

# ANSIBLE



## Sommaires

- introduction ansible
- préparatif
- installation de ansible + autre module
- explication de l'inventaire
- création de playbook

## ➤ introduction ansible

Ansible est un outil d'automatisation informatique pour la gestion de configurations, le déploiement de logiciels et bien d'autre tâche. Ansible peut être intégrer sur de petite infrastructure comme sur de beaucoup plus grande et plus complexe car ce logiciel ne pas besoin d'agent relais pour fonctionner.

Ansible est également réputé pour son écosystème riche et dynamique. La communauté autour d'Ansible a développé une vaste bibliothèque de modules et de rôles disponibles via Ansible Galaxy, la plateforme de partage de la communauté. Ces modules et rôles couvrent une large gamme de tâches d'automatisation, des opérations de base du système à la gestion des applications les plus complexes, rendant Ansible extrêmement versatile et capable de s'adapter à presque tous les environnements informatiques.

## ➤ Préparatif

Avent de commencer l'installation de ansible il faut préparer toutes les machine Windows qui vont être utiliser pour cela il faut premièrement activer WinRM (Windows Remote Management).

Il faut lancer powershell en administrateur puis entrer le première commande : `winrm quickconfig` puis entrée : **y**

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

Testez le nouveau système multiplateforme PowerShell https://aka.ms/pscore6

PS C:\Windows\system32> winrm quickconfig
WinRM n'est pas configuré pour recevoir des demandes sur cet ordinateur.
Les modifications suivantes doivent être effectuées :

Démarrez le service WinRM.
Affectez un démarrage automatique à retardement au service WinRM.

Effectuer ces modifications [y/n] ? y

WinRM a été mis à jour pour recevoir des demandes.

Le type du service WinRM a été correctement modifié.
Le service WinRM a démarré.
WSManFault
  Message
    ProviderFault
      WSManFault
        Message = L'exception de pare-feu WinRM ne fonctionnera pas car l'un des types de connexion réseau de cet ordinateur est défini à Public. Changez le type de connexion réseau en Domaine ou Privé, puis recommencez.

Numéro d'erreur : -2144108183 0x80338169
L'exception de pare-feu WinRM ne fonctionnera pas car l'un des types de connexion réseau de cet ordinateur est défini à Public. Changez le type de connexion réseau en Domaine ou Privé, puis recommencez.

PS C:\Windows\system32> ■
```

On peut vérifier avec la commande `get-service winrm` si le service est bien lancé.

```
➤ Administrateur : Windows PowerShell
PS C:\Windows\system32> get-service winrm
Status     Name           DisplayName
-----     --           -----
Running    winrm          Gestion à distance de Windows (Gest...
```

Par la suite avec deux commandes on va activer les paramètres qu'il faut :

- `winrm set winrm/config/service/auth '@{Basic="true"}'`
- `winrm set winrm/config/service '@{AllowUnencrypted="true"}'`

```
➤ Administrateur : Windows PowerShell
PS C:\Windows\system32> winrm set winrm/config/service/auth '@{Basic="true"}'
Auth
  Basic = true
  Kerberos = true
  Negotiate = true
  Certificate = false
  CredSSP = false
  CbtHardeningLevel = Relaxed

PS C:\Windows\system32> ■
```

```
PS C:\Windows\system32> winrm set winrm/config/service '@{AllowUnencrypted="true"}'  
Service  
  RootSDDL = O:NG:BAD:P(A;;GA;;;BA)(A;;GR;;;IU)S:P(AU;FA;GA;;;WD)(AU;SA;GXGW;;;WD)  
  MaxConcurrentOperations = 4294967295  
  MaxConcurrentOperationsPerUser = 1500  
  EnumerationTimeoutms = 240000
```

Regarder si il est possible de crypter la communication winrm

F

## ➤ installation de ansible + autre module

Pour mon exemple je suis sur une machine debian.

Commande :

```
apt update -y  
apt install ansible -y  
apt install python3-pip -y  
python3 -m pip install --user --ignore-installed pywinrm  
python3 -c "import winrm"  
python3 -c "import requests"  
ansible-galaxy collection install -vvvv ansible.windows  
mkdir /var/ansible/  
mkdir /var/ansible/config_win/  
mkdir /var/ansible/win_playbooks/
```

```
nano /var/ansible/config_win/ansible.cfg
```

```
[defaults]
roles_path=/var/ansible/config_win/roles
inventory=/var/ansible/config_win/hosts
```

```
GNU nano 3.4                               /var/ansible/
[defaults]
roles_path=/var/ansible/config_win/roles
inventory=/var/ansible/config_win/hosts
```

## ➤ explication de l'inventaire

ouvrir le fichier hosts : nano /var/ansible/config\_win/hosts

attention le décalage n'est pas une tabulation mais deux espace

win: //création du groupe windows

hosts: // initialisation des hosts

test1: // nom donnez a un hosts

ansible\_host: 10.132.23.53 // ip de l'host

ansible\_user: ansible // identifiant de l'utilisateur pour faire les commande

ansible\_password: ansible1234 // mot de passe

ansible\_port: 5985 // port d'utilisation de winrm

ansible\_connection: winrm // protocole utiliser

#ansible\_winrm\_scheme: http // peut etre utiliser mais je ne sais pas pourquoi

ansible\_winrm\_server\_cert\_validation: ignore // ignorer la certification

```
ansible_python_interpreter: 'C:\program files\Python311\python.exe' // utiliser pour renforcer
                                         certain scrypte
```

```
test2: // nom du deuxième host
```

```
ansible_host: 10.132.23.66
ansible_user: administrateur
ansible_password: ADMIN1234
ansible_port: 5985
ansible_connection: winrm
ansible_winrm_scheme: http
ansible_winrm_server_cert_validation: ignore
ansible_python_interpreter: 'C:\program files\Python311\python.exe'
```

```
portable: // nouveau groupe
```

```
hosts:
```

```
portable1:
```

```
ansible_host: 10.132.23.72
ansible_user: ansible@iutchalon.prive
ansible_password: Ansible1234
ansible_port: 5985
ansible_connection: winrm
ansible_winrm_scheme: http
ansible_winrm_server_cert_validation: ignore
ansible_python_interpreter: 'C:\program files\Python311\python.exe'
```

```
GNU nano 3.4          .../config_win/hosts

win:
  hosts:
    test1:
      ansible_host: 10.132.23.53
      ansible_user: ansible
      ansible_password: ansible1234
      ansible_port: 5985
      ansible_connection: winrm
      ansible_winrm_scheme: http
      ansible_winrm_server_cert_validation: ignore
      ansible_python_interpreter: 'C:\program files\Python311\python.exe'

    test2:
      ansible_host: 10.132.23.66
      ansible_user: administrateur
      ansible_password: ADMIN1234
      ansible_port: 5985
      ansible_connection: winrm
      ansible_winrm_scheme: http
      ansible_winrm_server_cert_validation: ignore
      ansible_python_interpreter: 'C:\program files\Python311\python.exe'

portable:
  hosts:
    portable1:
      ansible_host: 10.132.23.72
      ansible_user: ansible@iutchalons.prive
      ansible_password: Ansible1234
      ansible_port: 5985
      ansible_connection: winrm
      ansible_winrm_scheme: http
      ansible_winrm_server_cert_validation: ignore
      ansible_python_interpreter: 'C:\program files\Python311\python.exe'
```

Il est possible de crypter le fichier d'inventaire pour sécuriser les mot de passe admin avec la fonction vault.

commande :

```
ansible-vault encrypt /var/ansible/config_win/hosts
```

Pour pouvoir utiliser les playbook avec l'inventaire chiffré la commande suivante est utilisée :

```
ansible-playbook /var/ansible/win_playbook/playbook.yml -i /var/ansible/config_win/hosts --ask-vaul-pass
```

Cela va vous demander le mot de passe utiliser pour crypté l'inventaire donc si le réseau est sur écoute les mot de passe peuvent passer en claire donc il faut utiliser winrm de façon sécurisé.

## ➤ création de playbook

Pour la création de playbook rien de plus simple :

```
nano /var/ansible/win_playbook/votre_playbook.yml
```

### playbook wak on lan

```
--- // pour annoncer à ansible que c'est le début du programme
- name: Envoyer un paquet Wake-on-LAN // le nom de ce que va faire le programme
  hosts: localhost // quel hosts va être utilisé (ex : le groupe win, juste une machine test1 ou comme
               ici local hosts
  gather_facts: false // collecter des informations sur l'host donc argument yes/no
  tasks: // début de la tâche
    - name: Utiliser le module command pour exécuter wakeonlan // nom de la tâche
      command: wakeonlan 74:E6:E2:E6:36:53 // commande qui s'exécute sur le localhost avec la
               commande wakeonlan pour un ordinateur
    - name: portable wake up
      command: wakeonlan 74:78:27:1B:F6:7A // même commande mais pour un autre ordinateur
```

```
---  
- name: Envoyer un paquet Wake-on-LAN  
  hosts: localhost  
  gather_facts: false  
  tasks:  
    - name: Utiliser le module command pour exécuter wakeonlan  
      command: wakeonlan 74:E6:E2:E6:36:53  
    - name: portable wake up  
      command: wakeonlan 74:78:27:1B:F6:7A
```

### playbook installer

```
---  
- name: Install Firefox using Chocolatey // nom du programme  
  hosts: test2 // hosts machine windows  
  tasks: // les tâches à exécuter  
    - name: Install Firefox // nom de la tâche
```

```
win_chocolatey: // utilisation du module windows chocolatey pour récupérer le logiciel
  name: firefox // nom du logiciel
  state: present // adication qu'il doit l'installer
  become_user: administrateur // utilisateur qui installe le logiciel
```

<https://community.chocolatey.org/packages?q=>  
lien de tous les logiciel qui peuvent être installer avec le module chocolatey