Create a Virtual Machine Running Linux

NOTE:

Azure has two different deployment models for creating and working with resources: [Resource Manager and classic](https://azure.microsoft.com/en-in/documentation/articles/resource-manager-deployment-model/). This article covers using the Resource Manager deployment model, which Microsoft recommends for most new deployments instead of the classic deployment model.

Creating an Azure virtual machine (VM) that runs Linux is easy to do from the command line or from the portal. This tutorial shows you how to use the Azure Command-Line Interface for Mac, Linux, and Windows (the Azure CLI) to create quickly an Ubuntu Server VM running in Azure, connect to it using **ssh**, and creating and mounting a new disk. (This topic uses an Ubuntu Server VM, but you can also create Linux VMs using [your own images as templates](https://azure.microsoft.com/en-in/documentation/articles/virtual-machines-linux-create-upload-vhd/).)

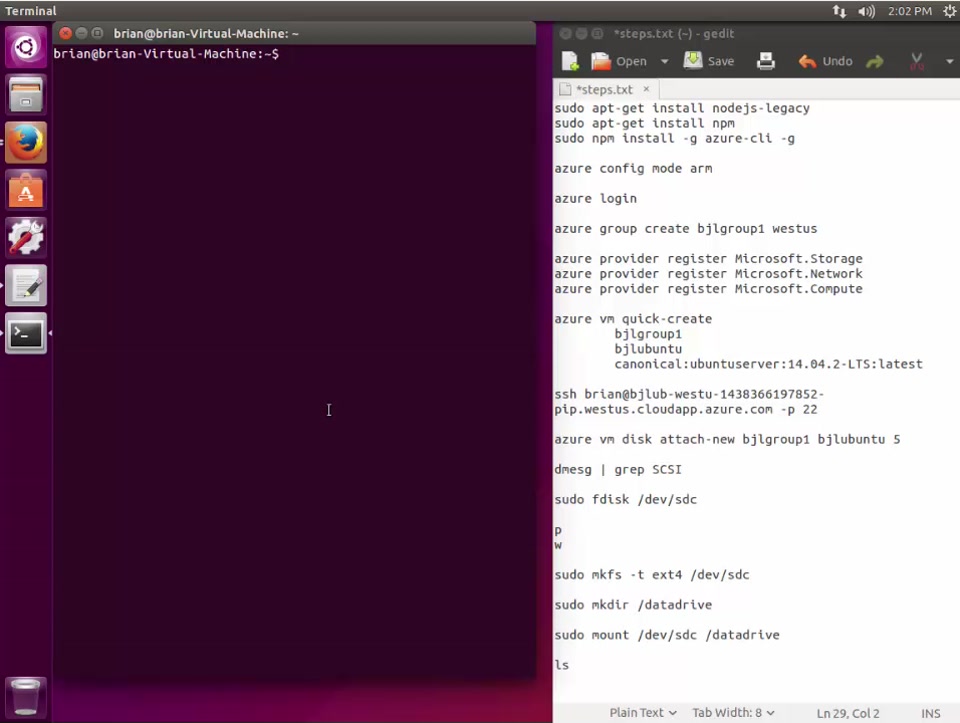
NOTE:

You need an Azure account to complete this tutorial:

* You can [open an Azure account for free](https://azure.microsoft.com/pricing/free-trial/?WT.mc_id=A261C142F): You get credits you can use to try out paid Azure services, and even after they're used up you can keep the account and use free Azure services, such as Websites. Your credit card will never be charged, unless you explicitly change your settings and ask to be charged.
* You can [activate MSDN subscriber benefits](https://azure.microsoft.com/pricing/member-offers/msdn-benefits-details/?WT.mc_id=A261C142F): Your MSDN subscription gives you credits every month that you can use for paid Azure services.

Video walkthrough

Here's a walkthrough of this tutorial.

[[](https://azure.microsoft.com/en-us/documentation/videos/building-a-linux-virtual-machine-tutorial/)08-04-2015 10 min, 34 sec](https://azure.microsoft.com/en-us/documentation/videos/building-a-linux-virtual-machine-tutorial/)

Install the Azure CLI

The first step is to [install the Azure CLI](https://azure.microsoft.com/en-in/documentation/articles/xplat-cli-install/).

Good. Now make sure you're in the Resource Manager mode by typing azure config mode arm.

Even better. Now [log in with your work or school id](https://azure.microsoft.com/en-in/documentation/articles/xplat-cli-connect/#use-the-log-in-method) by typing azure login and following the prompts for an interactive login experience to your Azure account.

NOTE:

If you have a work or school ID and you know you do not have two-factor authentication enabled, you can use azure login -u along with the work or school ID to log in without an interactive session. If you don't have a work or school ID, you can [create a work or school id from your personal Microsoft account](https://azure.microsoft.com/en-in/documentation/articles/resource-group-create-work-id-from-personal/).

Create your Azure VM

Type azure group create <my-group-name> westus replacing *<my-group-name>* with a group name that's unique to you (you can use a different region if you want). You should see something like the following:

Copy to clipboardCopy

azure group create myuniquegroupname westus

info: Executing command group create

+ Getting resource group myuniquegroupname

+ Creating resource group myuniquegroupname

info: Created resource group myuniquegroupname

data: Id: /subscriptions/xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx/resourceGroups/myuniquegroupname

data: Name: myuniquegroupname

data: Location: westus

data: Provisioning State: Succeeded

data: Tags:

data:

info: group create command OK

Now create your VM by typing azure vm quick-create, and you'll receive prompts to input the remaining parameters. Use the name of the resource group that you just created, above, and for the **ImageURN** value, use canonical:ubuntuserver:14.04.2-LTS:latest, so that your experience looks something like the following. Note that the azure vm quick-create command prompts for basic information it requires to create, host, and connect to a Linux VM, including:

* the resource group name and VM name
* a deployment location
* the operating system type and the image URN string
* a username and password

and then creates the infrastructure necessary to host the VM. This includes:

* An Azure storage account for VHD storage and extra disks
* A NIC for the VM
* a vnet with a subnet
* a public IP address
* a subdomain

azure vm quick-create info: Executing command vm quick-create Resource group name: myuniquegroupname Virtual machine name: myuniquevmname Location name: westus Operating system Type [Windows, Linux]: Linux ImageURN (format: "publisherName:offer:skus:version"): canonical:ubuntuserver:14.04.2-LTS:latest User name: ops Password: \*\*\*\*\*\*\*\*\* Confirm password: \*\*\*\*\*\*\*\*\*

* + Looking up the VM "myuniquevmname" info: Using the VM Size "Standard\_D1" info: The [OS, Data] Disk or image configuration requires storage account
  + Retrieving storage accounts info: Could not find any storage accounts in the region "westus", trying to create new one
  + Creating storage account "cli3c0464f24f1bf4f014323" in "westus"
  + Looking up the storage account cli3c0464f24f1bf4f014323
  + Looking up the NIC "myuni-westu-1432328437727-nic" info: An nic with given name "myuni-westu-1432328437727-nic" not found, creating a new one
  + Looking up the virtual network "myuni-westu-1432328437727-vnet" info: Preparing to create new virtual network and subnet / Creating a new virtual network "myuni-westu-1432328437727-vnet" [address prefix: "10.0.0.0/16"] with subnet "myuni-westu-1432328437727-snet"+[address prefix: "10.0.1.0/24"]
  + Looking up the virtual network "myuni-westu-1432328437727-vnet"
  + Looking up the subnet "myuni-westu-1432328437727-snet" under the virtual network "myuni-westu-1432328437727-vnet" info: Found public ip parameters, trying to setup PublicIP profile
  + Looking up the public ip "myuni-westu-1432328437727-pip" info: PublicIP with given name "myuni-westu-1432328437727-pip" not found, creating a new one
  + Creating public ip "myuni-westu-1432328437727-pip"
  + Looking up the public ip "myuni-westu-1432328437727-pip"
  + Creating NIC "myuni-westu-1432328437727-nic"
  + Looking up the NIC "myuni-westu-1432328437727-nic"
  + Creating VM "myuniquevmname"
  + Looking up the VM "myuniquevmname"
  + Looking up the NIC "myuni-westu-1432328437727-nic"
  + Looking up the public ip "myuni-westu-1432328437727-pip" data: Id :/subscriptions/xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx/resourceGroups/myuniquegroupname/providers/Microsoft.Compute/virtualMachines/myuniquevmname data: ProvisioningState :Succeeded data: Name :myuniquevmname data: Location :westus data: FQDN :myuni-westu-1432328437727-pip.westus.cloudapp.azure.com data: Type :Microsoft.Compute/virtualMachines data: data: Hardware Profile: data: Size :Standard\_D1 data: data: Storage Profile: data: Image reference: data: Publisher :canonical data: Offer :ubuntuserver data: Sku :14.04.2-LTS data: Version :latest data: data: OS Disk: data: OSType :Linux data: Name :cli3c0464f24f1bf4f0-os-1432328438224 data: Caching :ReadWrite data: CreateOption :FromImage data: Vhd: data: Uri :https://cli3c0464f24f1bf4f014323.blob.core.windows.net/vhds/cli3c0464f24f1bf4f0-os-1432328438224.vhd data: data: OS Profile: data: Computer Name :myuniquevmname data: User Name :ops data: Linux Configuration: data: Disable Password Auth :false data: data: Network Profile: data: Network Interfaces: data: Network Interface #1: data: Id :/subscriptions/xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx/resourceGroups/myuniquegroupname/providers/Microsoft.Network/networkInterfaces/myuni-westu-1432328437727-nic data: Primary :true data: MAC Address :00-0D-3A-31-55-31 data: Provisioning State :Succeeded data: Name :myuni-westu-1432328437727-nic data: Location :westus data: Private IP alloc-method :Dynamic data: Private IP address :10.0.1.4 data: Public IP address :191.239.51.1 data: FQDN :myuni-westu-1432328437727-pip.westus.cloudapp.azure.com info: vm quick-create command OK

Your VM is up and running and waiting for you to connect.

Connecting to your VM

With Linux VMs, you typically connect using **ssh**.

NOTE:

This topic connects to a VM using usernames and passwords; to use public and private key pairs to communicate with your VM, see [How to Use SSH with Linux on Azure](https://azure.microsoft.com/en-in/documentation/articles/virtual-machines-linux-use-ssh-key/). You can modify the **SSH** connectivity of VMs created with the azure vm quick-create command by using the azure vm reset-access command to reset **SSH** access completely, add or remove users, or add public key files to secure access. This article uses username and password with **SSH** for brevity.

If you're not familiar with connecting with **ssh**, the command takes the form ssh <username>@<publicdnsaddress> -p <the ssh port>. In this case, we use the username and password from the previous step and port 22, which is the default **ssh** port.

Copy to clipboardCopy

ssh ops@myuni-westu-1432328437727-pip.westus.cloudapp.azure.com -p 22

The authenticity of host 'myuni-westu-1432328437727-pip.westus.cloudapp.azure.com (191.239.51.1)' can't be established.

ECDSA key fingerprint is bx:xx:xx:xx:xx:xx:xx:xx:xx:x:x:x:x:x:x:xx.

Are you sure you want to continue connecting (yes/no)? yes

Warning: Permanently added 'myuni-westu-1432328437727-pip.westus.cloudapp.azure.com,191.239.51.1' (ECDSA) to the list of known hosts.

ops@myuni-westu-1432328437727-pip.westus.cloudapp.azure.com's password:

Welcome to Ubuntu 14.04.2 LTS (GNU/Linux 3.16.0-37-generic x86\_64)

\* Documentation: https://help.ubuntu.com/

System information as of Fri May 22 21:02:32 UTC 2015

System load: 0.37 Memory usage: 2% Processes: 207

Usage of /: 41.4% of 1.94GB Swap usage: 0% Users logged in: 0

Graph this data and manage this system at:

https://landscape.canonical.com/

Get cloud support with Ubuntu Advantage Cloud Guest:

http://www.ubuntu.com/business/services/cloud

0 packages can be updated.

0 updates are security updates.

The programs included with the Ubuntu system are free software;

the exact distribution terms for each program are described in the

individual files in /usr/share/doc/\*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by

applicable law.

ops@myuniquevmname:~$

Now that you're connected to your VM, you're ready to attach a disk.

Attach and mount a disk

Attaching a new disk is quick. Just type azure vm disk attach-new <myuniquegroupname> <myuniquevmname> <size-in-GB> to create and attach a new GB disk for your VM. It should look something like this:

Copy to clipboardCopy

azure vm disk attach-new myuniquegroupname myuniquevmname 5

info: Executing command vm disk attach-new

+ Looking up the VM "myuniquevmname"

info: New data disk location: https://cliexxx.blob.core.windows.net/vhds/myuniquevmname-20150526-0xxxxxxx43.vhd

+ Updating VM "myuniquevmname"

info: vm disk attach-new command OK

Now let's find the disk, using dmesg | grep SCSI (the method you use to discover your new disk may vary). In this case, it looks something like:

Copy to clipboardCopy

dmesg | grep SCSI

[ 0.294784] SCSI subsystem initialized

[ 0.573458] Block layer SCSI generic (bsg) driver version 0.4 loaded (major 252)

[ 7.110271] sd 2:0:0:0: [sda] Attached SCSI disk

[ 8.079653] sd 3:0:1:0: [sdb] Attached SCSI disk

[ 1828.162306] sd 5:0:0:0: [sdc] Attached SCSI disk

and in the case of this topic, the sdc disk is the one that we want. Now partition the disk with sudo fdisk /dev/sdc -- assuming that in your case the disk was sdc, and make it a primary disk on partition 1, and accept the other defaults.

Copy to clipboardCopy

sudo fdisk /dev/sdc

Device contains neither a valid DOS partition table, nor Sun, SGI or OSF disklabel

Building a new DOS disklabel with disk identifier 0x2a59b123.

Changes will remain in memory only, until you decide to write them.

After that, of course, the previous content won't be recoverable.

Warning: invalid flag 0x0000 of partition table 4 will be corrected by w(rite)

Command (m for help): n

Partition type:

p primary (0 primary, 0 extended, 4 free)

e extended

Select (default p): p

Partition number (1-4, default 1): 1

First sector (2048-10485759, default 2048):

Using default value 2048

Last sector, +sectors or +size{K,M,G} (2048-10485759, default 10485759):

Using default value 10485759

Create the partition by typing p at the prompt:

Copy to clipboardCopy

Command (m for help): p

Disk /dev/sdc: 5368 MB, 5368709120 bytes

255 heads, 63 sectors/track, 652 cylinders, total 10485760 sectors

Units = sectors of 1 \* 512 = 512 bytes

Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk identifier: 0x2a59b123

Device Boot Start End Blocks Id System

/dev/sdc1 2048 10485759 5241856 83 Linux

Command (m for help): w

The partition table has been altered!

Calling ioctl() to re-read partition table.

Syncing disks.

And write a file system to the partition by using the **mkfs** command, specifying your filesystem type and the device name. In this topic, we're using ext4 and /dev/sdc1 from above:

Copy to clipboardCopy

sudo mkfs -t ext4 /dev/sdc1

mke2fs 1.42.9 (4-Feb-2014)

Discarding device blocks: done

Filesystem label=

OS type: Linux

Block size=4096 (log=2)

Fragment size=4096 (log=2)

Stride=0 blocks, Stripe width=0 blocks

327680 inodes, 1310464 blocks

65523 blocks (5.00%) reserved for the super user

First data block=0

Maximum filesystem blocks=1342177280

40 block groups

32768 blocks per group, 32768 fragments per group

8192 inodes per group

Superblock backups stored on blocks:

32768, 98304, 163840, 229376, 294912, 819200, 884736

Allocating group tables: done

Writing inode tables: done

Creating journal (32768 blocks): done

Writing superblocks and filesystem accounting information: done

Now we create a directory to mount the file system using mkdir:

Copy to clipboardCopy

sudo mkdir /datadrive

And you mount the directory using mount:

Copy to clipboardCopy

sudo mount /dev/sdc1 /datadrive

The data disk is now ready to use as /datadrive.

Copy to clipboardCopy

ls

bin datadrive etc initrd.img lib64 media opt root sbin sys usr vmlinuz

boot dev home lib lost+found mnt proc run srv tmp var

NOTE:

You can also connect to your Linux virtual machine using an SSH key for identification. For details, see [How to Use SSH with Linux on Azure](https://azure.microsoft.com/en-in/documentation/articles/virtual-machines-linux-use-ssh-key/).

Next Steps

Remember, that your new disk will not typically be available to the VM if it reboots unless you write that information to your [fstab](http://en.wikipedia.org/wiki/Fstab) file. If you want, you can add several more disks and [configure RAID](https://azure.microsoft.com/en-in/documentation/articles/virtual-machines-linux-configure-raid/).

To learn more about Linux on Azure, see:

* [Linux and Open-Source Computing on Azure](https://azure.microsoft.com/en-in/documentation/articles/virtual-machines-linux-opensource/)
* [How to use the Azure Command-Line Interface](https://azure.microsoft.com/en-in/documentation/articles/virtual-machines-command-line-tools/)
* [Deploy a LAMP app using the Azure CustomScript Extension for Linux](https://azure.microsoft.com/en-in/documentation/articles/virtual-machines-linux-script-lamp/)
* [The Docker Virtual Machine Extension for Linux on Azure](https://azure.microsoft.com/en-in/documentation/articles/virtual-machines-docker-vm-extension/)