Ansible Installation

Ansible is an open source tool for automating tasks. It manages the configuration of your Linux and Windows servers. It works without an agent which means that Ansible uses SSH authorization.

You can use Ansible to automate three types of tasks:

- **Provisioning**: Set up several servers you need in your infrastructure.
- **Configuration management**: Change the configuration of an application, OS, or device; start and stop services; install or update applications; implement a security policy or perform a wide variety of other configuration tasks.
- **Application deployment**: Make DevOps easier by automating the deployment of internally developed applications to your production systems.

Ansible can be install various ways.

if you have internet connection , we can utilize epel repo

```
 #yum update          → First we need to update the OS to avoid dependencies related issues while installing package
```

 # yum install ansible

Once ansible is installed , we can use ansible --version command to check ansible version & installation path.

```
[root@ansible ~]# ansible --version
ansible 2.7.9
  config file = /etc/ansible/ansible.cfg
  configured module search path = [u'/root/.ansible/plugins/modules', u'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python2.7/site-packages/ansible
  executable location = /usr/bin/ansible
  python version = 2.7.5
```

If you doesn't have internet connection , we can install ansible offline method by downloading below packages.

ansible-2.7.9-1.el6.ans.noarch.rpm

libyaml-0.1.3-4.el6_6.x86_64

python-babel-0.9.4-5.1.el6.noarch

python-crypto-2.6.1-1.el6.rfx.x86_64

python-jinja2-2.2.1-3.el6.x86_64

python-paramiko-1.7.6-1.el6.rfx.noarch

python-setuptools-0.6.10-4.el6_9.noarch

python-six-1.9.0-2.el6.noarch

PyYAML-3.10-3.1.el6.x86_64

sshpass-1.05-1.el6.rf.x86_64

Each files need to be install one by one **#rpm -ivh packagename.rpm**

Ansible configuration

We have to create ansible configuration file and inventory file on below directory.

#/home/asp2manage/ansible/

ansible.cfg & inventory

```
[root@ansible ansible]# cat ansible.cfg
[defaults]
remote_user=asp2manage
inventory=/home/asp2manage/ansible/inventory
timeout=60
host_key_check=false
[privilege_escalation]
become=True
```

```
[root@ansible ansible]# cat inventory
[nodes]
target-ipaddress  ansible_connection=ssh  ansible_password=*******   ansible_become_password=*******
target-ipaddress ansible_connection=ssh  ansible_password=*******   ansible_become_password=*******
target-ipaddress ansible_connection=ssh  ansible_password=*******   ansible_become_password=*******
target-ipddresss ansible_connection=ssh  ansible_password=********  ansible_become_password=*******

[all:vars]
ansible_user=asp2manage
ansible_become_user=as2manage
```

Below output shows ansible control node able to reach managed nodes

```
[root@ansible ansible]# ansible all -m ping
192.168.56.101 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
192.168.56.106 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
```