

*****Procédure Installation de Chocolatey avec Ansible *****

Contexte : Dans le cadre de la mise en place d'un environnement de test pour les développeurs, nous avons utilisé Ansible pour configurer automatiquement une VM Windows 10/11. L'objectif principal de cette procédure est l'installation de Chocolatey, qui nous servira par la suite à automatiser l'installation de VSCode, Git, Docker, etc.

Prérequis

- Une machine Windows 10/11 (client)
- Un conteneur Ansible déployé (dans un environnement Proxmox ici)
- Connectivité réseau entre la machine Ansible et le client Windows
- WinRM activé sur la machine Windows (port 5985/5986)

Étapes de la configuration

Configuration de WinRM sur la machine Windows

Activation de WinRM

```
winrm quickconfig -q
winrm set winrm/config/winrs '@{MaxMemoryPerShellMB="512"}'
winrm set winrm/config '@{MaxTimeoutms="1800000"}'
winrm set winrm/config/service '@{AllowUnencrypted="true"}'
winrm set winrm/config/service/auth '@{Basic="true"}'
```

Ouvre le port 5985 (par défaut pour WinRM en HTTP)

```
netsh advfirewall firewall add rule name="Ansible WinRM" dir=in  
action=allow protocol=TCP localport=5985
```

Redémarre le service WinRM

Restart-Service WinRM

Création d'un fichier d'inventaire pour Ansible

- Se rendre dans /etc/ansible/
- Créer votre fichier inventaire : `sudo touch inventory.ini`
- Modifier le fichier inventaire : `sudo nano inventory.ini`

[windows]

windows11 ansible_host=[192.168.xxx.xxx](#)

[windows:vars]

ansible_user="username de l'utilisateur (ayant des droits) de la machine windows cliente"

ansible_password="le mot de passe de cet utilisateur"

ansible_connection=winrm

ansible_winrm_transport=basic

ansible_winrm_server_cert_validation=ignore

Test de connectivité

```
ansible windows -m win_ping
```

Installation de Chocolatey avec Ansible

- Création d'un playbook :

```
cd etc/ansible/playbook/
```

```
sudo nano install_chocolatey.yml
```

Ne pas oublier d'ajouter "-" devant "name" -> - name: Installer xxx

- name: Installer Chocolatey sur Windows

hosts: windows

gather_facts: no

tasks:

- name: Téléchargement et exécution du script Chocolatey

win_shell: |

Set-ExecutionPolicy Bypass -Scope Process -Force

[System.Net.ServicePointManager]::SecurityProtocol =

[System.Net.ServicePointManager]::SecurityProtocol -bor 3072

iex ((New-Object System.Net.WebClient).DownloadString('https://
chocolatey.org/install.ps1'))

args:

executable: powershell.exe

- name: Vérifier que Chocolatey est bien installé

win_command: choco -v

register: choco_version

- name: Afficher la version de Chocolatey

debug:

msg: "Chocolatey version installée : {{ choco_version.stdout }}"

Débogage

Erreur : UNREACHABLE via WinRM (port 5986 ou 5985)

- Cause : Port fermé ou service WinRM non démarré
- Solution : S'assurer que le pare-feu Windows autorise les connexions WinRM et que le service est bien actif :

```
Start-Service WinRM
```

Erreur de téléchargement Chocolatey : "Le nom distant n'a pas pu être résolu : "chocolatey.org"

- Cause : Pas de connexion Internet sur la VM
- Solution : Vérifier que la machine cliente Windows a bien accès à Internet ou à un DNS fonctionnel.

Erreur de syntaxe PowerShell avec Set-ExecutionPolicy

- Cause : Tout était mis sur une seule ligne, PowerShell interprétait mal la commande
- Solution : Séparer correctement les instructions :
win_shell: | Set-ExecutionPolicy Bypass -Scope Process -Force

Lancer l'automatisation

```
ansible-playbook -i inventory.ini install_choco.yml --ask-become-pass
```

On précisera que la commande change en fonction des chemins et de l'arborescence du projet.

Si "install_choco.yml" était dans un dossier "playbook" on aurait eu :

```
ansible-playbook -i inventory.ini playbook/install_choco.yml --ask-become-pass
```

Si "inventory.ini" était dans un dossier "inventories" on aurait eu :

```
ansible-playbook -i inventories/inventory.ini install_choco.yml --ask-become-pass
```

Si "inventory.ini" était dans un dossier "inventories" et "install_choco.yml" était dans un dossier "playbook" on aurait eu :

```
ansible-playbook -i inventories/inventory.ini playbook/install_choco.yml --ask-become-pass
```