

Setting up Ansible on EC2 instances

- Create a key pair or use existing
- Create two ec2 instances



- We need to create a user on both instances
- aws ec2 instances donot support password based authentication by default.
- Lets enable password based authentication for the time being

```
sudo vi /etc/ssh/sshd_config
# Change passwordAuthentication to yes
sudo service sshd restart
```

- Lets create a user called as devops

```
sudo adduser devops
```

- We need to provide sudo access without password to devops user

```
sudo visudo
```

```

GNU nano 6.2 /etc/sudoers.tmp *
# Ditto for GPG agent
#Defaults:%sudo env_keep += "GPG_AGENT_INFO"

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root    ALL=(ALL:ALL) ALL

# Members of the admin group may gain root privileges
%admin  ALL=(ALL) ALL

# Allow members of group sudo to execute any command
%sudo  ALL=(ALL:ALL) ALL
devops  ALL=(ALL:ALL) NOPASSWD:ALL
# See sudoers(5) for more information on "@include" directives:

@includedir /etc/sudoers.d

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify

```

- Now lets install ansible on ansible control node as python exists on both machines

```

sudo apt update
sudo apt install software-properties-common
sudo add-apt-repository --yes --update ppa:ansible/ansible
sudo apt install ansible -y

```

- Now execute `ansible --version` on ansible control node

```

devops@ip-172-31-32-106:~$ ansible --version
ansible [core 2.16.8]
  config file = /etc/ansible/ansible.cfg
  configured module search path = ['/home/devops/.ansible/plugins/modules', '/usr/share/ansible/
plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  ansible collection location = /home/devops/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/bin/ansible
  python version = 3.10.12 (main, Nov 20 2023, 15:14:05) [GCC 11.4.0] (/usr/bin/python3)
  jinja version = 3.0.3
  libyaml = True
devops@ip-172-31-32-106:~$

```

- Now lets do the connectivity check

```
echo '172.31.44.144' > hosts
ansible -i hosts -m ping -k all
```

```
devops@ip-172-31-32-106:~$ ansible -i hosts -m ping -k all
SSH password:
172.31.44.144 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
devops@ip-172-31-32-106:~$
```

- Now lets create a key pair for devops user on ansible control node

```
ssh-keygen
```

```
devops@ip-172-31-32-106:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/devops/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/devops/.ssh/id_rsa
Your public key has been saved in /home/devops/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:Ak+uE0ZeInI4wA5rl0ejLF0bCNiWH0N8bLck7cLcCJg devops@ip-172-31-32-106
The key's randomart image is:
+---[RSA 3072]-----+
|+. *.. .
|=oE.=o= +
|Bo*.XO+B .
|.X @.O= +
|. + = +.S
|. o .
| o
|.
+-----[SHA256]-----+
devops@ip-172-31-32-106:~$
```

- Now lets copy the public key to the node1 from ansible control node

```
devops@ip-172-31-32-106:~$ ssh-copy-id devops@172.31.44.144
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/devops/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are
already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to
install the new keys
devops@172.31.44.144's password:

Number of key(s) added: 1

Now try logging into the machine, with:  "ssh 'devops@172.31.44.144'"
and check to make sure that only the key(s) you wanted were added.

devops@ip-172-31-32-106:~$
```

- From this moment ansible control node can communicate with node 1 using its private key

```
devops@ip-172-31-32-106:~$ ssh 172.31.44.144
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 6.5.0-1022-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Fri Jul  5 02:33:53 UTC 2024

System load:  0.0           Processes:            102
Usage of /:   23.6% of 7.57GB Users logged in:      1
Memory usage: 24%          IPv4 address for eth0: 172.31.44.144
Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

1 update can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Fri Jul  5 02:29:14 2024 from 172.31.32.106
devops@ip-172-31-44-144:~$
```

- Now ansible is configured to use private key

```
devops@ip-172-31-32-106:~$ ansible -i hosts -m ping all
172.31.44.144 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
devops@ip-172-31-32-106:~$
```

- Exercise: Try configuring ansible with 3 node setup
 - 1 ansible control node
 - 2 nodes