colonnes

- Nom;Prenom;Code;Login;Parent

- La colonne 1 contient le nom de famille de l'utilisateur.
- La colonne 2 contient le prénom de l'utilisateur.
- La colonne 3 contient un code qui sera ajouté au nom complet de l'utilisateur.
- La colonne 4 contient le nom d'ouverture de session de l'utilisateur.
- La colonne 5 contient le distinguishedName du parent de l'utilisateur.

Informations sur la création des utilisateurs

Le nom du compte de chaque utilisateur sera identique au nom d'ouverture de session de l'utilisateur.

Le nom complet de chaque utilisateur sera "**prénom nom – code"**

Le compte de chaque utilisateur sera activé.

Le mot de passe **AAAaaa111** sera le même pour tous vos utilisateurs.
Le mot de passe n'expire jamais.

Votre script doit compter les utilisateurs créés et afficher le nombre total.

À chaque création d'utilisateur, vous devez afficher le nom d'ouverture de session de l'utilisateur.

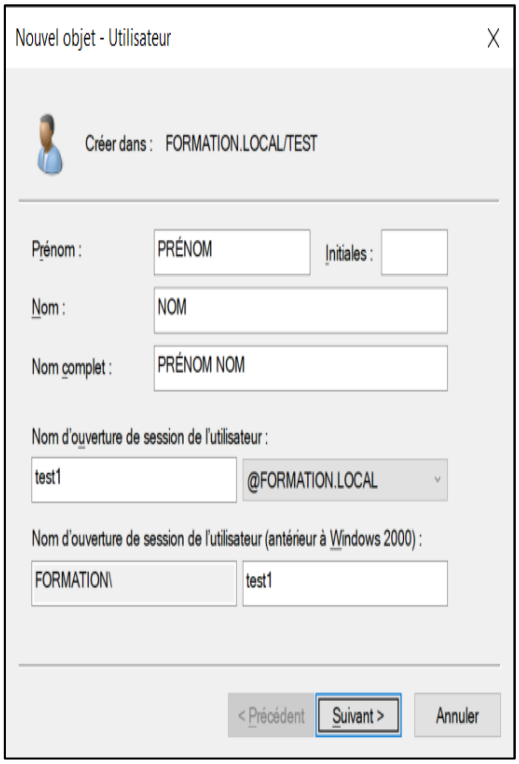
Au sujet de la destruction de vos usagers au début du script :il n'est pas nécessaire d'ajouter un try-catch avec les lignes suivantes :

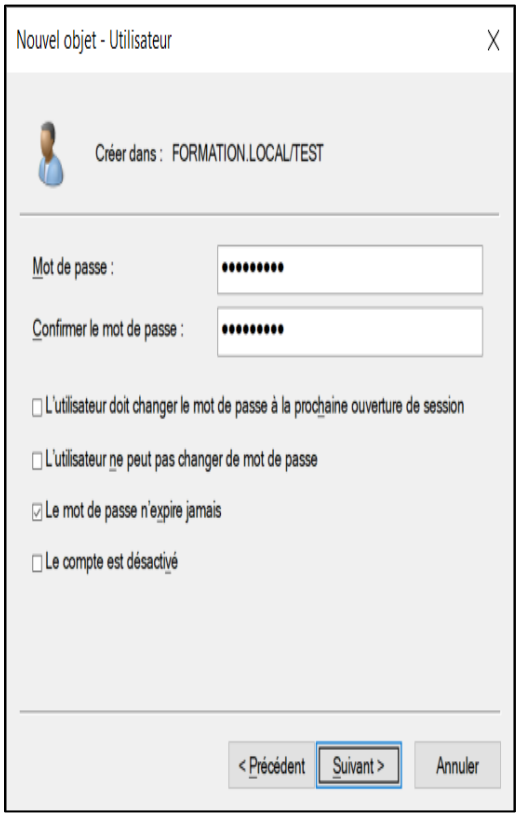
```
$chemin = "OU=FORMATION,DC=FORMATION,DC=LOCAL"
Get-ADUser -Filter 'SamAccountName -like "EMP*"' `
           -SearchBase $chemin | Remove-ADUser -Confirm:$false
```

Raison : le parameter -filter ne génère pas d'erreur PowerShell si l'objet n'existe pas (contrairement au paramètre -identity)

ACTIVE DIRECTORY :résumé des principales propriétés d'un utilisateur

Pour un usager créé dans l'unité d'organisation TEST, qui est directement sous le domaine FORMATION.LOCAL.

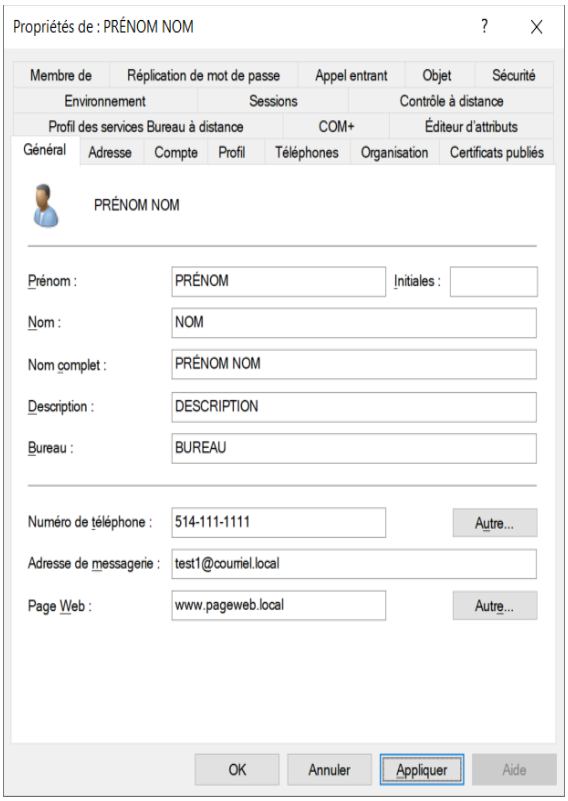


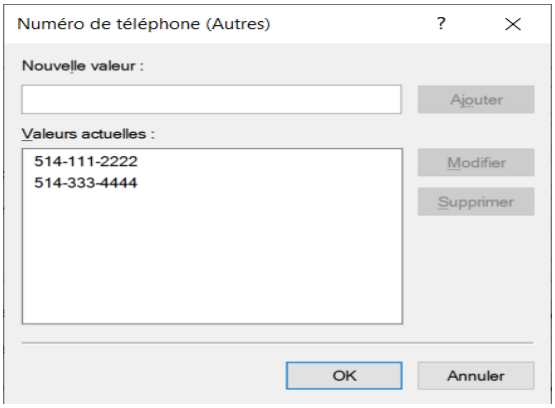


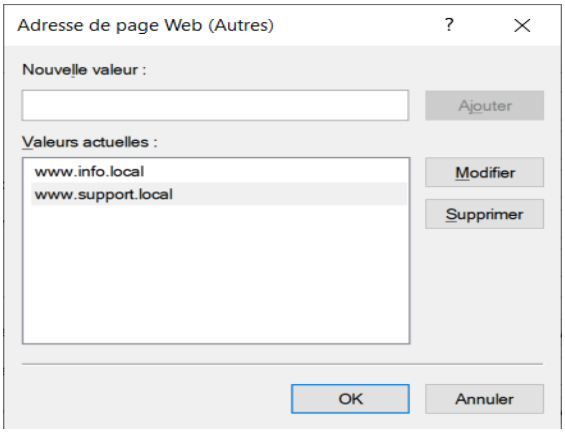
Champs lors de la création de l'utilisateur	Attributs dans Active Directory	Propriétés dans PowerShell
Prénom (ex : PRÉNOM)	GivenName	-GivenName
Nom (ex : NOM) (c'est le nom de famille))	sn	-SurName
Nom complet (ex : PRÉNOM NOM)	DisplayName	-DisplayName
Nom d'ouverture de session de l'utilisateur (ex : test1@FORMATION.LOCAL)	UserPrincipalName	-UserPrincipalName
Nom d'ouverture de session de l'utilisateur (antérieur à Windows 2000) (ex : FORMATION\test1)	sAMAccountName seulement test1	-sAMAccountName on doit écrire seulement test1 : (ex : -sAMAccountName test1)
	DistinguishedName (ex :CN=PRÉNOM NOM,OU=TEST,DC=FORMATION, DC=LOCAL)	?
	Cn (ex :PRÉNOM NOM)	?
	Name (ex :PRÉNOM NOM) (Surtout pour PowerShell)	-Name Paramètre particulier en PSH Seul obligatoire Correspond au Nom du compte
	…il y en a plusieurs autres… Voir l'onglet Éditeurs d'attributs (Lab 6a, pages 6 et 7 pour un exemple)	

Dans l'interface graphique, une fois l'utilisateur créé, on peut aller dans ses propriétés pour continuer la configuration :

Onglet Général :

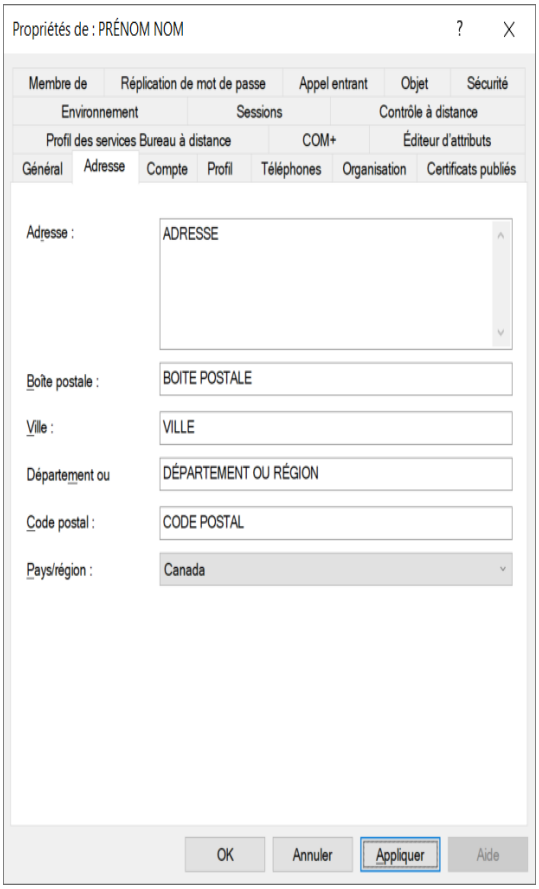






Champs lors de la création de l'utilisateur	Attributs dans Active Directory	Propriétés dans PowerShell
Description (ex : DESCRIPTION)	Description	- Description
Bureau (ex : BUREAU)	PhysicalDeliveryOfficeName	-Office
Numéro de téléphone (ex : 514-111-1111) (tél au bureau) Numéro(s) de tél suppl : Bouton Autre…	TelephoneNumber OtherTelephone	-OfficePhone (si un seul numéro de téléphone) -OtherAttributes avec OtherTelephone (s'il y a des numéros de téléphone supplémentaires)
Adresse de messagerie (ex : test1@courriel.local) (courriel du bureau)	Mail	-EmailAddress
Page Web (ex : www.pageweb.local) (web du bureau) Page(s) Web suppl : Bouton Autre…	wwwHomePage url	-HomePage -OtherAttributes avec url

Onglet Adresse

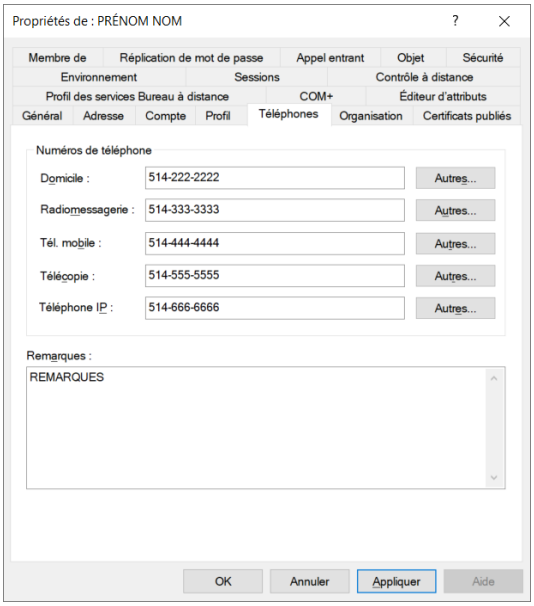


Champs lors de la création de l'utilisateur	Attributs dans Active Directory	Propriétés dans PowerShell
Adresse (ex :ADRESSE)	StreetAddress	-StreetAddress
Boîte postale (ex : BOÎTE POSTALE)	PostOfficeBox	- POBox
Ville (ex : VILLE)	L	-City
Département ou Région (ex : DÉPARTEMENT OU RÉGION)	St	-State
Code Postal (ex : CODE POSTAL)	PostalCode	-PostalCode
Pays/Région (ex : Canada)	Il y a 3 attributs par pays : C=CA Co=Canada CountryCode=124	-Country (Donne 1 des 3 seulement) (exemple = CA) -OtherAttributes avec c, co et countrycode

Liste de tous les codes ISO des pays, tels que décrits dans le standard international ISO 3166 :

https://www.iban.com/country-codes

Onglet Téléphones



Champs lors de la création de l'utilisateur	Attributs dans Active Directory	Propriétés dans PowerShell
Domicile (ex : 514-222-2222) Numéro(s) de tél du domicile suppl : Bouton Autre…	HomePhone OtherHomePhone	- HomePhone -OtherAttributes avec OtherHomeTelephone
Radiomessagerie (ex : 514-333-3333) Numéro(s) de radiomessagerie suppl : Bouton Autre…	Pager OtherPager	-Pager -OtherAttributes avec OtherPager
Tél mobile (ex : 514-444-4444) Numéro(s) de tél mobile suppl : Bouton Autre…	Mobile OtherMobile	-MobilePhone -OtherAttributes avec OtherMobile
Télécopie (ex : 514-555-5555) (courriel du bureau) Numéro(s) de télécopie suppl : Bouton Autre…	FacsimileTelephoneNumber Other FacsimileTelephoneNumber	-Fax -OtherAttributes avec OtherFacsimileTelephoneNumber
Téléphonie IP (ex : 666-6666) Numéro(s) de téléphonie IP suppl : Bouton Autre…	IpPhone OtherIPPhone	-IpPhone -OtherAttributes avec OtherIPPhone
Remarques	Info	-OtherAttributes avec info

Exemple de script PowerShell pour la création d'un usager dans l'AD avec des propriétés de base :

```
$mdp = ConvertTo-SecureString -AsPlainText "AAAaaa111" -Force

New-ADUser -Name "PRÉNOM NOM" `
           -SamAccountName test1 `
           -UserPrincipalName "test1@formation.local" `
           -Path "OU=TEST,DC=FORMATION,DC=LOCAL" `
           -GivenName "PRÉNOM" `
           -Surname "NOM" `
           -DisplayName "PRÉNOM NOM" `
           -Description "DESCRIPTION" `
           -Office "BUREAU" `
           -OfficePhone "514-111-1111" `
           -EmailAddress "test1@courriel.local" `
           -HomePage "www.pageweb.local" `
           -OtherAttributes @{'c'="CA";'co'="CANADA";'countryCode'=124;
                              'otherTelephone'="514-111-2222","514-333-4444";
                              'url'="www.info.local","www.support.local"} `
           -AccountPassword $mdp `
           -PasswordNeverExpires $true `
           -Enabled $true
```

Tableau Résumé des principales propriétés d'un usager :

Champs lors de la création de l'utilisateur	Attributs dans Active Directory	Propriétés dans PowerShell
Prénom (ex : PRÉNOM)	GivenName	-GivenName
Nom (ex : NOM) (c'est le nom de famille))	sn	-SurName
Nom complet (ex : PRÉNOM NOM)	DisplayName	-DisplayName
Nom d'ouverture de session de l'utilisateur (ex : test1@FORMATION.LOCAL)	UserPrincipalName	-UserPrincipalName
Nom d'ouverture de session de l'utilisateur (antérieur à Windows 2000) (ex : FORMATION\test1)	sAMAccountName seulement test1	-sAMAccountName on doit écrire seulement test1 : (ex : -sAMAccountName test1)
	DistinguishedName (ex :CN=PRÉNOM NOM,OU=TEST,DC=FORMATION, DC=LOCAL)	?
	Cn (ex :PRÉNOM NOM)	?
	Name (ex :PRÉNOM NOM) (Surtout pour PowerShell)	-Name Paramètre particulier en PSH Seul obligatoire Correspond au Nom du compte
	…il y en a plusieurs autres… Voir l'onglet Éditeurs d'attributs (Lab 6a, pages 6 et 7 pour un exemple)	

Champs lors de la création de l'utilisateur	Attributs dans Active Directory	Propriétés dans PowerShell
Description (ex : DESCRIPTION)	Description	- Description
Bureau (ex : BUREAU)	PhysicalDeliveryOfficeName	-Office
Numéro de téléphone (ex : 514-111-1111) (tél au bureau) Numéro(s) de tél suppl : Bouton Autre…	TelephoneNumber OtherTelephone	-OfficePhone (si un seul numéro de téléphone) -OtherAttributes avec OtherTelephone (s'il y a des numéros de téléphone supplémentaires)
Adresse de messagerie (ex : test1@courriel.local) (courriel du bureau)	Mail	-EmailAddress
Page Web (ex : www.pageweb.local) (web du bureau) Page(s) Web suppl : Bouton Autre…	wwwHomePage url	-HomePage -OtherAttributes avec url

Champs lors de la création de l'utilisateur	Attributs dans Active Directory	Propriétés dans PowerShell
Adresse (ex :ADRESSE)	StreetAddress	-StreetAddress
Boîte postale (ex : BOÎTE POSTALE)	PostOfficeBox	- POBox
Ville (ex : VILLE)	L	-City
Département ou Région (ex : DÉPARTEMENT OU RÉGION)	St	-State
Code Postal (ex : CODE POSTAL)	PostalCode	-PostalCode
Pays/Région (ex : Canada)	Il y a 3 attributs par pays : C=CA Co=Canada CountryCode=124	-Country (Donne 1 des 3 seulement) (exemple = CA) -OtherAttributes avec c, co et countrycode

Champs lors de la création de l'utilisateur	Attributs dans Active Directory	Propriétés dans PowerShell
Domicile (ex : 514-222-2222) Numéro(s) de tél du domicile suppl : Bouton Autre…	HomePhone OtherHomePhone	- HomePhone -OtherAttributes avec OtherHomeTelephone
Radiomessagerie (ex : 514-333-3333) Numéro(s) de radiomessagerie suppl : Bouton Autre…	Pager OtherPager	-Pager -OtherAttributes avec OtherPager
Tél mobile (ex : 514-444-4444) Numéro(s) de tél mobile suppl : Bouton Autre…	Mobile OtherMobile	-MobilePhone -OtherAttributes avec OtherMobile
Télécopie (ex : 514-555-5555) (courriel du bureau) Numéro(s) de télécopie suppl : Bouton Autre…	FacsimileTelephoneNumber Other FacsimileTelephoneNumber	-Fax -OtherAttributes avec OtherFacsimileTelephoneNumber
Téléphonie IP (ex : 666-6666) Numéro(s) de téléphonie IP suppl : Bouton Autre…	IpPhone OtherIPPhone	-IpPhone -OtherAttributes avec OtherIPPhone
Remarques	Info	-OtherAttributes avec info

```
$mdp = ConvertTo-SecureString -AsPlainText "AAAaaa111" -Force

New-ADUser -Name "PRÉNOM NOM" `
           -SamAccountName test1 `
           -UserPrincipalName "test1@formation.local" `
           -Path "OU=TEST,DC=FORMATION,DC=LOCAL" `
           -GivenName "PRÉNOM" `
           -Surname "NOM" `
           -DisplayName "PRÉNOM NOM" `
           -Description "DESCRIPTION" `
           -Office "BUREAU" `
           -OfficePhone "514-111-1111" `
           -EmailAddress "test1@courriel.local" `
           -HomePage "www.pageweb.local" `
           -OtherAttributes @{'c'="CA";'co'="CANADA";'countryCode'=124;
                              'otherTelephone'="514-111-2222","514-333-4444";
                              'url'="www.info.local","www.support.local"} `
           -AccountPassword $mdp `
           -PasswordNeverExpires $true `
           -Enabled $true
```

Gestion des partages sur les serveurs de fichiers

Ce laboratoire doit être fait individuellement sur l'ordinateur virtuel 2

Objectif

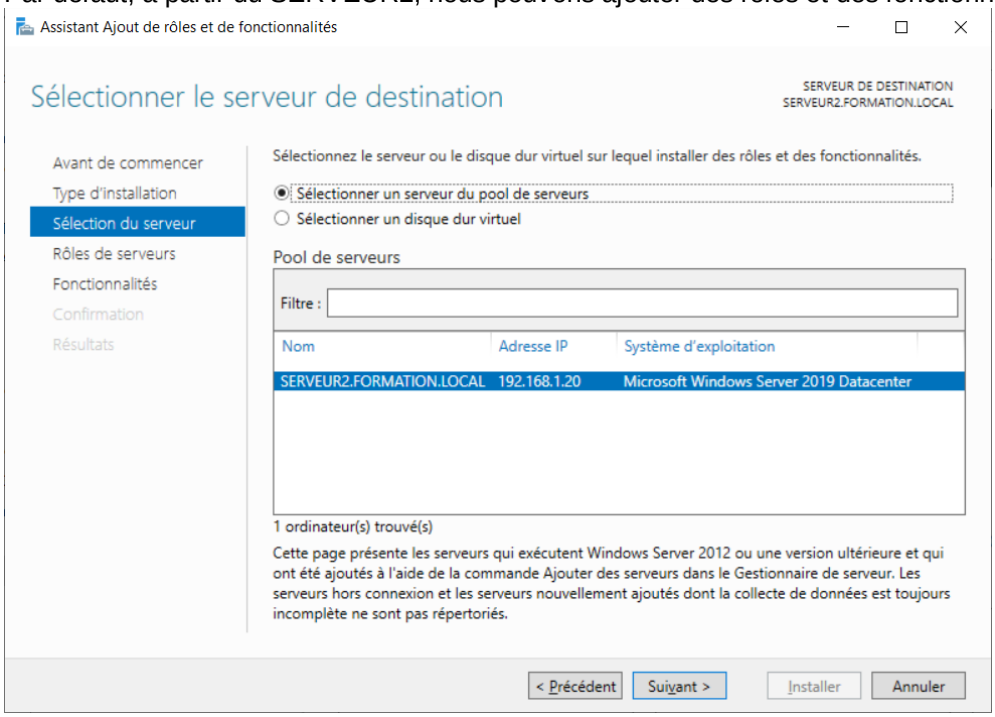
Dans le laboratoire qui suit celui-ci (laboratoire 6F), nous avons besoin de créer des partages sur le SERVEUR1 mais à partir du SERVEUR2.

Pour pouvoir créer ces partages, il faut **ajouter le rôle "Gestionnaire de ressources du serveur de fichiers" sur les deux serveurs.**

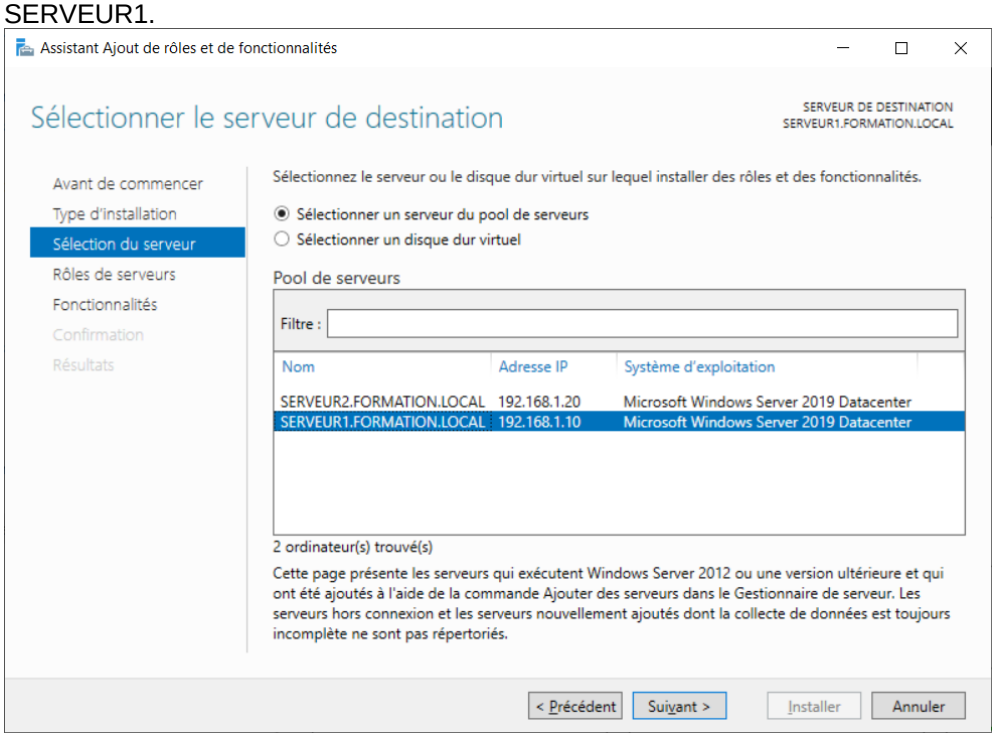
Il est possible d'installer des rôles sur le SERVEUR1 à partir du SERVEUR2 à condition d'ajouter le SERVEUR1 à l'aide de l'option "Gestionnaire de serveur / Gérer / Ajouter des serveurs".

Démarrer le gestionnaire de serveur.

Par défaut, à partir du SERVEUR2, nous pouvons ajouter des rôles et des fonctionnalités sur le SERVEUR2.



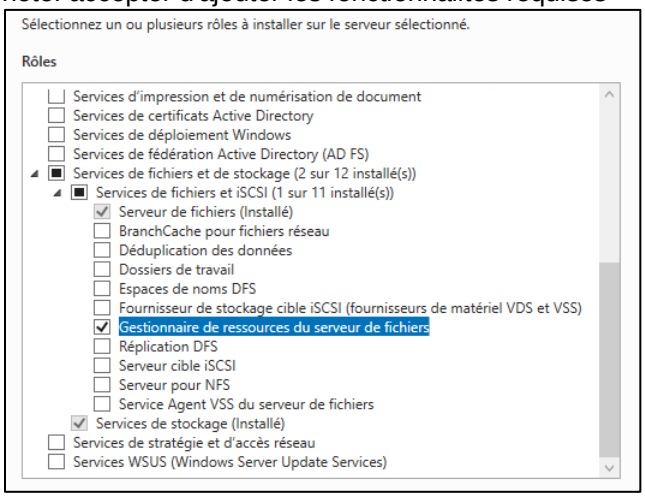
À partir du SERVEUR2, nous pouvons aussi ajouter des rôles et des fonctionnalités sur le SERVEUR2 et le SERVEUR1.



Remarquez les 4 lignes au bas de la dernière fenêtre : on vous indique quoi faire si votre Serveur1 n'apparaît pas.

Vous devez effectuer les prochaines étapes sur le SERVEUR2 et sur le SERVEUR1.
Dans la console "**Gestionnaire de serveur**"

- Sélectionner le serveur
- Ajouter le rôle "**Gestionnaire de ressources du serveur de fichiers**"
  note: accepter d'ajouter les fonctionnalités requises



Rappel : l'ajout du rôle **Gestionnaire de ressources du serveur de fichiers** et de ses fonctionnalités sur SERVEUR1 er SERVEUR2 (ce que vous avez fait dans ce laboratoire) est une étape nécessaire pour pouvoir créer des partages sur SERVEUR1 à partir de SERVEUR2 dans le laboratoire suivant (laboratoire 6F).

Dossiers personnels

Ce laboratoire doit être fait individuellement sur l'ordinateur virtuel 2

Objectifs

- Révision des autorisations NTFS, du champ d'application et de la propagation
- Révision des concepts et droits sur les partages
- Utilisation de la console "Gestionnaire de serveur \ Services de fichiers et de stockage \ Partages"
- Création d'une structure de dossiers personnels par l'interface graphique et par programmation

Directives générales

La configuration de dossiers personnels au sein d'une corporation demande une certaine planification. Afin de faciliter la gestion de ces dossiers les administrateurs de réseau vont préférer centraliser ces dossiers sur un serveur de fichiers et sous un même partage.

Selon les corporations et le besoin de sécurité, différents emplacements peuvent être utilisés. Par exemple au CVM, les professeurs et les étudiants ont des dossiers personnels sur des serveurs de fichiers différents.

Il est certain que la sécurité est un critère très important. Un utilisateur peut s'attendre à ce que ses informations personnelles soient à l'abri du regard des autres utilisateurs.

Dans le cadre de ce laboratoire, et afin de bien maîtriser les dossiers personnels, vous allez créer plusieurs structures de dossier

- Sur le serveur virtuel 1 en utilisant l'interface graphique du serveur virtuel 2
- Sur le serveur virtuel 1 par programmation PowerShell à partir du serveur virtuel 2

Utilisateurs pour le laboratoire

Pour ce laboratoire, vous devez créer deux utilisateurs.
Comme d'habitude, vous devez les créer à partir de SERVEUR2 avec la console UOAD. Vos deux utilisateurs seront créés dans l'AD (sur SERVEUR1).

- Nom complet et nom d'ouverture de session du premier utilisateur: U1
  Emplacement: CN=users,DC=formation,DC=local

- Nom complet et nom d'ouverture de session du deuxième utilisateur: U2
  Emplacement: CN=users,DC=formation,DC=local

Documentation sur les partages dans Windows

Permet de partager une ressource (dossier, imprimante, lecteur DVD, ...) avec d'autres utilisateurs en utilisant le réseau.

Windows utilise le protocole SMB "Server Message Block".
"Windows Serveur 2019" et "Windows 10" utilisent la version SMB 3.1.1

Pour accéder à un partage on utilise la syntaxe UNC (Universal Naming Convention)
\\NomOrdinateur\NomDuPartage

Les partages administratifs
Pour accéder aux partages administratifs (C$, D$, E$, ADMIN$, ...) on doit obligatoirement être membre du groupe Administrateurs.

Résumé des étapes que vous ferez dans ce laboratoire pour la configuration des dossiers personnels

Assigner un dossier personnel à un utilisateur lui permet de centraliser ses documents sur un serveur de fichiers.

Voici les trois grandes étapes à effectuer pour réussir la configuration des dossiers personnels.

Création du dossier racine et du partage
- Création d'un dossier racine qui va contenir les dossiers personnels des utilisateurs
- Attribution des autorisations NTFS adéquates sur le dossier racine
- Création d'un partage sur le dossier racine
- Attribution des autorisations de partage sur le dossier racine

Création des dossiers personnels pour les utilisateurs
- Création d'un sous-dossier pour l'utilisateur
- Attribution des autorisations NTFS adéquates sur le sous-dossier de l'utilisateur

Modification des propriétés des utilisateurs
- nom du dossier personnel
- lettre du dossier personnel

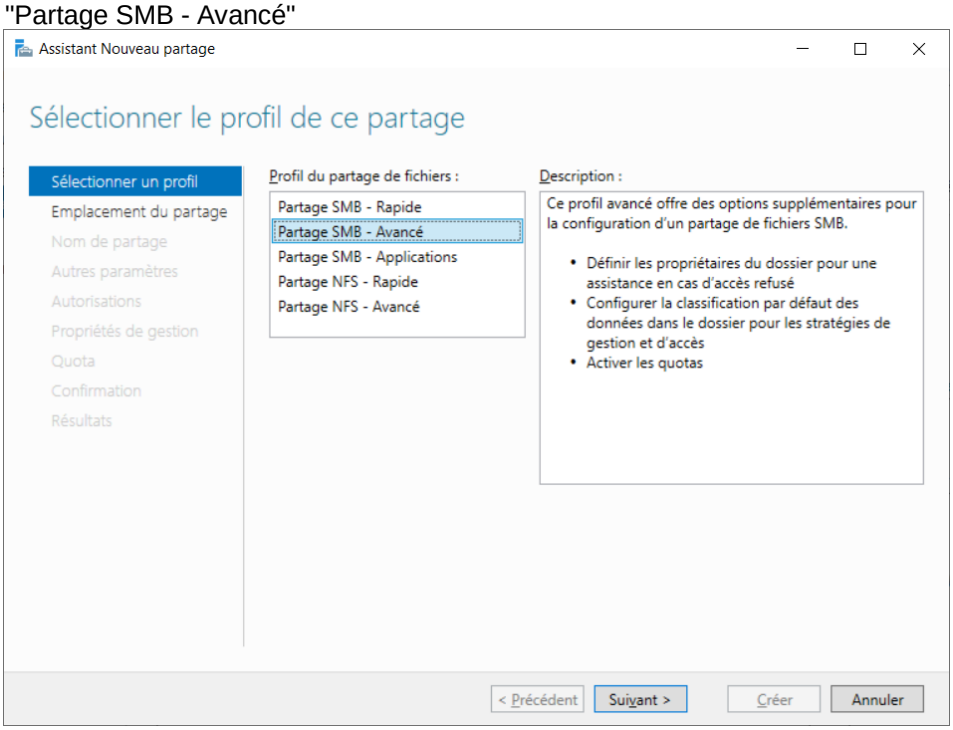
Étape 1 – Création d'un dossier personnel en utilisant les consoles

Les dossiers seront sur le SERVEUR1 mais le travail se fait à partir du SERVEUR2

Création du dossier racine et du partage

Dans la console "Gestionnaire de serveur \ Services de fichiers et de stockage \ Partages"

- Bouton "Tâche" de la fenêtre **"Ressources partagées"**
- Nouveau partage...
  - "Partage SMB - Avancé"

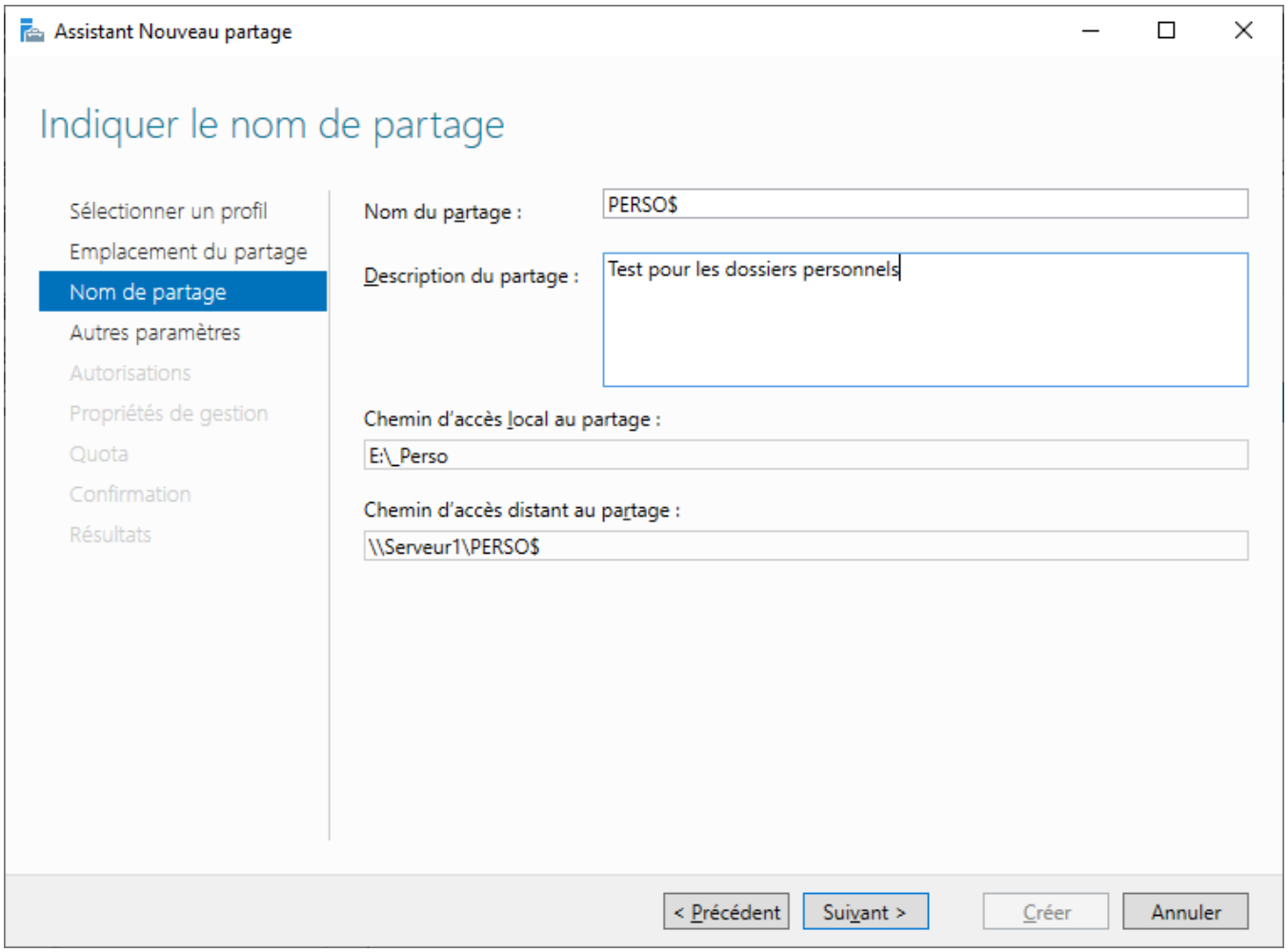


Sélectionner le serveur et le chemin d'accès au partage (encore ici, si SERVEUR1 n'apparaît pas dans la fenêtre, il est possible de l'ajouter (**Gestionnaire de serveur/Gérer/Ajouter des serveurs**).

- Serveur: **SERVEUR1**
- Tapez un chemin personnalisé: **E:_Perso**

Indiquer le nom de partage

- Nom de partage: **PERSO$**
- Description du partage: **Test pour les dossiers personnels**



Rappel du cours de système d'exploitation (C22) :
remarquez la syntaxe du chemin local au partage (donc, nom du dossier qui est partagé, à partir de la racine du E:\ de SERVEUR1 **(E:_Perso)**)
et
la syntaxe d'accès distant au partage, qu'on appelle aussi non UNC :
\\nom de l'ordinateur sur lequel est le partage\nom du partage
(\\Serveur1\PERSO$)

Configurer les paramètres de partage

- Cocher "**Activer l'énumération basée sur l'accès**"
- Décocher "**Autoriser la mise en cache du partage**"

Spécifier les autorisations pour contrôler l'accès

- Personnaliser les autorisations... (note :on parle ici d'autorisations NTFS sur E:_Perso)

ONGLET "Autorisations"

- **Désactiver l'héritage sur E:_Perso**
  - ✓ "Convertir les autorisations héritées en autorisations explicites sur cet objet"

VOICI LES AUTORISATIONS FINALES QUE DOIT AVOIR LE DOSSIER

FORMATION\TECH	Contrôle total	Ce dossier, les sous-dossiers et les fichiers
FORMATION\Administrateurs	Contrôle total	Ce dossier, les sous-dossiers et les fichiers
Système	Contrôle total	Ce dossier, les sous-dossiers et les fichiers
DROITS DU PROPRIÉTAIRE	Modification	Ce dossier, les sous-dossiers et les fichiers
Utilisateurs du domaine	Lecture et exécution	Ce dossier seulement

ONGLET "Partage"

- **Modifier**

Autoriser	"Tout le monde"	"Contrôle total"

Spécifier les propriétés de gestion des dossiers

- **Ne rien sélectionner**

Appliquer le quota à un dossier ou un volume

- **Ne pas appliquer de quota**

Confirmer les sélections

- **Cliquer sur le bouton "Créer"**

Création du dossier personnel pour l'utilisateur U1

Créer le dossier \\SERVEUR1\E$_PERSO\U1

Sur ce dossier, ajouter les autorisations NTFS suivantes:

- **U1 Modification Ce dossier, les sous-dossiers et les fichiers**

Modification des propriétés de l'utilisateur U1

Dans la console UOAD

Sélectionner l'utilisateur U1

Dans les propriétés de l'utilisateur U1, sélectionner l'onglet "Profil"

- Inscrire les propriétés pour le "Dossier de base"
  - **Connecter: X:**
  - **À: \\SERVEUR1\PERSO$\U1**

IMPORTANT: Il ne faut pas utiliser le chemin suivant: \\SERVEUR1\E$_PERSO\U1

Tester si l'utilisateur U1 a accès à son X: lorsqu'il se connecte à partir du serveur virtuel 2

Étape 2 – Création d'un dossier personnel par programmation PowerShell

Les dossiers seront sur le SERVEUR1 mais le travail se fait à partir du SERVEUR2

Écrire un script PowerShell pour créer plusieurs utilisateurs dans Active Directory

Création du dossier racine

```
Vous allez utiliser la commande New-Item

Avec New-Item, le chemin (-path) doit être un nom UNC si le dossier est distant :
$chemin = "\\SERVEUR1\E$\_PERSO2"

Vous devez chercher dans la syntaxe de New-Item pour les autres paramètres
```

Attribution des autorisations NTFS adéquates sur le dossier racine

```
# Pour désactiver l'héritage et supprimer les autorisations NTFS existantes

icacls.exe $chemin /inheritance:r

# Avec icacls.exe, le chemin doit être un nom UNC si le dossier est distant
# S-1-5-18 est le SID pour "Système"
# S-1-3-4  est le SID pour "DROITS DU PROPRIÉTAIRE"
icacls.exe $chemin /grant "Administrateurs:(OI)(CI)(F)"
icacls.exe $chemin /grant "TECH:(OI)(CI)(F)"
icacls.exe $chemin /grant "*S-1-5-18:(OI)(CI)(F)"
icacls.exe $chemin /grant "*S-1-3-4:(OI)(CI)(M)"
icacls.exe $chemin /grant "Utilisateurs du domaine:(RX)"
```

Création du partage sur le dossier racine

- Nom du dossier racine: **E:_Perso2**
- Nom du partage: **PERSO2$**
- Autorisations de partage: **"Contrôle total" pour "Tout le monde"**
- Autres paramètres
  - Activer l'énumération basée sur l'accès
  - Désactiver "Autoriser la mise en cache du partage"

```
# Avec New-SMBShare, le chemin est toujours un chemin local
# si le dossier est distant, il faut utiliser le paramètre -CIMSession

New-SMBShare -Name "PERSO2$" `
             -Path "E:\_Perso2" `
             -FullAccess "Tout le monde" `
             -FolderEnumerationMode AccessBased `
             -CachingMode none `
             -CIMSession "SERVEUR1"
```

création du dossier personnel pour l'utilisateur U2

Avec la commande New-Item

modification des autorisations NTFS sur le dossier personnel de l'utilisateur U2

Avec la commande ICACLS

modification des propriétés de l'utilisateur U2

la lettre pour accéder au dossier personnel de l'utilisateur U2 sera "X:"

```
# IMPORTANT: Il ne faut pas utiliser le chemin suivant: \\SERVEUR1\E$\_PERSO2\U2
Set-ADUser -Identity "U2" `
           -HomeDrive "X:" `
           -HomeDirectory "\\SERVEUR1\PERSO2$\U2"
```

Étape 3 - Création des dossiers PERSO pour les utilisateurs EMP01 à EMP32

Les utilisateurs EMP01 à EMP32 doivent avoir un dossier personnel.

Vous devez rechercher les utilisateurs de l'unité d'organisation FORMATION par programmation.

Les dossiers personnels seront sur le serveur virtuel 1

- Nom partage: **\\SERVEUR1\PERSO$**
  note: le partage PERSO$ existe déjà, vous l'avez créé à l'étape 1
- Nom du dossier pour les utilisateurs: **\\SERVEUR1\PERSO$\EMP***
  note: chaque utilisateur aura le droit "Modification" sur son dossier personnel
- Le dossier personnel de chaque utilisateur sera associé à la lettre X:
- Il n'y a pas de fichier .csv. Vous allez utiliser la commande GET_ADUSER pour aller chercher vos usagers :

```
$chemin = "OU=FORMATION,DC=FORMATION,DC=LOCAL"
$noms = (Get-ADuser -SearchBase $chemin -Filter 'SamAccountName -like "EMP*"').Name

foreach ($nom in $noms)
{ ……

}


#==========================================================================
# AUTEUR: Richard Jean
```

```
# DATE  : 10 février 2023
#
# Modification du dossier personnel de l'utilisateur U2
#==============================================================================
Clear-Host

$NomOrdi        = "SERVEUR1"
$NomDossierPerso = "_Perso2"
$chemin         = "\\$NomOrdi\E$\$NomDossierPerso"

$NomPartagePerso = "Perso2$"
$cheminUNC       = "\\$NomOrdi\$NomPartagePerso"

#Création du dossier racine
New-Item -Path $chemin `
        -ItemType directory

icacls.exe $chemin /inheritance:r

# S-1-5-18 est le SID pour "Système"
# S-1-3-4  est le SID pour "DROITS DU PROPRIÉTAIRE"
icacls.exe $chemin /grant "Administrateurs:(OI)(CI)(F)"
icacls.exe $chemin /grant "TECH:(OI)(CI)(F)"
icacls.exe $chemin /grant "*S-1-5-18:(OI)(CI)(F)"
icacls.exe $chemin /grant "*S-1-3-4:(OI)(CI)(M)"
icacls.exe $chemin /grant "Utilisateurs du domaine:(RX)"

#Création du partage sur le dossier racine
New-SMBShare -Name $NomPartagePerso `
           -Path "C:\$NomDossierPerso" `
           -FullAccess "Tout le monde" `
           -FolderEnumerationMode AccessBased `
           -CachingMode none `
           -CIMSession $NomOrdi

#==============================================================================
#Création du dossier personnel pour l'utilisateur U2
$nomUsager = "U2"

New-Item -Path "$chemin\$nomUsager" `
        -ItemType directory

icacls.exe "$chemin\$nomUsager" /grant $nomUsager":(OI)(CI)(M)"

#==============================================================================
#Modification des propriétés de l'utilisateur U2 pour son dossier personnel
Set-ADUser -Identity $nomUsager `
          -HomeDrive "X:" `
          -HomeDirectory "$cheminUNC\$nomUsager"
```

```
#==============================================================================
# AUTEUR: Richard Jean
# DATE  : 10 février 2023
#
```

```
# Tous les utilisateurs de la OU FORMATION
#
# Création de la structure des dossiers personnels
#==============================================================================
Clear-Host

$NomOrdi = "SERVEUR1"

$NomPersoPartage  = "PERSO$"

$lettre = "X:"

#recherche tous les utilisateurs de la OU FORMATION si le nom débute par EMP
$chemin = "OU=FORMATION,DC=FORMATION,DC=LOCAL"
$noms = (Get-ADuser -SearchBase $chemin -Filter 'SamAccountName -like "EMP*"').Name

foreach ($nom in $noms)
{
  #-------------------------------------------------------
  #Création du dossier personnel pour l'utilisateur
  $cheminPerso = "\\$NomOrdi\$NomPersoPartage\$nom"

  New-Item -Path $cheminPerso `
          -ItemType directory

  icacls.exe $cheminPerso /grant $nom":(OI)(CI)(M)"

  #-------------------------------------------------------
  #Modification des propriétés de l'utilisateur
  Set-ADUser -Identity $nom `
            -HomeDrive $lettre `
            -HomeDirectory $cheminPerso
}
```

Syntaxes Get-AdUser et autres CmdLets utiles

1. Get-Aduser

a)Get-AdUser

[-AuthType <ADAuthType>]

[-Credential <PSCredential>]

-Filter <String>

[-Properties <String[]>]

[-ResultPageSize <Int32>]

[-ResultSetSize <Int32>]

[-SearchBase <String>]

[-SearchScope <ADSearchScope>]

[-Server <String>]

[<CommonParameters>]

Exemple: Faire afficher la liste des usagers de l'Active Directory dont le nom dans l'AD ne commence pas par A en utilisant le paramètre FILTER :

Get-AdUser -Filter 'Name -notlike "a* " '

b)Get-AdUser

[-AuthType <ADAuthType>]

[-Credential <PSCredential>]

[-Identity] <ADUser> (**-Identity** est le nom d'ouverture de session (login) de l'usager)

[-Partition <String>]

[-Properties <String[]>]

[-Server <String>]

[<CommonParameters>]

Avec cette syntaxe, c'est le nom d'ouverture de session (nom de login) qui est utilisé dans la recherche.

Ex : Commande pour faire afficher l'information sur l'usager dont le nom d'ouverture de session est : MTardif

Ex : **Get-AdUser "MTardif"** (en omettant le mot optionnel Identity)

Autre différence entre -filter et -identity : un des deux (-identity)retourne un message d'erreur PSH si non trouvé et l'autre non (retourne rien). Donc, dans un cas, on a besoin d'un try catch et dans l'autre non.

2.Get-ADForest

Note : le paramètre -Filter n'est pas utilisé avec les commandes Get-AdDomain et Get-AdForest

```
Get-ADForest
  [-AuthType <ADAuthType>]
  [-Credential <PSCredential>]
  [-Current <ADCurrentForestType>]
  [-Server <String>]
  [<CommonParameters>]
```

```
Get-ADForest
  [-AuthType <ADAuthType>]
  [-Credential <PSCredential>]
  [-Identity] <ADForest>
  [-Server <String>]
  [<CommonParameters>]
```

3.Get-ADDomain

Note :le paramètre -Filter n'est pas utilisé avec les commandes Get-AdDomain et Get-AdForest

```
Get-ADDomain
  [-AuthType <ADAuthType>]
  [-Credential <PSCredential>]
  [-Current <ADCurrentDomainType>]
  [-Server <String>]
  [<CommonParameters>]
```

```
Get-ADDomain
  [-AuthType <ADAuthType>]
  [-Credential <PSCredential>]
  [-Identity] <ADDomain>
  [-Server <String>]
  [<CommonParameters>]
```

4.Get-AdOrganizationalUnit

5.Get-AdComputer

6.Get-AdGroup

7.Get-AdObject

GET-ADUSER SYNTAXE 1 (avec -Filter)

Get-AdUser

[-AuthType <ADAuthType>]

[-Credential <PSCredential>]

-Filter <String>

[-Properties <String[]>]

[-ResultPageSize <Int32>]

[-ResultSetSize <Int32>]

[-SearchBase <String>]

[-SearchScope <ADSearchScope>]

[-Server <String>]

[<CommonParameters>]

Les paramètres FILTER, PROPERTIES, SEARCHBASE et SEARCHSCOPE

(Exemples avec la commande Get-AdUser)

Syntaxe (non complète) : **Get-AdUser <-Filter> [-Properties] [-SearchBase] [-SearchScope]**

1.Le paramètre -FILTER : (sélectionne des objets)

Ex 1: Faire afficher tous les usagers de l'AD dont le nom est Pierre Lavoie

Get-AdUser -Filter 'Name -eq "Pierre Lavoie" '

DistinguishedName : CN=Pierre Lavoie,OU=Niveau0,DC=FORMATION,DC=LOCAL

Enabled : True

GivenName : Pierre *(Prénom de l'usager)*

Name : Pierre Lavoie *(Nom affiché dans l'AD)*

ObjectClass : user

ObjectGUID : f1ff5c10-dc60-417f-98cf-6141831becd8

SamAccountName : PLavoie *(Nom d'ouverture de session (login) de l'usager)*

SID : S-1-5-21-565238128-3990956042-1647380780-1107

Surname : Lavoie *(Nom de famille de l'usager)*

UserPrincipalName : plavoie@FORMATION.LOCAL

On remarque que ce sont seulement les principales propriétés qui sont affichées.

Ex 2 : Faire afficher tous les usagers de l'AD dont le nom commence par P

Get-AdUser -Filter 'Name -like "P*"'

DistinguishedName : CN=toto,OU=Niveau0,DC=FORMATION,DC=LOCAL

Enabled : True

GivenName : Pierre

Name : Pierre Lavoie

ObjectClass : user

ObjectGUID : f1ff5c10-dc60-417f-98cf-6141831becd8

SamAccountName : PLavoie

SID : S-1-5-21-565238128-3990956042-1647380780-1107

Surname : Lavoie

UserPrincipalName : plavoie@FORMATION.LOCAL

Ex 3: Faire afficher TOUS les usagers de l'AD

Get-AdUser -Filter *

DistinguishedName : CN=Administrateur,CN=Users,DC=FORMATION,DC=LOCAL

Enabled : True

GivenName :

Name : Administrateur

ObjectClass : user

ObjectGUID : 6cd33276-7bd0-4120-8831-4a6d08e1c3f8

SamAccountName : Administrateur

SID : S-1-5-21-565238128-3990956042-1647380780-500

Surname :

UserPrincipalName :

DistinguishedName : CN=Michelle Girard,CN=Users,DC=FORMATION,DC=LOCAL

Enabled : True

GivenName : Michelle

Name : Michelle Girard

ObjectClass : user

ObjectGUID : 7ac7e824-08fd-432b-8627-d9b7c79fe9f7

SamAccountName : Tech

SID : S-1-5-21-565238128-3990956042-1647380780-1106

Surname : Girard

UserPrincipalName : Tech@FORMATION.LOCAL

DistinguishedName : CN=Pierre Lavoie,OU=Niveau0,DC=FORMATION,DC=LOCAL

Enabled : True

GivenName : Pierre

Name : Pierre Lavoie

ObjectClass : user

ObjectGUID : f1ff5c10-dc60-417f-98cf-6141831becd8

SamAccountName : PLavoie

SID : S-1-5-21-565238128-3990956042-1647380780-1107

Surname : Lavoie

UserPrincipalName : plavoie@FORMATION.LOCAL

DistinguishedName : CN=Richard Jean,OU=test,DC=FORMATION,DC=LOCAL

Enabled : True

GivenName : Richard

Name : Richard Jean

ObjectClass : user

ObjectGUID : 022abe8a-76fb-499a-a3d2-fb9be21fbe99

SamAccountName : test1

SID : S-1-5-21-565238128-3990956042-1647380780-1108

Surname : Jean

UserPrincipalName : test1@FORMATION.LOCAL

<u>Principaux opérateurs à utiliser avec le paramètre FILTER :</u>

-eq -ne

-le -ge -lt -gt

-like -notlike

-and -or -not

2.Gestion des propriétés :

On a vu que par défaut, ce sont les propriétés de base (DistinguishedName, Enabled, GivenName, Name, ObjectClass, ObjectGUID, SamAccountName, SID, Surname, UserPrincipalName) qui sont affichées.

1.Ajouter une (ou des) propriété(s) à la liste de base : paramètre **-properties**

Ex : Ajouter la propriété "Numéro de téléphone" à la liste des propriétés de base affichées

Get-AdUser -Filter 'Name -eq "Pierre Lavoie"' -properties homephone

DistinguishedName : CN=Pierre Lavoie,OU=Niveau0,DC=FORMATION,DC=LOCAL

Enabled : True

GivenName : Pierre

HomePhone : 333-3333

Name : Pierre Lavoie

ObjectClass : user

ObjectGUID : f1ff5c10-dc60-417f-98cf-6141831becd8

SamAccountName : PLavoie

SID : S-1-5-21-565238128-3990956042-1647380780-1107

Surname : Lavoie

UserPrincipalName : plavoie@FORMATION.LOCAL

2.Faire afficher **moins** de propriétés que la liste des propriétés de base

2.1 SELECT utilisé avec le piping

Ex : Faire afficher SEULEMENT les propriétés "GivenName" et "Surname" des usagers de l'AD

Get-AdUser -Filter * | Select-Object GivenName,Surname

GivenName Surname

--- ---

Michelle Girard

Pierre Lavoie

Richard Jean

(Administrateur : pas de GivenName ni de Surname)

2.2 FT utilisé avec le piping

Ex : Faire afficher SEULEMENT les propriétés "GivenName" et "Surname" des usagers de l'AD

Get-AdUser -Filter * | FT GivenName, Surname

(Sortie format TABLE)

GivenName Surname

--- ---

Michelle Girard

Pierre Lavoie

Richard Jean

(Administrateur : pas de GivenName ni de Surname)

2.3 **Point** avec le nom de la propriété

Ex: Faire afficher SEULEMENT la propriété DistinguishedName

(Get-AdUser -Filter *).SamAccountName

Administrateur

Tech

PLavoie

Test1

Cette commande ne ramène pas les **noms** des propriétés, on a seulement les valeurs, contrairement à FT **et SELECT-Object.** C'est nécessaire d'avoir seulement les valeurs quand on se sert du Get-AdUser pour les mettre dans une variable pour par exemple l'utiliser dans un ForEach.

Exemple :

$Noms = **(Get-AdUser -Filter *).SamAccountName**

Si on fait afficher $Noms, on aura :

Administrateur

Tech

PLavoie

Test1

On peut ensuite se servir de $Noms dans un ForEach :

ForEach ($Nom in $Noms)……

Par contre, si on utilise FT dans l'affectation :

$Noms = **Get-AdUser -Filter * | FT SamAccountname**

Si on fait afficher $Noms, on aura :

SamAccountname

Administrateur

Tech

PLavoie

Test1

Note : pour **une seule** propriété à la fois

3.Trier les propriétés lors de l'affichage

(Get-AdUser -Filter *).SamAccountName | Sort-Object

3.Le paramètre -SearchBase

Définition :indique à partir de quel objet de l'Active Directory commencer la recherche

Exemple : pour commencer la recherche à partir de l'OU Corpo :

Get-AdUser -SearchBase "OU=Corpo,OU=Formation,DC=formation,DC=local"

4.Le paramètre -SearchScope

Définition : indique sur combien de niveaux faire la recherche

Exemple 1 : pour faire la recherche dans l'OU Corpo:

Get-AdUser `

-SearchBase "OU=Corpo,OU=Formation,DC=formation,DC=local" `

-SearchScope OneLevel

Exemple 2 : pour faire la recherche dans l'OU Corpo et dans les OU en dessous :

Get-AdUser `

-SearchBase "OU=Corpo,OU=Formation,DC=formation,DC=local" `

-SearchScope SubTree

Le paramètre SearchScope peut être utilisé sans SearchBase. La recherche commence alors en haut de l'arborescence.

GET-ADUSER SYNTAXE 2 (avec -Identity)

b)Get-AdUser

[-AuthType <ADAuthType>]

[-Credential <PSCredential>]

[-Identity] <ADUser> (**-Identity** est le nom d'ouverture de session (login) de l'usager)

[-Partition <String>]

[-Properties <String[]>]

[-Server <String>]

[<CommonParameters>]

Utilisé quand on veut rapidement avoir de l'information sur UN usager

Programmation avec le GET

Ce laboratoire doit être fait individuellement sur l'ordinateur virtuel 2

Objectif

- Utiliser les cmdlets de PowerShell afin de se familiariser avec les objets d'un domaine
- Maîtriser l'utilisation des cmdlets GET du module ActiveDirectory

Mise en place

Pour conserver dans un fichier texte le contenu de l'aide sur le paramètre "-Filter" des cmdlets de l'Active Directory en exécutant la commande suivante:

Get-Help about_ActiveDirectory_Filter | Out-File "c:\temp\About_ActiveDirectory_Filter.txt" -Encoding utf8

Si le fichier est vide mettre à jour l'aide du module ActiveDirectory avec la commande suivante:
Update-Help -Module ActiveDirectory

Exercice 1 - Accès à la structure du domaine

Dans l'invite PowerShell faire les commandes suivantes et répondre aux questions.

Get-ADForest

Trouvez les valeurs des attributs suivants:

Nom de l'attribut	Valeur de l'attribut
ForestMode	Windows2016Forest
DomainNamingMaster	Serveur1.FORMATION.LOCAL
GlobalCatalogs	Serveur1.FORMATION.LOCAL
SchemaMaster	Serveur1..FORMATION.LOCAL

Get-ADDomain

Trouvez les valeurs des attributs suivants:

Nom de l'attribut	Valeur de l'attribut
DistinguishedName	DC=FORMATION,DC=LOCAL
DNSRoot	FORMATION.LOCAL
DomainMode	Windows2016Domain
Name	FORMATION
PDCEmulator	Serveur1.FORMATION.LOCAL

- Écrire la commande qui retourne SEULEMENT le nom pleinement qualifié de votre domaine:
  réponse: `(Get-ADDomain).DistinguishedName`

- Écrire la commande qui retourne SEULEMENT le nom selon la norme DNS de votre domaine:
  réponse: `(Get-ADDomain).DNSRoot`

Exercice 2 - Accès aux objets d'un domaine

Utiliser le cmdlet **Get-ADcomputer** et au besoin utiliser **-Filter** et/ou **-Properties**

- Écrire la commande qui affiche les principales propriétés de votre ordinateur Serveur2.
  réponse: `Get-ADComputer -Filter 'Name -eq "Serveur2"'`

- Écrire la commande qui affiche toutes les propriétés de votre ordinateur Serveur2.

réponse: Get-ADComputer -Filter 'Name –eq "Serveur2"' -Properties *

- Écrire la commande qui affiche les principales propriétés des ordinateurs dont le nom commence par "Se"
  réponse: `Get-ADcomputer -Filter {name -like "Se*"}`

- Écrire la commande qui affiche le nom de tous les ordinateurs de votre domaine
  réponse: `(Get-ADComputer -Filter *).name`

- Écrire la commande qui affiche les propriétés "Name,DNSHostName,LastLogon" de tous les ordinateurs de votre domaine

ponse: Get-ADComputer -Filter * -Properties LastLogon | FT Name,DNSHostName,LastLogon ou Get-ADComputer -Filter * -Properties LastLogon | Select-object Name,DNSHostName,LastLogon

Note : l'information que vous donnent LastLogon (et AccountExpires) est en nano secondes.

Exemple avec LastLogon

```
Get-ADUser -Identity TECH `
           -Properties accountExpires,LastLogon

accountExpires      : 9223372036854775807
DistinguishedName   : CN=Richard Jean,CN=Users,DC=FORMATION,DC=LOCAL
Enabled             : True
GivenName           : Richard
LastLogon           : 132672441821805799
Name                : Richard Jean
ObjectClass         : user
ObjectGUID          : 58aae375-36dd-432e-b2c6-20395ad28cb1
SamAccountName      : TECH
SID                 : S-1-5-21-3523712887-290544647-1780752054-1104
Surname             : Jean
UserPrincipalName   : TECH@FORMATION.LOCAL

# Avec PowerShell, c'est facile de trouver la date qui correspond au LastLogon
[DateTime]::FromFileTime(132672441821805799)

3 juin 2021 21:36:22
```

3 juin 2021 21:36:22

Utiliser le cmdlet **Get-ADGroup**

- Écrire la commande qui affiche les principales propriétés du groupe Administrateurs
  réponse: `Get-ADGroup -Filter 'Name -eq "Administrateurs"'`
- Écrire la commande qui affiche toutes les propriétés du groupe Administrateurs
  réponse: Get-ADGroup -Filter 'Name –eq "Administrateurs"' -Properties *

Utiliser le cmdlet **Get-ADUser** et au besoin utiliser **-Filter** et/ou **-Properties**

- Écrire la commande qui affiche les principales propriétés de l'utilisateur dont le nom d'ouverture de session est Emp01

réponse: Get-ADuser –Filter 'SamAccountName –eq "Emp01"' ou Get-ADuser Emp01

- Écrire la commande qui affiche les principales propriétés des utilisateurs dont le nom dans l'AD est John Doe

réponse: Get-ADuser -Filter 'Name –eq "John Doe"'

- Écrire la commande qui affiche les principales propriétés ainsi que la propriété de son tél mobile de l'utilisateur dont le nom de famille est *Doe*

réponse: Get-ADuser -Filter 'SurName –eq "*Doe*"' -Properties Mobile

- Écrire la commande qui affiche toutes les propriétés de l'utilisateur dont le nom d'ouverture de session est Emp08

  Réponse Get-ADuser –Filter 'SamAccountName –eq "EMp08"' –Properties * ou Get-ADuser Emp08 –Properties *

- Écrire la commande qui affiche les principales propriétés des utilisateurs dont le nom d'ouverture de session débute par U

  réponse: Get-ADUser -Filter {SamaccountName -like "U*"} (On peut remplacer les apostrophes par des accolades dans un Filter)

- Écrire la commande qui affiche seulement la propriété **SID** de l'utilisateur dont le nom d'ouverture de session est Emp06

Get-ADuser –Filter 'SamAccountName –eq "EMp06"'| Select SID (on peut aussi utiliset FT au lieu de Select)

- Écrire la commande qui affiche seulement la propriété **Téléphone à la maison** des utilisateurs dont le nom de famille commence par D

Get-ADuser –Filter 'Surname –like "D*"' -Properties HomePhone | Select-object HomePhone (on peut aussi utiliser FT au lieu de Select-object) ou (Get-ADuser –Filter 'Surname –like "D*"' -Properties HomePhone).HomePhone

- Écrire la commande qui affiche seulement la propriété **Nom de famille** l'utilisateur Emp10. On veut ensuite mettre les résultats dans une variable pour les utiliser dans un ForEach.

  (Get-AdUser -Filter 'SamAccountname -eq "EMp10"').Surname

  Ou (Get-AdUser Emp10).Surname

  Ici, on ne peut pas utiliser Select-Object ou FT

- Écrire la commande qui affiche seulement la propriété **Prénom** des utilisateurs dont le nom de login commence par Emp et qui sont dans l'unité d'organisation Comptable

  Get-AdUser -Filter 'SamAccountname -like "Emp*"' -SearchBase "ou=comptables,ou=comptabilite,ou=employes,ou=formation,dc=formation,dc=local " -searchscope=onelevel |Select-object GivenName (on peut aussi utiliser FT au lieu de Select-object) ou (Get-ADuser –Filter ……onelevel).GivenName

Trouver la liste des utilisateurs dont le DistinguishedName se termine par ou=formation,dc=formation,dc=local

Essai 1: La commande ne fonctionne pas parce qu'on utilise **-like** avec la propriété **DistinguishedName** qui elle est calculée pendant l'exécution de **Get-ADUser**
~~**Get-aduser -Filter {DistinguishedName -like "*ou=formation,dc=formation,dc=local"}**~~

Essai 2: On doit utiliser **Where-Object** qui s'exécute sur le résultat de la commande qui est à gauche
Get-ADUser -Filter * | Where-Object {$PSItem.DistinguishedName -like "*ou=formation,dc=formation,dc=local"}

Utiliser le cmdlet **Get-ADOrganizationalUnit** et au besoin utiliser **-Filter** et/ou **-Properties**

- Écrire la commande qui affiche les principales propriétés de l'unité d'organisation FORMATION
  réponse: `Get-ADOrganizationalUnit "OU=formation,DC=formation,DC=local"`
- Écrire la commande qui affiche toutes les propriétés de l'unité d'organisation FORMATION
  réponse: `Get-ADOrganizationalUnit "OU=formation,DC=formation,DC=local" -properties *`
- Écrire la commande qui affiche les principales propriétés des unités d'organisation dont le nom contient la suite de caractères "mat"
  réponse: `Get-ADOrganizationalUnit -Filter {Name -like "*mat*"}`
- Écrire la commande qui affiche les principales propriétés des unités d'organisation dont le nom commence par "Tec"
  réponse: `Get-ADOrganizationalUnit -Filter {Name -like "Tec*"}`
- Écrire la commande qui affiche les principales propriétés de toutes les unités d'organisation de votre domaine
  réponse: `Get-ADOrganizationalUnit -Filter *`
- Écrire la commande qui affiche les propriétés "Name" et "DistinguishedName" de toutes les unités d'organisation de votre domaine
  réponse: `Get-ADOrganizationalUnit -Filter * | Format-Table Name,DistinguishedName`
- Écrire la commande qui affiche les propriétés "Name" et "Created" de toutes les unités de votre domaine
  réponse: `Get-ADOrganizationalUnit -Filter * -Properties created | Format-Table Name,Created`

Get-ADObject

Utiliser le cmdlet **Get-ADObject** et au besoin utiliser **-Filter** et/ou **-Properties**

Requête qui permet de trouver les objets de l'Active Directory dont le nom débute par "Ad"

$sb = "OU=FORMATION,DC=FORMATION,DC=LOCAL"
Get-ADObject -SearchBase $sb -Filter {name -like "Ad*"} **|** format-table DistinguishedName,ObjectClass