TechPledge®
Connect.Consult.Content

Lab Manual- Setup and Manage Ansible for Azure DevOps

Prepared for: TechPledge

Date: 18th Nov 2018

Prepared by: Bipin Sinhaa

Document Name: Lab Manual

Document Number DevOpsLab308

Contributor:

Shruti Sinhaa

TechPledge®
Connect.Consult.Content

Table of Contents

1 OBJECTIVE

Ansible is a powerful configuration management solution written in python. There are many configuration management solutions available, all with pros and cons, ansible stands apart from many of them for its simplicity. What makes ansible different than many of the most popular configuration management systems is that its agent-less, no need to setup agents on every node you want to control. Deploying your software. In This Lab will cover the basics of using Ansible with Azure Cloud Shell.

- Active Azure Cloud Cell
- Create Ansible Playbook

- Run Ansible Playbook

2 PRE-REQUISISTE

- Prior knowledge of Linux
- Accounts in Azure

3 How Ansible Works

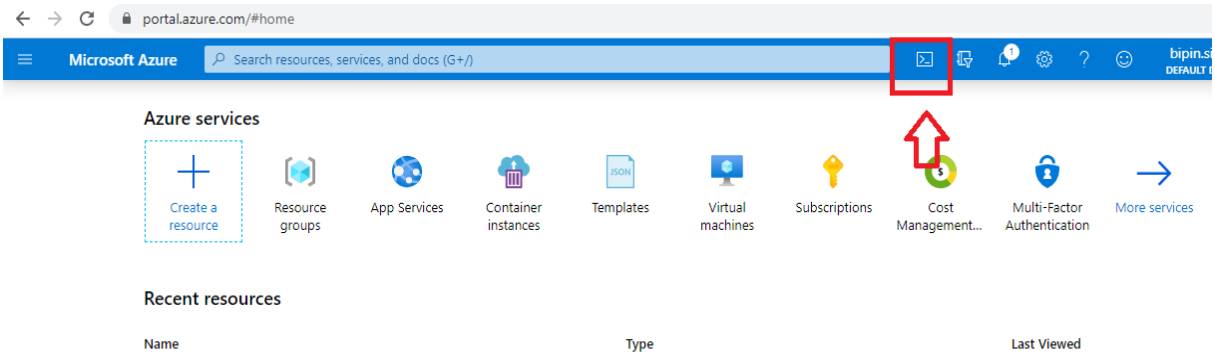
- There are many similar automation tools available like **Puppet, Capistrano, Chef, Salt, Space Walk** etc, but Ansible categorize into two types of server: **controlling machines** and **nodes.**
- The controlling machine, where Ansible is installed and Nodes are managed by this controlling machine over **SSH.** The location of nodes are specified by controlling machine through its **inventory**.
- The controlling machine (Ansible) deploys modules to nodes using SSH protocol and these modules are stored temporarily on remote nodes and communicate with the Ansible machine through a **JSON** connection over the standard output.
- Ansible is agent-less, that means no need of any agent installation on remote nodes, so it means there are no any background daemons or programs are executing for Ansible, when it's not managing any nodes.
- Ansible can handle **100's** of nodes from a single system over **SSH connection** and the entire operation can be handled and executed by one single command 'ansible'. But, in some cases, where you required to execute multiple commands for a deployment, here we can build playbooks.
- **Playbooks** are bunch of commands which can perform multiple tasks and each **playbooks** are in **YAML** file format.

4 Setup Up Ansible with Azure Cloud Shell

Cloud Shell is essentially a **browser based CLI** that you run from directly within the Azure portal.

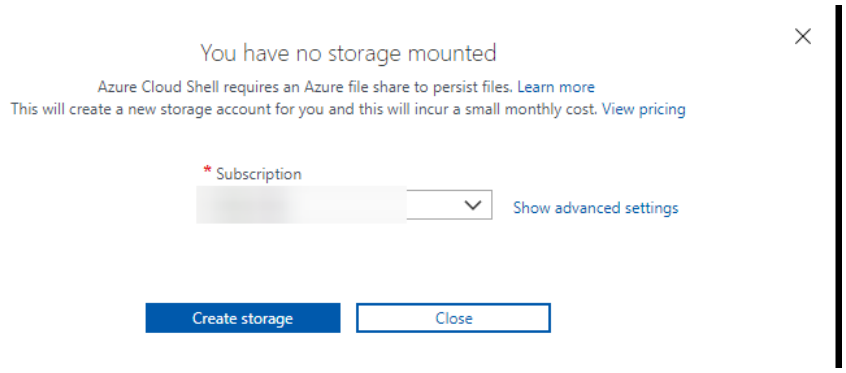
4.1 Enable Cloud Shell

- Click the >_ icon, located at the top right of the Azure portal.



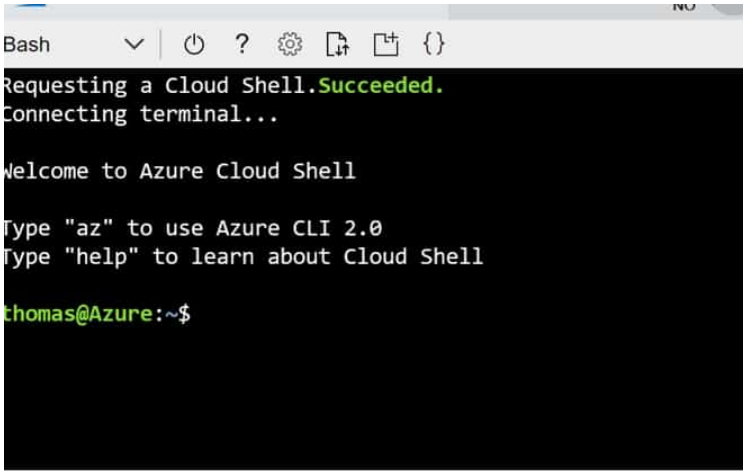
- The browser will display a new frame at the bottom of the window.

- Click **Bash Shell**
- When you sign in using Cloud Shell for the first time you will have to create a storage account



- Click Show Advance Storage and defined all parameters, the regions you should choose **"EastUS"** and Create Resource group**,** Storage Account and Fileshare and Click **Create Storage.**

- Now You will see the prompt.



4.2 Create Ansible Playbook

4.2.1 Create Azure Resource Group

Step1: Create an YAML playbook

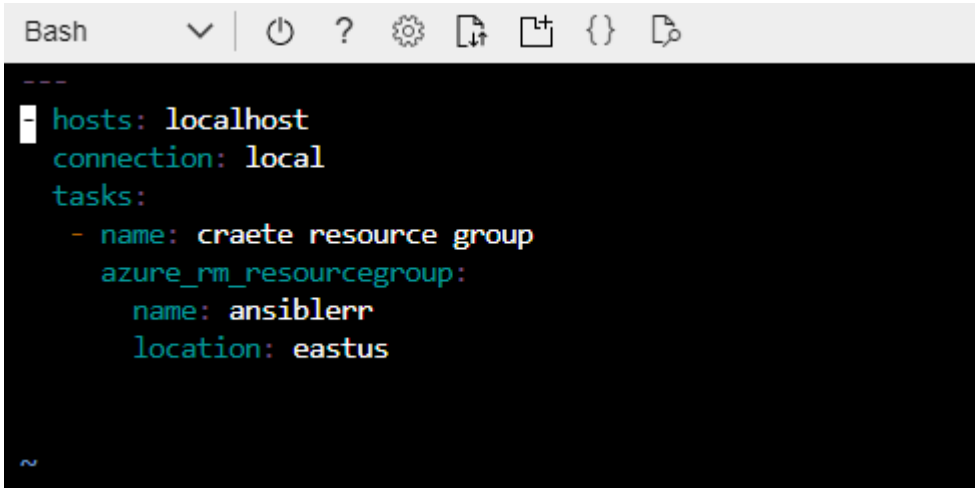
$ vi test.yaml

```
bipeen@Azure:~$ vi test.yaml
```

Step2 : Write YAML code as below:

```
---
- hosts: localhost
```

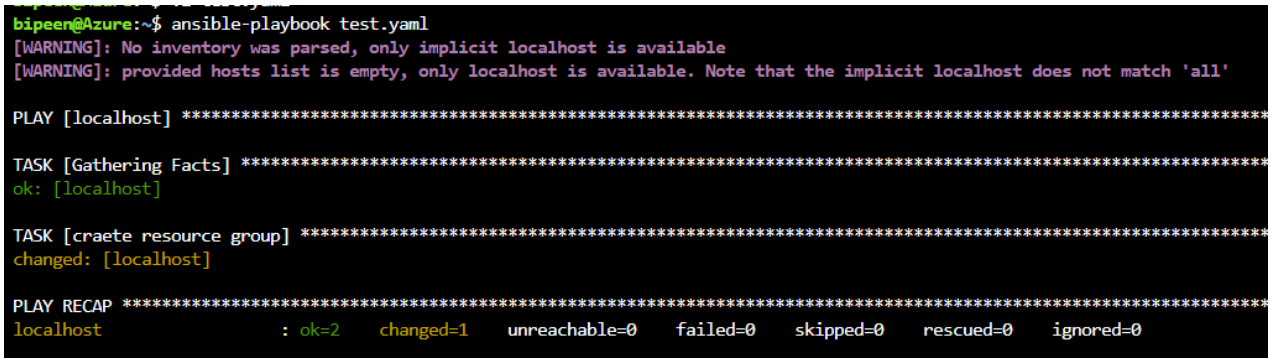
```
connection: local
tasks:
  - name: craete resource group
    azure_rm_resourcegroup:
      name: ansiblerr
      location: eastus
```



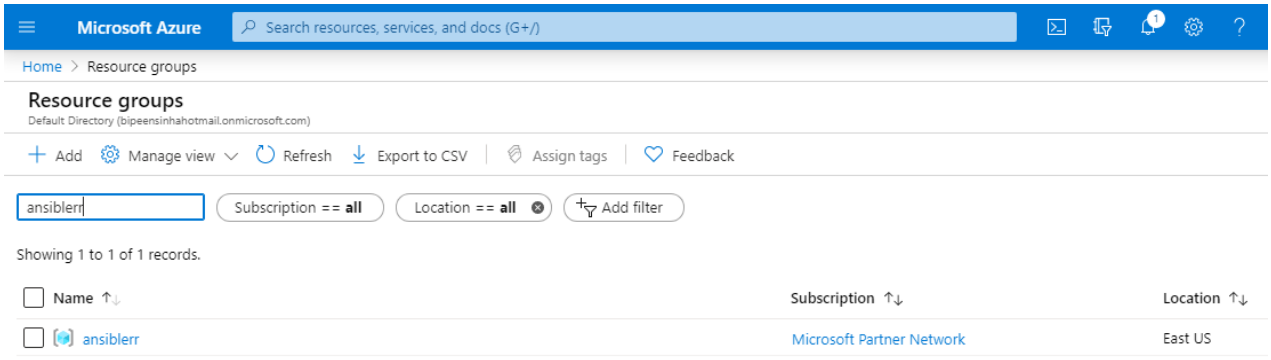
Step 3: Type CTRL+[and type shift+: and the wq

Step 4 : Type Below comkamnd to run the playbook

$ ansible-playbook test.yaml



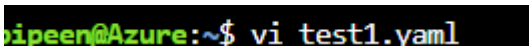
Step 5: Check the Azure Portal you will see the Resource group ansiblerr has been created



4.2.2 Create Virtual Network

Step1: Create an YAML playbook

$ vi test1.yaml



Step2 : Write YAML code as below:

```
---
- hosts: localhost
  connection: local
  tasks:
   - name: Create Virtual Network
     azure_rm_virtualnetwork:
       resource_group: ansiblerr
       name: ansiblevnet
       address_prefixes: "10.0.0.0/16"
    - name: Add Subnet in Vnet
      azure_rm_subnet:
        name: techpledgesub
        resource_group: ansiblerr
        address_prefix: "10.0.1.0/24"
```

```
virtual_network: ansiblevnet
```

```
Bash
---
- hosts: localhost
  connection: local
  tasks:
   - name: Create Virtual Network
     azure_rm_virtualnetwork:
       resource_group: ansiblerr
       name: ansiblevnet
       address_prefixes: "10.0.0.0/16"
   - name: Add Subnet in Vnet
     azure_rm_subnet:
        name: techpledgesub
        resource_group: ansiblerr
        address_prefix: "10.0.1.0/24"
        virtual_network: ansiblevnet
```

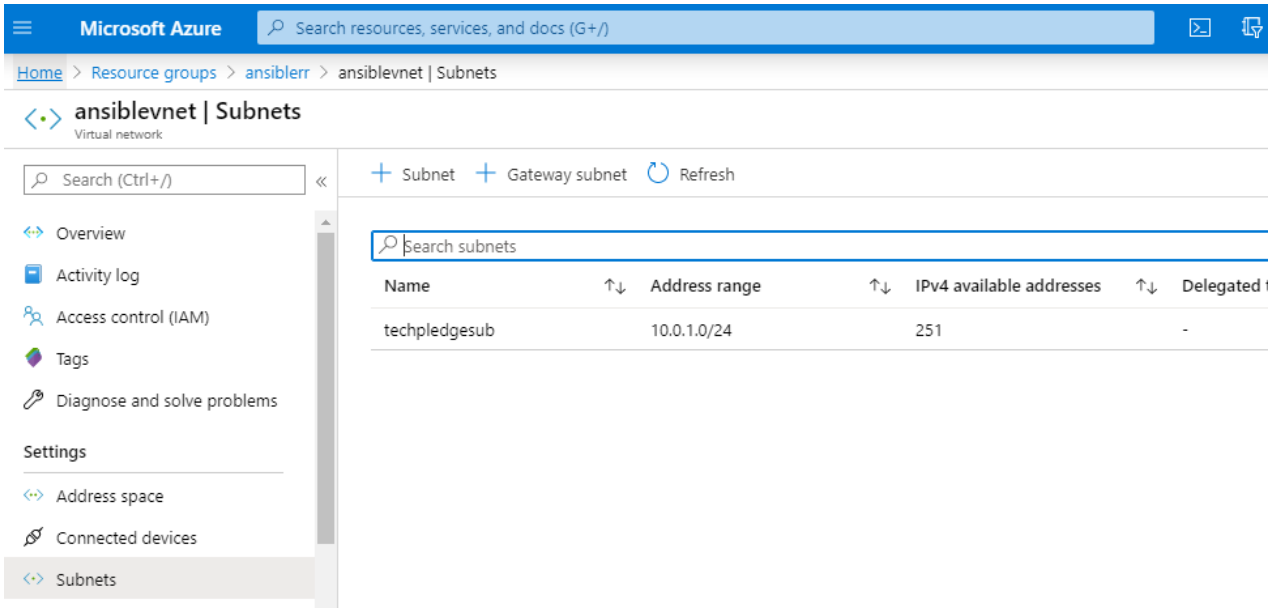
Step 3: Type CTRL+[and type shift+: and the wq

Step 4 : Type Below comkamnd to run the playbook

```
$ ansible-playbook test1.yaml
```

```
bipeen@Azure:~$ ansible-playbook test1.yaml
[WARNING]: No inventory was parsed, only implicit localhost is available
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'

PLAY [localhost] ***************************************************************************************************************

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

TASK [Create Virtual Network] **************************************************************************************************
ok: [localhost]

TASK [Add Subnet in Vnet] ******************************************************************************************************
changed: [localhost]

PLAY RECAP *********************************************************************************************************************
localhost                  : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

Step 5: Check the Azure Portal you will see the Resource group ansiblerr and notice the Virtual Network ansiblevnet and Subnet techpledgesub has been created.



4.2.3 Create Resource Group , Virtual Network and Linux Virtual Machine

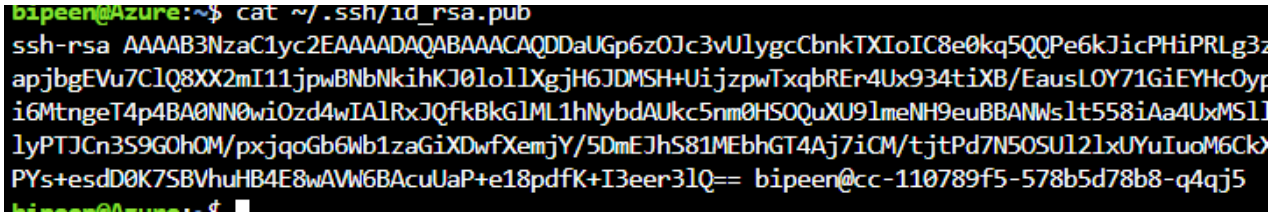
Step1: First Generqate bthe SSH key to be used for Linux

$ ssh-keygen -m PEM -t rsa -b 4096

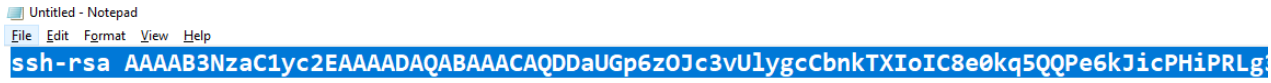
```
bipeen@Azure:~$ ssh-keygen -m PEM -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/bipeen/.ssh/id_rsa):
Created directory '/home/bipeen/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/bipeen/.ssh/id_rsa.
Your public key has been saved in /home/bipeen/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:Q4tMzn/17JJTazSgyI/ivmSBPjWRH8CmF7XlZDK0Hz0 bipeen@cc-110789f5-578b5d78b8-q4qj5
The key's randomart image is:
+---[RSA 4096]----+
|      ..o= +     |
|       +o X .    |
|      o+.= o E   |
|     .*.= + ...  |
|     ..O.S.....  |
|    . . +o... o+ |
|     o o .o.  +oo|
|      +. ... +.o |
|      o+o     +. |
+----[SHA256]-----+
```

Step2: Now view the key using cat command

$ cat ~/.ssh/id_rsa.pub



Step3: Now copy the key and paste it in notepad and without **word wrap**



Step4: Create an YAML playbook

$ vi test2.yaml

Step5: Write below code , please replace the `key_data:` from your own key (copy and paste from notepad)

$ vi test2.yaml

```
- name: Create Azure VM
  hosts: localhost
  connection: local
  tasks:
  - name: Create resource group
    azure_rm_resourcegroup:
      name: myResourceGroup
      location: eastus
  - name: Create virtual network
    azure_rm_virtualnetwork:
      resource_group: myResourceGroup
      name: myVnet
      address_prefixes: "10.0.0.0/16"
  - name: Add subnet
    azure_rm_subnet:
```

```
      resource_group: myResourceGroup
      name: mySubnet
      address_prefix: "10.0.1.0/24"
      virtual_network: myVnet
  - name: Create public IP address
    azure_rm_publicipaddress:
      resource_group: myResourceGroup
      allocation_method: Static
      name: myPublicIP
    register: output_ip_address
  - name: Dump public IP for VM which will be
created
    debug:
      msg: "The public IP is {{
output_ip_address.state.ip_address }}."
  - name: Create Network Security Group that
allows SSH
    azure_rm_securitygroup:
      resource_group: myResourceGroup
      name: myNetworkSecurityGroup
      rules:
        - name: SSH
          protocol: Tcp
          destination_port_range: 22
          access: Allow
          priority: 1001
          direction: Inbound
  - name: Create virtual network interface card
    azure_rm_networkinterface:
      resource_group: myResourceGroup
      name: myNIC
      virtual_network: myVnet
      subnet: mySubnet
      public_ip_name: myPublicIP
      security_group: myNetworkSecurityGroup
  - name: Create VM
    azure_rm_virtualmachine:
      resource_group: myResourceGroup
      name: myVM
      vm_size: Standard_DS1_v2
      admin_username: azureuser
```

```
        ssh_password_enabled: false
        ssh_public_keys:
          - path:
/home/azureuser/.ssh/authorized_keys
            key_data: ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAACAQDDaUGp6zOJc3vUlygcCbnkTXIoIC8e0kq5QQPe6kJicPHiPR
Lg3zJ8zijT32PRqpPoST+ljL3CZl1TKX/cgXOOx7QZVy/C7SxV9vMdLnNZ/xnzYWyOU5xXzFw6Uxp8
HvUzapjbgEVu7ClQ8XX2mI11jpwBNbNkihKJOlollXgjH6JDMSH+UijzpwTxqbREr4Ux934tiXB/Ea
usLOY71GiEYHcOypLV6WX6pj9Bjuj+oSSkOKv7RxiIIViyX/Gw5ZZAiWjAp2uusTx6h1HtzjywfnVC
rId1MDB8isIdOztui6MtngeT4p4BAONNOwiOzd4wIAlRxJQfkBkGlML1hNybdAUkc5nmOHSOQuXU9l
meNH9euBBANWslt558iAa4UxMSlljkL8dRiDkoIYnSckuaWbJiRi/XP/TKk7ujR6pOwZ1ON+7PDKwN
f5ouukDuVmg7KYgZ+bAmtOCdQOXslyPTJCn3S9GOhOM/pxjqoGb6Wb1zaGiXDwfXemjY/5DmEJhS81
MEbhGT4Aj7iCM/tjtPd7N5OSUl2lxUYuIuoM6CkXQJ9Mh8O7NE4lwEnsDPl3ROZeE46LgRFvJnpP+w
4q58IfmOfRDmyVLPRwLyBp1K/hsTYLVG6ZDmgOjzPYs+esdDOK7SBVhuHB4E8wAVW6BAcuUaP+e18p
dfK+I3eer3lQ== bipeen@cc-110789f5-578b5d78b8-q4qj5
        network_interfaces: myNIC
        image:
          offer: CentOS
          publisher: OpenLogic
          sku: '7.5'
          version: latest
```

```
    rules:
      - name: SSH
        protocol: Tcp
        destination_port_range: 22
        access: Allow
        priority: 1001
        direction: Inbound
  - name: Create virtual network interface card
    azure_rm_networkinterface:
      resource_group: myResourceGroup
      name: myNIC
      virtual_network: myVnet
      subnet: mySubnet
      public_ip_name: myPublicIP
      security_group: myNetworkSecurityGroup
  - name: Create VM
    azure_rm_virtualmachine:
      resource_group: myResourceGroup
      name: myVM
      vm_size: Standard_DS1_v2
      admin_username: azureuser
      ssh_password_enabled: false
      ssh_public_keys:
        - path: /home/azureuser/.ssh/authorized_keys
          key_data: ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAACAQDDaUGp6zOJc3vUlygcCbnkTXIoIC8e0kq5QQPe6kJicPHiPRLg3z
YWy0U5xXzFw6Uxp8HvUzapjbgEVu7ClQ8XX2mI11jpwBNbNkihKJ0lollXgjH6JDMSH+UijzpwTxqbREr4Ux934tiXB/EausLOY71GiEYHcOyp
fnVCrId1MDB8isId0ztui6MtngeT4p4BA0NN0wiOzd4wIAlRxJQfkBkGlML1hNybdAUkc5nm0HSOQuXU9lmeNH9euBBANWslt558iAa4UxMSll
Vmg7KYgZ+bAmtOCdQ0XslyPTJCn3S9GOhOM/pxjqoGb6Wb1zaGiXDwfXemjY/5DmEJhS81MEbhGT4Aj7iCM/tjtPd7N5OSUl2lxUYuIuoM6CkX
- name: Create Azure VM
  hosts: localhost
  connection: local
  tasks:
  - name: Create resource group
    azure_rm_resourcegroup:
      name: myResourceGroup
      location: eastus
  - name: Create virtual network
    azure_rm_virtualnetwork:
      resource_group: myResourceGroup
      name: myVnet
      address_prefixes: "10.0.0.0/16"
  - name: Add subnet
    azure_rm_subnet:
      resource_group: myResourceGroup
      name: mySubnet
      address_prefix: "10.0.1.0/24"
      virtual_network: myVnet
  - name: Create public IP address
    azure_rm_publicipaddress:
      resource_group: myResourceGroup
      allocation_method: Static
      name: myPublicIP
    register: output_ip_address
  - name: Dump public IP for VM which will be created
    debug:
      msg: "The public IP is {{ output_ip_address.state.ip_address }}."
  - name: Create Network Security Group that allows SSH
```

```
network_interfaces: myNIC
image:
  offer: CentOS
  publisher: OpenLogic
  sku: '7.5'
  version: latest
SERT --
```

Step 6: Type CTRL+[and type shift+: and the wq

Step 7 : Type Below comkamnd to run the playbook

$ ansible-playbook test2.yaml

```
TASK [Create virtual network interface card] ***********************************************************************
[DEPRECATION WARNING]: Setting ip_configuration flatten is deprecated and will be removed. Using ip_configurations list to c
will be removed in version 2.9. Deprecation warnings can be disabled by setting deprecation_warnings=False in ansible.cfg.
ok: [localhost]

TASK [Create VM] ***************************************************************************************************
[WARNING]: Module did not set no_log for ssh_password_enabled
changed: [localhost]

PLAY RECAP *********************************************************************************************************
localhost                  : ok=9    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

```
bipeen@Azure:~$ ansible-playbook test2.yaml
[WARNING]: No inventory was parsed, only implicit localhost is available
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'

PLAY [Create Azure VM] *********************************************************************************************

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

TASK [Create resource group] ***************************************************************************************
ok: [localhost]

TASK [Create virtual network] **************************************************************************************
ok: [localhost]

TASK [Add subnet] **************************************************************************************************
ok: [localhost]

TASK [Create public IP address] ************************************************************************************
ok: [localhost]

TASK [Dump public IP for VM which will be created] *****************************************************************
ok: [localhost] => {
    "msg": "The public IP is 52.226.68.231."
}

TASK [Create Network Security Group that allows SSH] ***************************************************************
ok: [localhost]

TASK [Create virtual network interface card] ***********************************************************************
```

Step 5: Check the Azure Portal you will see the Resource group myresourcegroup and Virtaul Machine , Stoarge in that resource grou.