

Version : **2023.02**

Dernière mise-à-jour : 2023/08/25 01:31

# LCF804 - Utilisation d'Ansible avec Docker et Windows

## Contenu du Module

- **LCF804 - Utilisation d'Ansible avec Docker et Windows**
  - Contenu du Module
  - LAB #1 - Ansible et Docker
    - 1.1 - Présentation de Docker
    - 1.2 - Installer Docker
    - 1.3 - La Connexion à Docker
  - LAB #2 - Préparer Windows 10
    - 2.1 - Mettre à Jour PowerShell et .NET
    - 2.2 - Configurer WinRM
    - 2.3 - Consulter les Informations sur WinRM
    - 2.4 - Créer un Utilisateur Local pour Ansible
  - LAB #3 - Préparer le Contrôleur Ansible
    - 3.1 - Installer pywinrm
    - 3.2 - Tester la Configuration
  - LAB #4 - Travailler avec Ansible et Windows
    - 4.1 - Obtenir les Informations sur Windows10
    - 4.2 - Exécuter une Commande
    - 4.3 - Exécuter un script PowerShell
    - 4.4 - Installer un Logiciel avec Chocolatey
    - 4.5 - Créer un Utilisateur Local

# LAB #1 - Ansible et Docker

## 1.1 - Présentation de Docker

Docker est une application de virtualisation légère qui utilise des **images** et des **conteneurs**.

Une **image** est un paquet exécutable contenant tout ce qu'il est nécessaire afin d'exécuter un logiciel donné, incluant :

- le code
- un runtime
- des bibliothèques,
- des variables d'environnement
- des fichiers de configuration

Un **conteneur** est une instance de l'image en cours d'exécution en mémoire. Elle est isolée de l'environnement de l'hôte par défaut mais peut accéder à des fichiers et de ports de l'hôte selon la configuration.

Les conteneurs exécutent des applications nativement en utilisant le noyau de la machine hôte. De ce fait les performances d'un conteneur sont supérieures à celles d'une machine virtuelle qui doit passer par un hyperviseur pour accéder aux ressources de la machine hôte :

Docker existe en deux versions **Docker-CE** (Docker Community Edition) et **Docker-EE** (Docker Enterprise Edition). Pour consulter les différences entre les deux versions, consultez le lien <https://docs.docker.com/engine/installation/>.

## 1.2 - Installer docker

Docker n'est pas dans le dépôts de CentOS. Afin de l'installer il convient d'ajouter le dépôt de docker :

```
[trainee@centos8 roles]$ su -  
Password: fenestros  
[root@centos8 ~]# dnf config-manager --add-repo=https://download.docker.com/linux/centos/docker-ce.repo  
Adding repo from: https://download.docker.com/linux/centos/docker-ce.repo
```

```
[root@centos8 ~]# dnf list docker-ce
Docker CE Stable - x86_64
83 kB/s | 20 kB      00:00
Last metadata expiration check: 0:00:01 ago on Wed 09 Mar 2022 12:38:58 EST.
Available Packages
docker-ce.x86_64                                3:20.10.12-3.el8
docker-ce-stable
```

Installez ensuite la dépendance **containerd** de docker-ce :

```
[root@centos8 ~]# dnf install
https://download.docker.com/linux/centos/7/x86_64/stable/Packages/containerd.io-1.2.10-3.2.el7.x86_64.rpm
Last metadata expiration check: 0:06:03 ago on Wed 09 Mar 2022 12:38:58 EST.
containerd.io-1.2.10-3.2.el7.x86_64.rpm
13 MB/s | 23 MB      00:01
Dependencies resolved.
=====
=====
Package                Architecture                Version
Repository              Size
=====
=====
Installing:
  containerd.io          x86_64                      1.2.10-3.2.el7
@commandline            23 M
  replacing  runc.x86_64 1.0.2-1.module_el8.5.0+911+f19012f9

Transaction Summary
=====
=====
Install 1 Package

Total size: 23 M
```

```
Is this ok [y/N]: y
```

Installez maintenant le paquet **docker-ce** :

```
[root@centos8 ~]# dnf install docker-ce --nobest -y
```

Dernièrement, démarrez docker et vérifiez la version de Docker client et serveur :

```
[root@centos8 ~]# systemctl enable docker
[root@centos8 ~]# systemctl start docker
[root@centos8 ~]# systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/usr/lib/systemd/system/docker.service; disabled; vendor preset: disabled)
   Active: activating (start) since Wed 2022-03-09 12:48:23 EST; 51s ago
     Docs: https://docs.docker.com
  Main PID: 59410 (dockerd)
    Tasks: 16
   Memory: 37.7M
    CGroup: /system.slice/docker.service
            └─59410 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

Mar 09 12:48:23 centos8.ittraining.loc dockerd[59410]: time="2022-03-09T12:48:23.125986199-05:00" level=info
msg="Starting up"
Mar 09 12:48:23 centos8.ittraining.loc dockerd[59410]: time="2022-03-09T12:48:23.128455293-05:00" level=info
msg="parsed scheme: \"unix\"" module=grpc
Mar 09 12:48:23 centos8.ittraining.loc dockerd[59410]: time="2022-03-09T12:48:23.128518502-05:00" level=info
msg="scheme \"unix\" not registered, fallback to default scheme" module=grpc
Mar 09 12:48:23 centos8.ittraining.loc dockerd[59410]: time="2022-03-09T12:48:23.128568972-05:00" level=info
msg="ccResolverWrapper: sending update to cc: [{unix:///run/containerd/containerd.sock 0 <nil>}] <>"
Mar 09 12:48:23 centos8.ittraining.loc dockerd[59410]: time="2022-03-09T12:48:23.128603681-05:00" level=info
msg="ClientConn switching balancer to \"pick_first\"" module=grpc
Mar 09 12:48:28 centos8.ittraining.loc dockerd[59410]: time="2022-03-09T12:48:28.105356011-05:00" level=info
msg="parsed scheme: \"unix\"" module=grpc
Mar 09 12:48:28 centos8.ittraining.loc dockerd[59410]: time="2022-03-09T12:48:28.105423457-05:00" level=info
```

```
msg="scheme \"unix\" not registered, fallback to default scheme" module=grpc
Mar 09 12:48:28 centos8.ittraining.loc dockerd[59410]: time="2022-03-09T12:48:28.105455755-05:00" level=info
msg="ccResolverWrapper: sending update to cc: {[{unix:///run/containerd/containerd.sock 0 <nil>}] <>"
Mar 09 12:48:28 centos8.ittraining.loc dockerd[59410]: time="2022-03-09T12:48:28.105470536-05:00" level=info
msg="ClientConn switching balancer to \"pick_first\"" module=grpc
Mar 09 12:48:28 centos8.ittraining.loc dockerd[59410]: time="2022-03-09T12:48:28.106759173-05:00" level=error
msg="Failed to built-in GetDriver graph btrfs /var/lib/docker"
[root@centos8 ~]# docker --version
Docker version 20.10.12, build e91ed57
[root@centos8 ~]# docker version
Client: Docker Engine - Community
 Version:           20.10.12
 API version:       1.40
 Go version:        go1.16.12
 Git commit:        e91ed57
 Built:             Mon Dec 13 11:45:22 2021
 OS/Arch:           linux/amd64
 Context:           default
 Experimental:      true

Server: Docker Engine - Community
 Engine:
  Version:           19.03.15
  API version:       1.40 (minimum version 1.12)
  Go version:        go1.13.15
  Git commit:        99e3ed8919
  Built:             Sat Jan 30 03:15:19 2021
  OS/Arch:           linux/amd64
  Experimental:     false
 containerd:
  Version:           1.2.10
  GitCommit:        b34a5c8af56e510852c35414db4c1f4fa6172339
 runc:
  Version:           1.0.0-rc8+dev
```

```
GitCommit:      3e425f80a8c931f88e6d94a8c831b9d5aa481657
docker-init:
Version:        0.18.0
GitCommit:      fec3683
```



**Important** - Docker est composé de trois éléments : un serveur, un client et un ou plusieurs **Repositories** ou Dépôts en français.

Démarrez un conteneur dénommé **postgresql** en mode détaché à partir d'une image **CentOS** :

```
[root@centos8 ~]# docker run -d --name postgresql centos sleep infinity
Unable to find image 'centos:latest' locally
latest: Pulling from library/centos
ald0c7532777: Pull complete
Digest: sha256:a27fd8080b517143cbbbab9dfb7c8571c40d67d534bbdee55bd6c473f432b177
Status: Downloaded newer image for centos:latest
1028e25f81d14d685678794902cd599aa618eb283d80e274526d71ec9708e69d
```

Vérifiez que le conteneur fonctionne :

```
[root@centos8 ~]# docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS          NAMES
1028e25f81d1   centos   "sleep infinity"        About a minute ago   Up 30 seconds          postgresql
```

Ajoutez **trainee** au groupe **docker** afin qu'il puisse gérer les conteneurs puis redevenir l'utilisateur **trainee** :

```
[root@centos8 ~]# usermod -aG docker trainee
[root@centos8 ~]# exit
logout
[trainee@centos8 roles]$
```

Rejoignez le groupe **docker** en tant que **trainee** :

```
[trainee@centos8 roles]$ newgrp docker
```

### 1.3 - La Connexion à Docker

Créez maintenant le Rôle **docker** :

```
[trainee@centos8 roles]$ mkdir docker
```

Modifiez ensuite le fichier `playbook.yaml` :

```
[trainee@centos8 roles]$ vi playbook.yaml
[trainee@centos8 roles]$ cat playbook.yaml
---
- hosts: all
  tasks:
    - name: message
      debug: msg="Conteneur - {{ inventory_hostname }} sous {{ansible_distribution}}"
```

Modifiez le fichier **inventory** :

```
[trainee@centos8 roles]$ cp inventory inventory.old
[trainee@centos8 roles]$ vi inventory
[trainee@centos8 roles]$ cat inventory
postgresql ansible_connection=docker
```

Supprimez le fichier **/home/trainee/.ansible/roles/group\_vars/all.yaml** :

```
[trainee@centos8 roles]$ rm -rf /home/trainee/.ansible/roles/group_vars/all.yaml
```

Exécutez la commande **ansible-playbook** :

```
[trainee@centos8 roles]$ ansible-playbook -i inventory playbook.yaml
```

```
[DEPRECATION WARNING]: Ansible will require Python 3.8 or newer on the controller starting with Ansible 2.12.
Current version: 3.6.8 (default, Sep 10 2021, 09:13:53) [GCC 8.5.0 20210514 (Red Hat 8.5.0-3)].
This feature will be removed from ansible-core in version 2.12. Deprecation warnings can be disabled by setting
deprecation_warnings=False in ansible.cfg.
```

```
PLAY [all]
```

```
*****
*****
```

```
TASK [Gathering Facts]
```

```
*****
*****
```

```
ok: [postgresql]
```

```
TASK [message]
```

```
*****
*****
```

```
ok: [postgresql] => {
  "msg": "Conteneur - postgresql sous CentOS"
}
```

```
PLAY RECAP
```

```
*****
*****
```

```
postgresql          : ok=2    changed=0    unreachable=0    failed=0    skipped=0    rescued=0
ignored=0
```

## LAB #2 - Préparer Windows 10

Ansible est capable de travailler avec les versions de Windows™ suivantes :

- Windows™ 7,

- Windows™ 8.1,
- Windows™ 10,
- Windows™ 11,
- Windows™ Server 2008,
- Windows™ Server 2008 R2,
- Windows™ Server 2012,
- Windows™ Server 2012 R2,
- Windows™ Server 2016,
- Windows™ Server 2019,
- Windows™ Server 2022.

Ansible nécessite sur chaque hôte :

- PowerShell 3.0 ou supérieur
- Au moins .NET 4.0.

## 2.1 - Mettre à jour PowerShell et .NET

Afin de mettre à jour les versions de PowerShell et .NET, Ansible fournit un script appelé **Upgrade-PowerShell.ps1**. Exécutez PowerShell en tant qu'Administrateur :

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

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

PS C:\Windows\system32> $url =
"https://raw.githubusercontent.com/jborean93/ansible-windows/master/scripts/Upgrade-PowerShell.ps1"
PS C:\Windows\system32> $file = "$env:temp\Upgrade-PowerShell.ps1"
PS C:\Windows\system32> $username = "trainee"
PS C:\Windows\system32> $password = "a39dae707d"
PS C:\Windows\system32> (New-Object -TypeName System.Net.WebClient).DownloadFile($url, $file)
PS C:\Windows\system32> Set-ExecutionPolicy -ExecutionPolicy Unrestricted -Force
```

```
PS C:\Windows\system32> &$file -Version 5.1 -Username $username -Password $password -Verbose
COMMENTAIRES : 2019-10-24T09:36:23 - INFO - starting script
COMMENTAIRES : 2019-10-24T09:36:23 - INFO - current and target PS version are the same, no action is required
COMMENTAIRES : 2019-10-24T09:36:23 - INFO - clearing auto logon registry properties
```

Au cas où, vous pouvez enlever l'auto-login de l'administrateur et remettre la politique d'exécution à **Restricted** avec les commandes suivantes :

```
PS C:\Windows\system32> Set-ExecutionPolicy -ExecutionPolicy Restricted -Force
PS C:\Windows\system32> $reg_winlogon_path = "HKLM:\Software\Microsoft\Windows NT\CurrentVersion\Winlogon"
PS C:\Windows\system32> Set-ItemProperty -Path $reg_winlogon_path -Name AutoAdminLogon -Value 0
PS C:\Windows\system32> Remove-ItemProperty -Path $reg_winlogon_path -Name DefaultUserName -ErrorAction SilentlyContinue
PS C:\Windows\system32> Remove-ItemProperty -Path $reg_winlogon_path -Name DefaultPassword -ErrorAction SilentlyContinue
PS C:\Windows\system32>
```

## 2.2 - Configurer WinRM

Pour mettre en place un **listener** WinRM en HTTP et un **listener** WinRM en HTTPS, créer un certificat auto-signé et activer l'authentification de base sur le service WinRM, utilisez le script **ConfigureRemotingForAnsible.ps1** :

```
PS C:\Windows\system32> $url =
"https://raw.githubusercontent.com/ansible/ansible/devel/examples/scripts/ConfigureRemotingForAnsible.ps1"
PS C:\Windows\system32> $file = "$env:temp\ConfigureRemotingForAnsible.ps1"
PS C:\Windows\system32> (New-Object -TypeName System.Net.WebClient).DownloadFile($url, $file)
PS C:\Windows\system32> powershell.exe -ExecutionPolicy ByPass -File $file
Self-signed SSL certificate generated; thumbprint: 17502EEEAC259F4C76D0F199A4B803E94E980CAD
```

```
wxf          : http://schemas.xmlsoap.org/ws/2004/09/transfer
a            : http://schemas.xmlsoap.org/ws/2004/08/addressing
w            : http://schemas.dmtf.org/wbem/wsman/1/wsman.xsd
lang        : fr-FR
```

```
Address          : http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous
ReferenceParameters : ReferenceParameters
```

Ok.

```
PS C:\Windows\system32>
```

## 2.3 - Consulter les Informations sur WinRM

Pour voir la liste des **listeners** ainsi créés, utilisez la commande suivante :

```
PS C:\Windows\system32> winrm enumerate winrm/config/Listener
```

```
Listener
```

```
Address = *
Transport = HTTP
Port = 5985
Hostname
Enabled = true
URLPrefix = wsman
CertificateThumbprint
ListeningOn = 10.0.2.58, 10.29.0.34, 127.0.0.1, ::1
```

```
Listener
```

```
Address = *
Transport = HTTPS
Port = 5986
Hostname = DESKTOP-10V5NMP
Enabled = true
URLPrefix = wsman
CertificateThumbprint = 17502EEEAC259F4C76D0F199A4B803E94E980CAD
ListeningOn = 10.0.2.58, 10.29.0.34, 127.0.0.1, ::1
```

```
PS C:\Windows\system32>
```

Pour obtenir les informations concernant le certificat, exécutez les commandes suivantes en remplaçant le contenu de \$thumbprint avec la valeur que vous avez obtenue :

```
PS C:\Windows\system32> $thumbprint = "17502EEEAC259F4C76D0F199A4B803E94E980CAD"  
PS C:\Windows\system32> Get-ChildItem -Path cert:\LocalMachine\My -Recurse | Where-Object { $_.Thumbprint -eq  
$thumbprint } | Select-Object *
```

```
PSPath :  
Microsoft.PowerShell.Security\Certificate::LocalMachine\My\17502EEEAC259F4C76D0F199A4B803E94  
E980CAD  
PSParentPath : Microsoft.PowerShell.Security\Certificate::LocalMachine\My  
PSChildName : 17502EEEAC259F4C76D0F199A4B803E94E980CAD  
PSDrive : Cert  
PSProvider : Microsoft.PowerShell.Security\Certificate  
PSIsContainer : False  
EnhancedKeyUsageList : {Authentification du serveur (1.3.6.1.5.5.7.3.1)}  
DnsNameList : {DESKTOP-10V5NMP, DESKTOP-10V5NMP}  
SendAsTrustedIssuer : False  
EnrollmentPolicyEndPoint : Microsoft.CertificateServices.Commands.EnrollmentEndPointProperty  
EnrollmentServerEndPoint : Microsoft.CertificateServices.Commands.EnrollmentEndPointProperty  
PolicyId :  
Archived : False  
Extensions : {System.Security.Cryptography.Oid, System.Security.Cryptography.Oid,  
System.Security.Cryptography.Oid, System.Security.Cryptography.Oid}  
FriendlyName :  
IssuerName : System.Security.Cryptography.X509Certificates.X500DistinguishedName  
NotAfter : 22/10/2022 11:45:21  
NotBefore : 23/10/2019 11:45:21  
HasPrivateKey : True  
PrivateKey : System.Security.Cryptography.RSACryptoServiceProvider  
PublicKey : System.Security.Cryptography.X509Certificates.PublicKey
```

```
RawData          : {48, 130, 5, 49...}
SerialNumber     : 73F2EADFBABB86B7438B30B157A1C3FB
SubjectName      : System.Security.Cryptography.X509Certificates.X500DistinguishedName
SignatureAlgorithm : System.Security.Cryptography.Oid
Thumbprint       : 17502EEEAC259F4C76D0F199A4B803E94E980CAD
Version          : 3
Handle           : 2829331536176
Issuer           : CN=DESKTOP-10V5NMP
Subject          : CN=DESKTOP-10V5NMP
```

```
PS C:\Windows\system32>
```

Dernièrement pour obtenir des informations concernant le service, exécutez les deux commandes suivantes :

```
PS C:\Windows\system32> winrm get winrm/config/Service
Service
  RootSDDL = 0:NSG:BAD:P(A;;GA;;;BA)(A;;GR;;;IU)S:P(AU;FA;GA;;;WD)(AU;SA;GXGW;;;WD)
  MaxConcurrentOperations = 4294967295
  MaxConcurrentOperationsPerUser = 1500
  EnumerationTimeoutms = 240000
  MaxConnections = 300
  MaxPacketRetrievalTimeSeconds = 120
  AllowUnencrypted = false
  Auth
    Basic = true
    Kerberos = true
    Negotiate = true
    Certificate = false
    CredSSP = false
    CbtHardeningLevel = Relaxed
  DefaultPorts
    HTTP = 5985
```

```
HTTPS = 5986
IPv4Filter = *
IPv6Filter = *
EnableCompatibilityHttpListener = false
EnableCompatibilityHttpsListener = false
CertificateThumbprint
AllowRemoteAccess = true
```

```
PS C:\Windows\system32> winrm get winrm/config/Winrs
```

```
Winrs
```

```
AllowRemoteShellAccess = true
IdleTimeout = 7200000
MaxConcurrentUsers = 2147483647
MaxShellRunTime = 2147483647
MaxProcessesPerShell = 2147483647
MaxMemoryPerShellMB = 2147483647
MaxShellsPerUser = 2147483647
```

```
PS C:\Windows\system32>
```

## 2.4 - Créer un Utilisateur Local pour Ansible

Exécutez ensuite la commande suivante :

```
PS C:\Windows\system32> winrm configSDDL default
```

```
Service
```

```
RootSDDL =
```

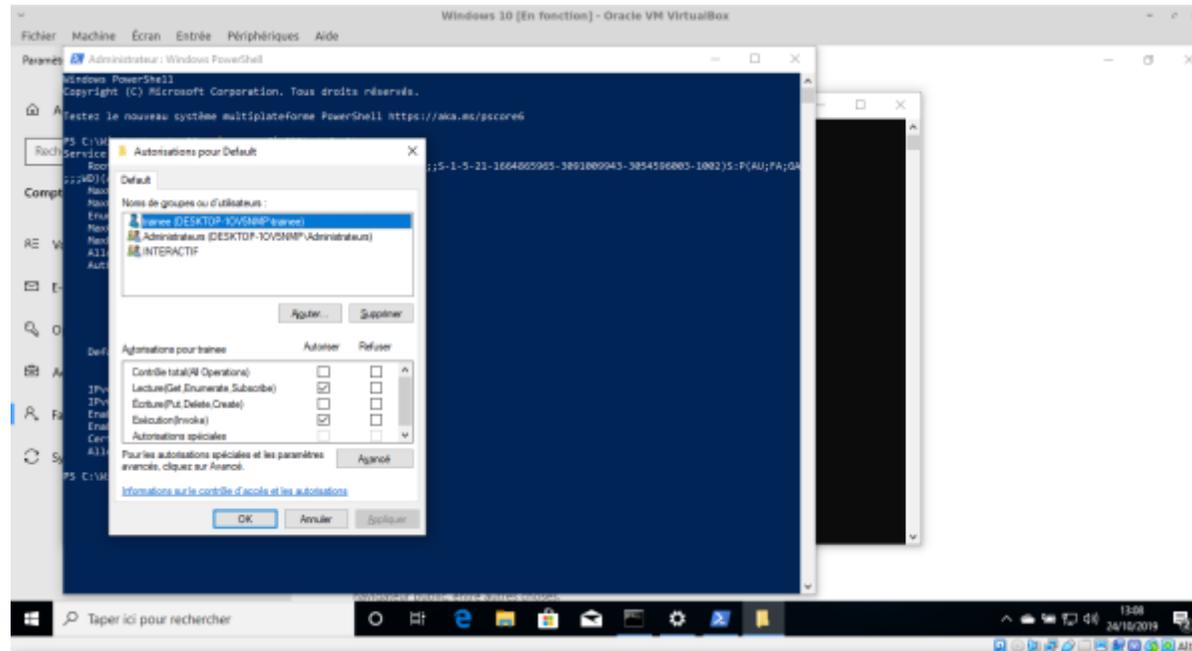
```
0:NSG:BAD:P(A;;;GA;;;BA)(A;;;GR;;;IU)(A;;;GXGR;;;S-1-5-21-1664865965-3091009943-3054596003-1002)S:P(AU;FA;GA;;;WD)(AU;SA;GXGW;;;WD)
```

```
MaxConcurrentOperations = 4294967295
MaxConcurrentOperationsPerUser = 1500
EnumerationTimeoutms = 240000
MaxConnections = 300
```

```
MaxPacketRetrievalTimeSeconds = 120
AllowUnencrypted = false
Auth
  Basic = true
  Kerberos = true
  Negotiate = true
  Certificate = false
  CredSSP = false
  CbtHardeningLevel = Relaxed
DefaultPorts
  HTTP = 5985
  HTTPS = 5986
IPv4Filter = *
IPv6Filter = *
EnableCompatibilityHttpListener = false
EnableCompatibilityHttpsListener = false
CertificateThumbprint
AllowRemoteAccess = true
```

```
PS C:\Windows\system32>
```

Dans la fenêtre **Autorisations pour Default**, ajoutez l'utilisateur pour Ansible ( trainee ) et accordez lui les permissions **Exécution** en plus des permissions **Lecture** :



## LAB #3 - Préparer le Contrôleur Ansible

### 3.1 - Installer pywinrm

Dans la machine virtuelle **centos8**, installez le support python pour WinRM :

```
[trainee@centos8 ~]$ su -  
Password: fenestros  
  
[root@centos8 ~]# pip3 install "pywinrm>=0.3.0"  
Collecting pywinrm>=0.3.0  
  Downloading pywinrm-0.4.2-py2.py3-none-any.whl (44 kB)  
    |████████████████████████████████████████| 44 kB 250 kB/s  
Collecting xmltodict  
  Downloading xmltodict-0.12.0-py2.py3-none-any.whl (9.2 kB)
```

```
Requirement already satisfied: requests>=2.9.1 in /usr/lib/python3.6/site-packages (from pywinrm>=0.3.0) (2.20.0)
Requirement already satisfied: six in /usr/lib/python3.6/site-packages (from pywinrm>=0.3.0) (1.11.0)
Collecting requests-ntlm>=0.3.0
  Downloading requests_ntlm-1.1.0-py2.py3-none-any.whl (5.7 kB)
Requirement already satisfied: chardet<3.1.0,>=3.0.2 in /usr/lib/python3.6/site-packages (from requests>=2.9.1->pywinrm>=0.3.0) (3.0.4)
Requirement already satisfied: idna<2.8,>=2.5 in /usr/lib/python3.6/site-packages (from requests>=2.9.1->pywinrm>=0.3.0) (2.5)
Requirement already satisfied: urllib3<1.25,>=1.21.1 in /usr/lib/python3.6/site-packages (from requests>=2.9.1->pywinrm>=0.3.0) (1.24.2)
Collecting ntlm-auth>=1.0.2
  Downloading ntlm_auth-1.5.0-py2.py3-none-any.whl (29 kB)
Requirement already satisfied: cryptography>=1.3 in /usr/local/lib64/python3.6/site-packages (from requests-ntlm>=0.3.0->pywinrm>=0.3.0) (36.0.1)
Requirement already satisfied: cffi>=1.12 in /usr/local/lib64/python3.6/site-packages (from cryptography>=1.3->requests-ntlm>=0.3.0->pywinrm>=0.3.0) (1.15.0)
Requirement already satisfied: pycparser in /usr/local/lib/python3.6/site-packages (from cffi>=1.12->cryptography>=1.3->requests-ntlm>=0.3.0->pywinrm>=0.3.0) (2.21)
Installing collected packages: ntlm-auth, xmltodict, requests-ntlm, pywinrm
Successfully installed ntlm-auth-1.5.0 pywinrm-0.4.2 requests-ntlm-1.1.0 xmltodict-0.12.0
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the
system package manager. It is recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv
```

Modifiez ensuite le fichier **/etc/hosts** :

```
[trainee@centos8 ~]$ su -
Password: fenestros
[root@centos8 ~]# vi /etc/hosts
[root@centos8 ~]# cat /etc/hosts
127.0.0.1    localhost localhost.localdomain localhost4 localhost4.localdomain4
::1        localhost localhost.localdomain localhost6 localhost6.localdomain6
10.0.2.45 centos8.ittraining.loc centos8
10.0.2.54 web01.i2tch.loc web01
10.0.2.55 web02.i2tch.loc web02
```

```
10.0.2.56 web03.i2tch.loc web03
10.0.2.57 web04.i2tch.loc web04
10.0.2.52 targeta.i2tch.loc targeta
10.0.2.53 targetb.i2tch.loc targetb
10.0.2.58 windows10
```

## 3.2 - Tester la Configuration

Pour tester la configuration, commencez par modifier le fichier inventory en tant que trainee :

```
[root@centos8 ~]# exit
logout
[trainee@centos8 ~]$ vi inventory
[trainee@centos8 ~]$ cat inventory
[windows]
windows10

[windows:vars]
ansible_user=trainee
ansible_password=a39dae707d
ansible_connection=winrm
ansible_winrm_server_cert_validation=ignore
```

Procédez au test :

```
[trainee@centos8 ~]$ ansible windows -i inventory -m win_ping
windows10 | SUCCESS => {
  "changed": false,
  "ping": "pong"
}
```

## LAB #4 - Travailler avec Ansible et Windows

### 4.1 - Obtenir les Informations sur Windows 10

De la même façon qu'avec Linux, il est possible d'utiliser le module **setup** pour obtenir des Facts :

```
[trainee@centos8 ~]$ ansible windows -i inventory -m setup
windows10 | SUCCESS => {
  "ansible_facts": {
    "ansible_architecture": "64 bits",
    "ansible_architecture2": "x86_64",
    "ansible_bios_date": "04/01/2014",
    "ansible_bios_version": "rel-1.14.0-0-g155821a1990b-prebuilt.qemu.org",
    "ansible_date_time": {
      "date": "2022-03-10",
      "day": "10",
      "epoch": "1646895432,89327",
      "epoch_int": 1646895433,
      "epoch_local": "1646899032,89327",
      "hour": "07",
      "iso8601": "2022-03-10T06:57:12Z",
      "iso8601_basic": "20220310T075712893274",
      "iso8601_basic_short": "20220310T075712",
      "iso8601_micro": "2022-03-10T06:57:12.893274Z",
      "minute": "57",
      "month": "03",
      "second": "12",
      "time": "07:57:12",
      "tz": "Romance Standard Time",
      "tz_offset": "+01:00",
      "weekday": "Thursday",
      "weekday_number": "4",
```

```
    "weeknumber": "9",
    "year": "2022"
  },
  "ansible_distribution": "Microsoft Windows 10 Professionnel",
  "ansible_distribution_major_version": "10",
  "ansible_distribution_version": "10.0.19042.0",
  "ansible_domain": "",
  "ansible_env": {
    "ALLUSERSPROFILE": "C:\\ProgramData",
    "APPDATA": "C:\\Users\\trainee\\AppData\\Roaming",
    "COMPUTERNAME": "DESKTOP-P0SAHCP",
    "ComSpec": "C:\\Windows\\system32\\cmd.exe",
    "CommonProgramFiles": "C:\\Program Files\\Common Files",
    "CommonProgramFiles(x86)": "C:\\Program Files (x86)\\Common Files",
    "CommonProgramW6432": "C:\\Program Files\\Common Files",
    "DriverData": "C:\\Windows\\System32\\Drivers\\DriverData",
    "HOMEDRIVE": "C:",
    "HOMEPATH": "\\Users\\trainee",
    "LOCALAPPDATA": "C:\\Users\\trainee\\AppData\\Local",
    "LOGONSERVER": "\\\\DESKTOP-P0SAHCP",
    "NUMBER_OF_PROCESSORS": "8",
    "OS": "Windows_NT",
    "OneDrive": "C:\\Users\\trainee\\OneDrive",
    "PATHEXT": ".COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC;.CPL",
    "PROCESSOR_ARCHITECTURE": "AMD64",
    "PROCESSOR_IDENTIFIER": "Intel64 Family 15 Model 6 Stepping 1, GenuineIntel",
    "PROCESSOR_LEVEL": "15",
    "PROCESSOR_REVISION": "0601",
    "PROMPT": "$P$G",
    "PSExecutionPolicyPreference": "Unrestricted",
    "PSModulePath": "C:\\Users\\trainee\\Documents\\WindowsPowerShell\\Modules;C:\\Program
Files\\WindowsPowerShell\\Modules;C:\\Windows\\system32\\WindowsPowerShell\\v1.0\\Modules",
    "PUBLIC": "C:\\Users\\Public",
    "Path":
```

```
"C:\\Windows\\system32;C:\\Windows;C:\\Windows\\System32\\Wbem;C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\;C:\\Windows\\System32\\OpenSSH\\;C:\\Users\\trainee\\AppData\\Local\\Microsoft\\WindowsApps",
  "ProgramData": "C:\\ProgramData",
  "ProgramFiles": "C:\\Program Files",
  "ProgramFiles(x86)": "C:\\Program Files (x86)",
  "ProgramW6432": "C:\\Program Files",
  "SystemDrive": "C:",
  "SystemRoot": "C:\\Windows",
  "TEMP": "C:\\Users\\trainee\\AppData\\Local\\Temp",
  "TMP": "C:\\Users\\trainee\\AppData\\Local\\Temp",
  "USERDOMAIN": "DESKTOP-P0SAHCP",
  "USERDOMAIN_ROAMINGPROFILE": "DESKTOP-P0SAHCP",
  "USERNAME": "trainee",
  "USERPROFILE": "C:\\Users\\trainee",
  "windir": "C:\\Windows"
},
"ansible_fqdn": "DESKTOP-P0SAHCP",
"ansible_hostname": "DESKTOP-P0SAHCP",
"ansible_interfaces": [
  {
    "connection_name": "Ethernet 2",
    "default_gateway": "10.0.2.1",
    "dns_domain": null,
    "interface_index": 13,
    "interface_name": "Intel(R) PRO/1000 MT Network Connection",
    "macaddress": "92:E1:64:91:2B:57"
  }
],
"ansible_ip_addresses": [
  "fe80::1c9:fbad:7f52:9542%13",
  "10.0.2.58"
],
"ansible_kernel": "10.0.19042.0",
"ansible_lastboot": "2022-03-10 07:13:15Z",
```

```
"ansible_machine_id": "S-1-5-21-309167381-963478272-52300377",
"ansible_memfree_mb": 14006,
"ansible_memtotal_mb": 16384,
"ansible_netbios_name": "DESKTOP-P0SAHCP",
"ansible_nodename": "DESKTOP-P0SAHCP",
"ansible_os_family": "Windows",
"ansible_os_installation_type": "Client",
"ansible_os_name": "Microsoft Windows 10 Professionnel",
"ansible_os_product_type": "workstation",
"ansible_owner_contact": "",
"ansible_owner_name": "trainee",
"ansible_pagefilefree_mb": 2432,
"ansible_pagefiletotal_mb": 2432,
"ansible_powershell_version": 5,
"ansible_processor": [
  "0",
  "GenuineIntel",
  "Common KVM processor",
  "1",
  "GenuineIntel",
  "Common KVM processor",
  "2",
  "GenuineIntel",
  "Common KVM processor",
  "3",
  "GenuineIntel",
  "Common KVM processor",
  "4",
  "GenuineIntel",
  "Common KVM processor",
  "5",
  "GenuineIntel",
  "Common KVM processor",
  "6",
```

```
    "GenuineIntel",
    "Common KVM processor",
    "7",
    "GenuineIntel",
    "Common KVM processor"
  ],
  "ansible_processor_cores": 8,
  "ansible_processor_count": 1,
  "ansible_processor_threads_per_core": 1,
  "ansible_processor_vcpus": 8,
  "ansible_product_name": "Standard PC (i440FX + PIIX, 1996)",
  "ansible_product_serial": null,
  "ansible_reboot_pending": true,
  "ansible_swaptotal_mb": 0,
  "ansible_system": "Win32NT",
  "ansible_system_description": "",
  "ansible_system_vendor": "QEMU",
  "ansible_uptime_seconds": 2647,
  "ansible_user_dir": "C:\\\\Users\\\\\\trainee",
  "ansible_user_gecos": "",
  "ansible_user_id": "trainee",
  "ansible_user_sid": "S-1-5-21-309167381-963478272-52300377-1001",
  "ansible_virtualization_role": "guest",
  "ansible_virtualization_type": "kvm",
  "ansible_win_rm_certificate_expires": "2025-03-08 07:34:49",
  "ansible_windows_domain": "WORKGROUP",
  "ansible_windows_domain_member": false,
  "ansible_windows_domain_role": "Stand-alone workstation",
  "gather_subset": [
    "all"
  ],
  "module_setup": true
},
"changed": false
```

```
}
```

## 4.2 - Exécutez une Commande

Créez le PlayBook **command.yml** :

```
[trainee@centos8 ~]$ vi command.yml
[trainee@centos8 ~]$ cat command.yml
---
- name: ipconfig
  hosts: windows
  tasks:
    - name: run ipconfig
      win_command: ipconfig
      register: ipconfig
    - debug: var=ipconfig
```

Exécutez le PlayBook :

```
[trainee@centos8 ~]$ ansible-playbook command.yml -i inventory
```

```
PLAY [ipconfig]
```

```
*****
*****
```

```
TASK [Gathering Facts]
```

```
*****
*****
```

```
ok: [windows10]
```

```
TASK [run ipconfig]
```

```
*****
*****
```



```
}
```

#### PLAY RECAP

```
*****  
*****  
windows10           : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0  
ignored=0
```

### 4.3 - Exécuter un script PowerShell

Créez un script PowerShell pour démarrer le service Windows Update :

```
[trainee@centos8 ~]$ vi script1.ps1  
[trainee@centos8 ~]$ cat script1.ps1  
Start-Service -Name wuauerv
```

Créez ensuite un PlayBook appelé **winupdate.yml** :

```
[trainee@centos8 ~]$ vi winupdate.yml  
[trainee@centos8 ~]$ cat winupdate.yml  
- name: "PowerShell script"  
  hosts: "windows"  
  gather_facts: "false"  
  tasks:  
    - name: "Windows Update"  
      script: "script1.ps1"
```

Exécutez le PlayBook :

```
[trainee@centos8 ~]$ ansible-playbook winupdate.yml -i inventory  
  
PLAY [PowerShell script]
```

```
*****
*****
TASK [Windows Update]
*****
*****
changed: [windows10]

PLAY RECAP
*****
*****
windows10          : ok=1    changed=1    unreachable=0    failed=0    skipped=0    rescued=0
ignored=0
```

## 4.4 - Installer un Logiciel avec Chocolatey

**Chocolatey** est un gestionnaire de paquets pour Windows™.

Créez le PlayBook **firefox.yml** :

```
[trainee@centos8 ~]$ vi firefox.yml
[trainee@centos8 ~]$ cat firefox.yml
---
- name: Install Firefox using Chocolatey
  hosts: all
  tasks:
    - name: Install Firefox
      win_chocolatey:
        name: firefox
        state: present
```

Exécutez le PlayBook :

```
[trainee@centos8 ~]$ ansible-playbook firefox.yml -i inventory

PLAY [Install Firefox using Chocolatey]
*****
*****

TASK [Gathering Facts]
*****
*****
ok: [windows10]

TASK [Install Firefox]
*****
*****
[WARNING]: Chocolatey was missing from this system, so it was installed during this task run.
changed: [windows10]

PLAY RECAP
*****
*****
windows10          : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0
ignored=0
```

## 4.5 - Créer un Utilisateur Local

Créez le PlayBook **user.yml** :

```
[trainee@centos8 ~]$ vi users.yml
[trainee@centos8 ~]$ cat users.yml
---
- name: Create a user
  hosts: windows
```

```
tasks:
  - name: Create Jean
    win_user:
      name: jean
      password: P@$w0rd
      state: present
      groups:
        - utilisateurs
```

Exécutez le PlayBook :

```
[trainee@centos8 ~]$ ansible-playbook users.yml -i inventory
```

```
PLAY [Create a user]
```

```
*****
*****
```

```
TASK [Gathering Facts]
```

```
*****
*****
```

```
ok: [windows10]
```

```
TASK [Create Jean]
```

```
*****
*****
```

```
changed: [windows10]
```

```
PLAY RECAP
```

```
*****
*****
```

```
windows10          : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0
ignored=0
```

Vérifiez que le compte a été créé :

