## **steps to successfully build a VM and login to it via SSH.**

## 

## • Setup a [free] account on Azure (You need to do this yourself).

## • Install Ansible, python3-pip, the Azure Ansible modules.

## • Install Azure CLI and login to your account.

## • Create a service principal and a assign role.

## • Get the account details for azure and create a credentials file:

## o SubscriptionID, tenant, ClientID & Password.

## • Test Ansible connectivity

## • Create a Resource group

## • Write a playbook to create a VM in Azure.

## • Then delete the resource group

—----------------------------------------------------------------------------------------------------------------------------

## [**Examples**](https://docs.ansible.com/ansible/latest/collections/azure/azcollection/azure_rm_manageddisk_module.html#id6)

**-** **name:** Create managed disk

**azure\_rm\_manageddisk:**

**name:** mymanageddisk

**location:** eastus

**resource\_group:** myResourceGroup

**disk\_size\_gb:** 4

**-** **name:** Create managed operating system disk from page blob

**azure\_rm\_manageddisk:**

**name:** mymanageddisk

**location:** eastus2

**resource\_group:** myResourceGroup

**create\_option:** import

**source\_uri:** https://storageaccountname.blob.core.windows.net/containername/blob-name.vhd

**storage\_account\_id:** /subscriptions/<uuid>/resourceGroups/myResourceGroup/providers/Microsoft.Storage/storageAccounts/storageaccountname

**os\_type:** windows

**storage\_account\_type:** Premium\_LRS

**-** **name:** Mount the managed disk to VM

**azure\_rm\_manageddisk:**

**name:** mymanageddisk

**location:** eastus

**resource\_group:** myResourceGroup

**disk\_size\_gb:** 4

**managed\_by:** testvm001

**attach\_caching:** read\_only

**-** **name:** Mount the managed disk to multiple VMs

**azure\_rm\_manageddisk:**

**resource\_group:** myResourceGroup

**name:** freddisk04

**max\_shares:** 4

**disk\_size\_gb:** 1024

**storage\_account\_type:** Premium\_LRS

**managed\_by\_extended:**

**-** **resource\_group:** myResourceGroup01

**name:** testVM01

**-** **resource\_group:** myResourceGroup02

**name:** testVM02

**zone:** 1

**-** **name:** Unmount the managed disk to VM

**azure\_rm\_manageddisk:**

**name:** mymanageddisk

**location:** eastus

**resource\_group:** myResourceGroup

**managed\_by:** ''

**disk\_size\_gb:** 4

**-** **name:** Delete managed disk

**azure\_rm\_manageddisk:**

**name:** mymanageddisk

**location:** eastus

**resource\_group:** myResourceGroup

**state:** absent

—----------------------------------------------------------------------------------------------------------------------------

—----------------------------------------------------------------------------------------------------------------------------

## **Creating Virtual Machines**

There are two ways to create a virtual machine, both involving the azure\_rm\_virtualmachine module. We can either create a storage account, network interface, security group and public IP address and pass the names of these objects to the module as parameters, or we can let the module do the work for us and accept the defaults it chooses.

### **Creating Individual Components**

An Azure module is available to help you create a storage account, virtual network, subnet, network interface, security group and public IP. Here is a full example of creating each of these and passing the names to the azure\_rm\_virtualmachine module at the end:

# **Build Azure Instances using Ansible**

Subscribe To Me On YouTube: <https://bit.ly/lon_sub>

Links:

1. <https://azure.microsoft.com/en-gb/free/?ref=microsoft.com>
2. <https://docs.microsoft.com/en-us/azure/developer/ansible/install-on-linux-vm>
3. <https://docs.microsoft.com/en-us/cli/azure/install-azure-cli-linux?pivots=apt>
4. <https://docs.microsoft.com/en-us/azure/developer/ansible/create-ansible-service-principal>
5. <https://docs.ansible.com/ansible/latest/scenario_guides/guide_azure.html>

## **Actual commands:**

sudo apt-get update

sudo apt-get install ca-certificates curl apt-transport-https lsb-release gnupg -y

curl -sL https://packages.microsoft.com/keys/microsoft.asc | gpg --dearmor | sudo tee /etc/apt/trusted.gpg.d/microsoft.gpg REDIRECT /dev/null

\*\* Replace REDIRECT with chevron for greater than (YT doesn't like chevrons!) \*\*

AZ\_REPO=$(lsb\_release -cs)

echo "deb [arch=amd64] https://packages.microsoft.com/repos/azure-cli/ $AZ\_REPO main" | sudo tee /etc/apt/sources.list.d/azure-cli.list

sudo apt-get update

sudo apt-get install azure-cli

sudo apt install ansible

sudo apt install python3-pip

pip3 install ansible[azure] --user

az login

az ad sp create-for-rbac --name ansible

az role assignment create --assignee appID --role Contributor

az account show --query '{tenantId:tenantId,subscriptionid:id}';

az ad sp list --display-name ansible --query '{clientId:[0].appId}'

vi ~/.azure/credentials

ansible localhost -m azure\_rm\_resourcegroup -a "name=test1234 location=uksouth"

ansible localhost -m azure\_rm\_resourcegroup -a "name=test1234 location=uksouth state=absent"

mkdir azure

cd azure

vi azure\_vm.yml

ansible localhost -m azure\_rm\_resourcegroup -a "name=Testing location=uksouth"

ansible-playbook azure\_vm.yml

cat azure\_vm.yml

ssh youtubedemo@IP\_ADDRESS

## **The Playbook:**

**-** name**:** Create resource to build a VM

hosts**:** localhost

collections**:**

**-** azure.azcollection

tasks**:**

**-** name**:** Create storage account

azure\_rm\_storageaccount**:**

resource\_group**:** ARGOID-TEST2-RESOURCE-GROUP-1

name**:** storage\_account001

account\_type**:** Standard\_LRS

**-** name**:** Create virtual network

azure\_rm\_virtualnetwork**:**

resource\_group**:** ARGOID-TEST2-RESOURCE-GROUP-1

name**:** argoid-test2-vpc

address\_prefixes**:** "10.10.0.0/16"

**-** name**:** Add subnet

azure\_rm\_subnet**:**

resource\_group**:** ARGOID-TEST2-RESOURCE-GROUP-1

name**:** argoid-test2-subnet1

address\_prefix**:** "10.10.0.0/24"

virtual\_network**:** argoid-test2-vpc

**-** name**:** Create public ip

azure\_rm\_publicipaddress**:**

resource\_group**:** ARGOID-TEST2-RESOURCE-GROUP-1

allocation\_method**:** Static

name**:** 20.219.19.30

**-** name**:** Create security group that allows SSH

azure\_rm\_securitygroup**:**

resource\_group**:** ARGOID-TEST2-RESOURCE-GROUP-1

name**:** argoid-test2-nsg

rules**:**

**-** name**:** SSH

protocol**:** Tcp

destination\_port\_range**:** 22, 443

access**:** Allow

priority**:** 100, 120

direction**:** Inbound

**-** name**:** Create NIC

azure\_rm\_networkinterface**:**

resource\_group**:** ARGOID-TEST2-RESOURCE-GROUP-1

name**:** argoid-test2-host-0004VMNic

virtual\_network**:** argoid-test2-vpc

subnet**:** argoid-test2-subnet1

public\_ip\_name**:** 20.219.19.30

security\_group**:** argoid-test2-nsg

**-** name**:** Create virtual machine

azure\_rm\_virtualmachine**:**

resource\_group**:** ARGOID-TEST2-RESOURCE-GROUP-1

name**:** argoid-test2-host-0001

vm\_size**:** Standard\_E4as\_v4

storage\_account**:** storage\_account001

storage\_container**:** argoid-test2-host-0001

storage\_blob**:** testvm001.vhd

admin\_username**:** manjunath

admin\_password**:** Password123

network\_interfaces**:** argoid-test2-host-0004VMNic

image**:**

offer**:** CentOS

publisher**:** OpenLogic

sku**:** '7.6'

version**:** latest

[Build Azure Instances with Ansible Tutorial](https://www.youtube.com/watch?v=JZwHIytkyvI) - with service principal

[A beginners guide to Ansible and Microsoft Azure](https://youtu.be/BG2NtZWj3RI)

—----------------------------------------------------------------------------------------------------------------------------

—----------------------------------------------------------------------------------------------------------------------------

### 

### **Creating a Virtual Machine with Default Options**

If you simply want to create a virtual machine without specifying all the details, you can do that as well. The only caveat is that you will need a virtual network with one subnet already in your resource group. Assuming you have a virtual network already with an existing subnet, you can run the following to create a VM:

azure\_rm\_virtualmachine**:**

resource\_group**:** ARGOID-TEST2-RESOURCE-GROUP-1

name**:** argoid-test2-host-0001

vm\_size**:** Standard\_E4as\_v4

admin\_username**:** manjunath

ssh\_password\_enabled**:** false

ssh\_public\_keys**:** "{{ssh\_keys}}"

image**:**

offer**:** CentOS

publisher**:** OpenLogic

sku**:** '7.6'

version**:** latest

**COMMAND TO CREATE AZURE VM**

az vm create --name argoid-prod3-host-0031 --resource-group argoid-prod3-resource-group-1 --location centralindia --image "OpenLogic:CentOS:7\_6-gen2:latest" --size Standard\_E4as\_v4 --authentication-type ssh --admin-username manjunath --ssh-key-values manju.pub --storage-sku Standard\_LRS --os-disk-size-gb 30 --vnet-name argoid-prod3-vpc --subnet argoid-prod3-vpc-subnet-1 --nsg argoid-prod3-nsg --private-ip-address 10.1.0.31 --public-ip-address ""

—--------------------------------------------------------------------------------------------------------------------------------------------------------------

—----------------------------------------------------------------------------------------------------------------------------

## **Dynamic Inventory Script**

For a given host, the inventory script provides the following host variables:

**{**

"ansible\_host"**:** "XXX.XXX.XXX.XXX"**,**

"computer\_name"**:** "computer\_name2"**,**

"fqdn"**:** **null,**

"id"**:** "/subscriptions/subscription-id/resourceGroups/galaxy-production/providers/Microsoft.Compute/virtualMachines/object-name"**,**

"image"**:** **{**

"offer"**:** "CentOS"**,**

"publisher"**:** "OpenLogic"**,**

"sku"**:** "7.1"**,**

"version"**:** "latest"

**},**

"location"**:** "westus"**,**

"mac\_address"**:** "00-00-5E-00-53-FE"**,**

"name"**:** "object-name"**,**

"network\_interface"**:** "interface-name"**,**

"network\_interface\_id"**:** "/subscriptions/subscription-id/resourceGroups/galaxy-production/providers/Microsoft.Network/networkInterfaces/object-name1"**,**

"network\_security\_group"**:** **null,**

"network\_security\_group\_id"**:** **null,**

"os\_disk"**:** **{**

"name"**:** "object-name"**,**

"operating\_system\_type"**:** "Linux"

**},**

"plan"**:** **null,**

"powerstate"**:** "running"**,**

"private\_ip"**:** "172.26.3.6"**,**

"private\_ip\_alloc\_method"**:** "Static"**,**

"provisioning\_state"**:** "Succeeded"**,**

"public\_ip"**:** "XXX.XXX.XXX.XXX"**,**

"public\_ip\_alloc\_method"**:** "Static"**,**

"public\_ip\_id"**:** "/subscriptions/subscription-id/resourceGroups/galaxy-production/providers/Microsoft.Network/publicIPAddresses/object-name"**,**

"public\_ip\_name"**:** "object-name"**,**

"resource\_group"**:** "galaxy-production"**,**

"security\_group"**:** "object-name"**,**

"security\_group\_id"**:** "/subscriptions/subscription-id/resourceGroups/galaxy-production/providers/Microsoft.Network/networkSecurityGroups/object-name"**,**

"tags"**:** **{**

"db"**:** "mysql"

**},**

"type"**:** "Microsoft.Compute/virtualMachines"**,**

"virtual\_machine\_size"**:** "Standard\_DS4"

**}**

—----------------------------------------------------------------------------------------------------------------------------

## [**Examples**](https://docs.ansible.com/ansible/latest/collections/community/general/parted_module.html#id5)

**-** **name:** Create a new ext4 primary partition

**community.general.parted:**

**device:** /dev/sdb

**number:** 1

**state:** present

**fs\_type:** ext4

**-** **name:** Remove partition number 1

**community.general.parted:**

**device:** /dev/sdb

**number:** 1

**state:** absent

**-** **name:** Create a new primary partition with a size of 1GiB

**community.general.parted:**

**device:** /dev/sdb

**number:** 1

**state:** present

**part\_end:** 1GiB

**-** **name:** Create a new primary partition for LVM

**community.general.parted:**

**device:** /dev/sdb

**number:** 2

**flags:** **[** **lvm** **]**

**state:** present

**part\_start:** 1GiB

**-** **name:** Create a new primary partition with a size of 1GiB at disk's end

**community.general.parted:**

**device:** /dev/sdb

**number:** 3

**state:** present

**fs\_type:** ext4

**part\_start:** -1GiB

*# Example on how to read info and reuse it in subsequent task*

**-** **name:** Read device information (always use unit when probing)

**community.general.parted:** device=/dev/sdb unit=MiB

**register:** sdb\_info

**-** **name:** Remove all partitions from disk

**community.general.parted:**

**device:** /dev/sdb

**number:** '{{ **item.num** }}'

**state:** absent

**loop:** '{{ **sdb\_info.partitions** }}'

**-** **name:** Extend an existing partition to fill all available space

**community.general.parted:**

**device:** /dev/sdb

**number:** "{{ **sdb\_info.partitions** **|** **length** }}"

**part\_end:** "100%"

**resize:** true

**state:** present

—----------------------------------------------------------------------------------------------------------------------------

- 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: Public IP of VM

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: "<key\_data>"

network\_interfaces: myNIC

image:

offer: CentOS

publisher: OpenLogic

sku: '7.5'

version: latest