**Virtual Machine:**

1. **Image:** Deployed onto VM, the OS
2. **Network**
3. **Block storage:** store VM and app data

* VMs are IaaS (full access to OS)
* Durable stateful storage

**VM types:**

* General purpose
* Compute optimized
* Memory optimized
* Storage optimized
* Graphics Processing Unit (GPU)
* High performance compute

**Azure Cloud Services**

* Web Roles and Worker Roles
* **Stateless** VM
* PaaS
* Limited operating systems
* Temporary local storage

**Azure Functions (BUILD ON TOP OF WEBJOBS)**

* Serverless compute
* Code as service
* Billing based on execution
* Various endpoints, triggers (timer, event, file being uploaded, etc)

Azure Web Jobs

* Backend processing
* Limited access to OS
* Limited local storage options

When creating virtual network for VM, specify a subnet

* by default, Azure will create a virtual network, network interface, and public IP for your VM.

**Create from Portal:**

1. **Virtual Network:** VNet enables many types of Azure resources, such as Azure Virtual Machines (VM), to securely communicate with each other, the internet, and on-premises networks.  
   1. Address Space:  
      A custom private IP address space using public and private (RFC 1918) addresses.  
      Azure assigns resources in a virtual network a private IP address from the address space that you assign.  
      For example, if you deploy a VM in a VNet with address space, 10.0.0.0/16, the VM will be assigned a private IP like 10.0.0.4.
   2. Subnet:  
      Enables us to segment the virtual network into one or more sub-networks and allocate a portion of the virtual network's address space to each subnet.  
      We can deploy Azure resources in a specific subnet.  
      Like in a traditional network, subnets allow us to segment VNet address space into segments that are appropriate for the organization's internal network.  
      This also improves address allocation efficiency. Securing resources: Network Security Groups.
      1. Name
      2. Address Range
2. **Virtual Machine**
   1. Managed storage
   2. Make sure to select inbound ports (specify RDP in case of Windows VM, SSH for Linux, but even stuff like HTTP, HTTPS, MS SQL are available)
   3. Post config stuff:
      1. Backups: Whole VM snapshots
      2. Managed Service Identity: Configure VM with Azure Active Directory
      3. Boot diagnostics: Logs, screenshots taken from VM (placed into Storage Account)

Note: When creating Linux VM, we can use an SSH public key, instead of passwords for authentication.

**VM sizing:**

**Storage options:** HDD or SSD. Choose Standard SSD disks if you have normal workloads but want better performance. Choose Premium SSD disks if you have I/O intensive workloads or mission-critical systems that need to process data very quickly.

Azure uses virtual hard disks (VHDs) to represent physical disks for the VM.  
VHDs replicate the logical format and data of a disk drive but are stored as page blobs in an Azure Storage account.  
You can choose on a per disk basis what type of storage it should use (SSD or HDD).

For a Linux VM, two VHDs are created:

* **One for the OS:** primary disk, max. 2048GB. labeled as /dev/sda by default.
* **One temporary:** Stores swap files, its size is based on the VM’s size. Labeled as /dev/sdb and is formatted and mounted to /mnt by the Azure Linux Agent. **Not persistent, do not write to this important stuff.**

Where to store data? Either on VHD for OS, or better: reate dedicated data disks. Data Disk can be up to 4095GBs. Can be created from a real disk, thus easing migrations.

1. Managed disks
   1. This is the modern, recommended approach
   2. I specify the disk type (Premium or Standard) and the size of the disk, and Azure creates and manages both the disk *and* the storage it uses
   3. **Increased reliability:** Azure ensures that VHDs associated with high-reliability VMs will be placed in different parts of Azure Storage to provide similar levels of resilience.
   4. **Better security:** Real managed resources in the resource group: means they can use role-based access control to restrict who can work with the VHD data.
   5. **Snapshot support:** Create a read-only copy of a VHD. You have to shut down the owning VM, but creating the snapshot only takes a few seconds. Once it's done, you can power on the VM and use the snapshot to create a duplicate VM to troubleshoot a production issue or roll back the VM to the point in time that the snapshot was taken.
   6. **Backup support:** Managed disks can be automatically backed up to different regions for disaster recovery with Azure Backup without affecting the service of the VM.
2. Unmanaged disks
   1. I am responsible for the storage accounts that are used to hold the VHDs that correspond to my VM disks
   2. pay the storage account rates for the amount of space I use
   3. A single storage account has a fixed rate limit of 20,000 I/O operations/sec. This means that a single storage account is capable of supporting 40 standard virtual hard disks at full throttle. If you need to scale out, then you need more than one storage account, which can get complicated.

**VNet**

Virtual machines communicate with external resources using a virtual network (VNet).

* A VNet represents a private network in a single region that your resources communicate on.
* Virtual Networks can be divided up with subnets to isolate resources, connect them to other networks (including on-premises networks), and apply traffic rules to govern inbound and outbound connections.

In a production environment where we already have other components, you'd want to utilize an existing virtual network. That way, your VM can communicate with the other cloud services in your solution. If there isn't one defined in this location yet, you can create it here and configure

Remote connections:

There are two approaches we can use to authenticate an SSH connection: **username and password**, or an **SSH key pair**.

Although SSH provides an encrypted connection, using passwords with SSH connections leaves the VM vulnerable to brute-force attacks of passwords. A more secure and preferred method of connecting to a Linux VM with SSH is a public-private key pair, also known as SSH keys.

With an SSH key pair, you can sign in to Linux-based Azure virtual machines without a password. This is a more secure approach if you only plan to sign in to the VM from a few computers. If you need to be able to access the Linux VM from a variety of locations, a username and password combination might be a better approach.

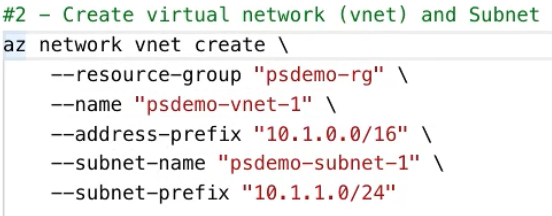
1. The public key is placed on your Linux VM or any other service that you wish to use with public-key cryptography. This can be shared with anyone.
2. The **private key** is what you present to verify your identity to your Linux VM when you make an SSH connection. Consider this confidential information and protect this like you would a password or any other private data.
   1. Private key passphrase: You can optionally provide a passphrase while generating your private key. This is a password you must enter when you use the key. This passphrase is used to access the private SSH key file and is not the user account password. When you add a passphrase to your SSH key, it encrypts the private key using 128-bit AES so that the private key is useless without the passphrase to decrypt it.

use the built-in ssh-keygen command to generate the SSH public and private key files.

# Create Linux VM with AzureCLI

(also works with –-n and –-l shorthands)

(az group list -o table)



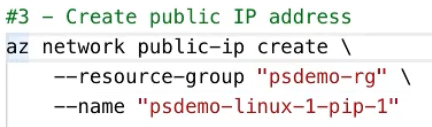
**useful later on:**

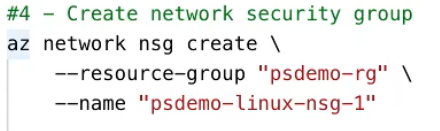
*az network vnet check-ip-address*

Check if a private IP address is available for use within a virtual network.

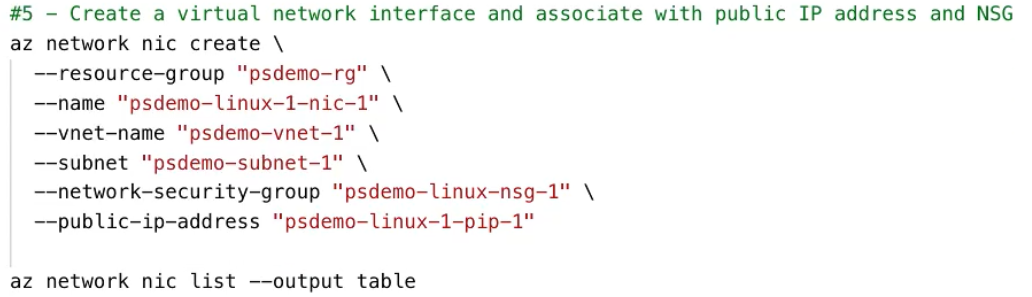
*az network vnet list (--output table)*

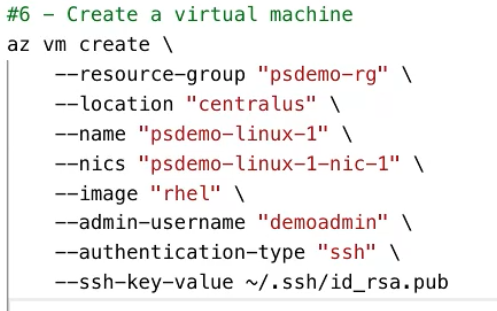
List virtual networks.



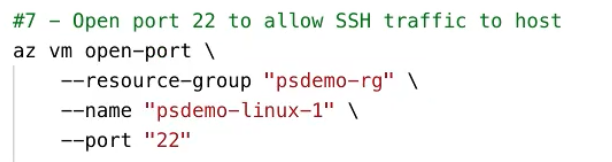


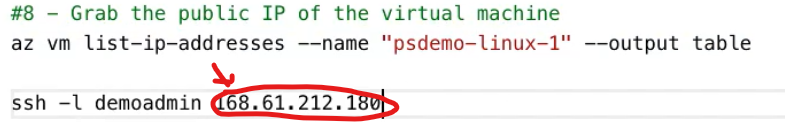
(az network nsg list)



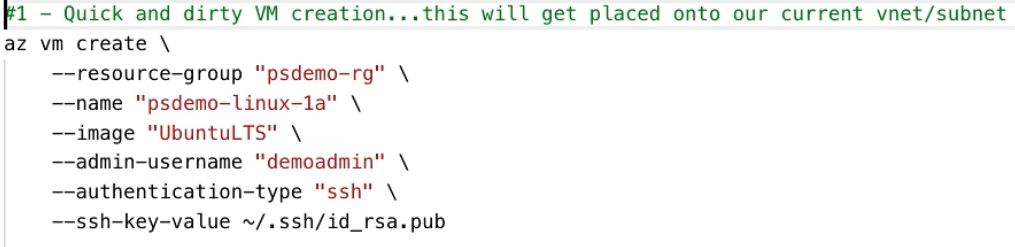


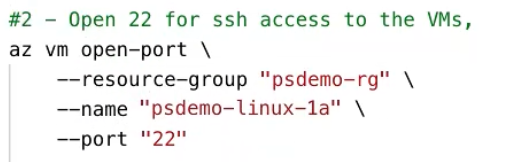
If things get a bit confusing: 

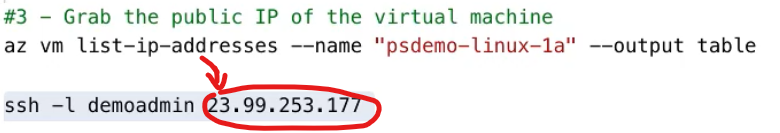




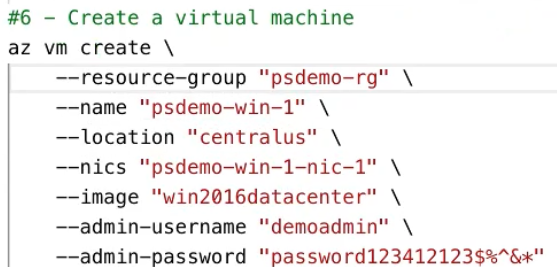
# Create Linux VM with AzureCLI less explicitly



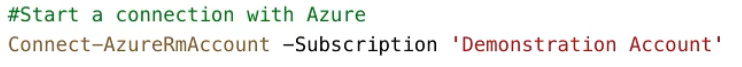




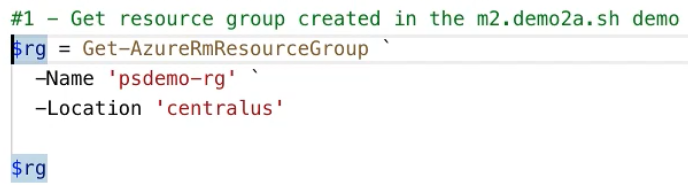
If trying to create a Windows VM:

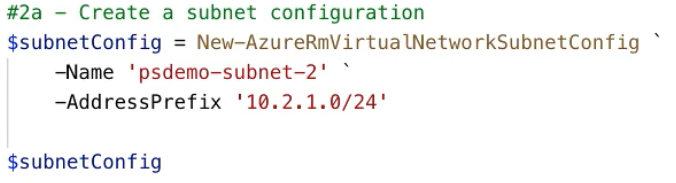


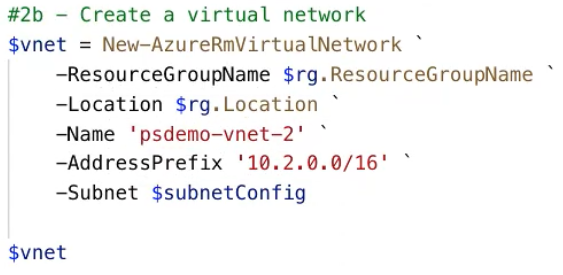
# Create Linux VM with PowerShell

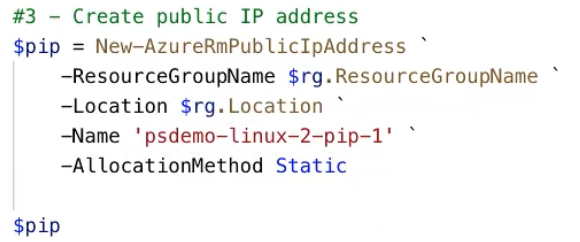


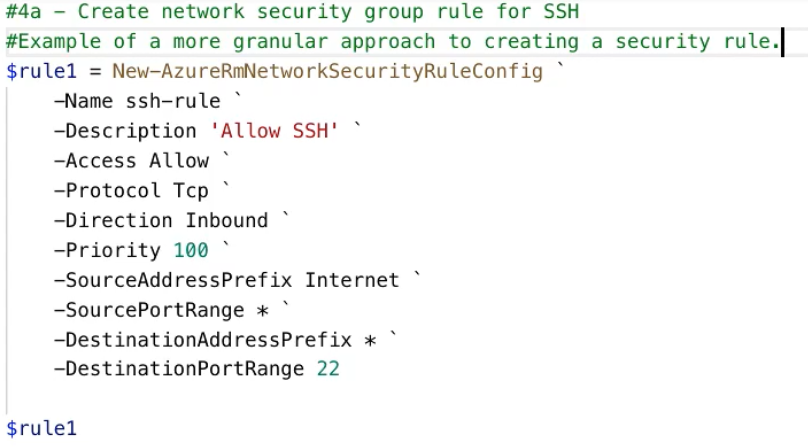






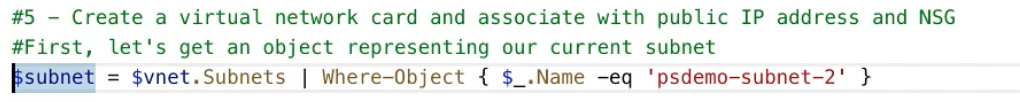




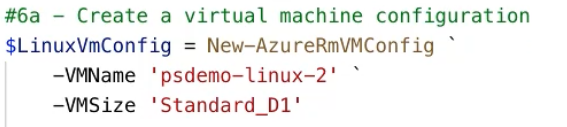


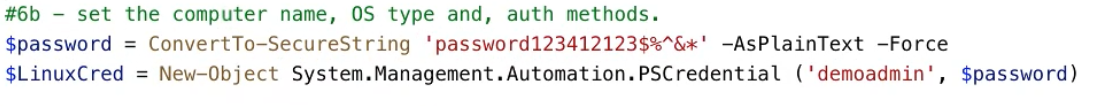
Previously we used the open port command: not with this.  
Priority: 100th (so if there’s a rule with Priority 101, it will get executed 1 later)

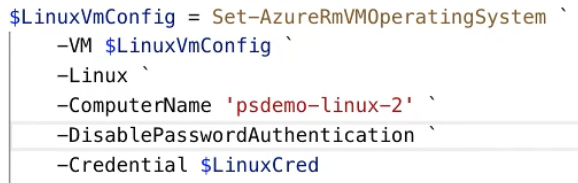


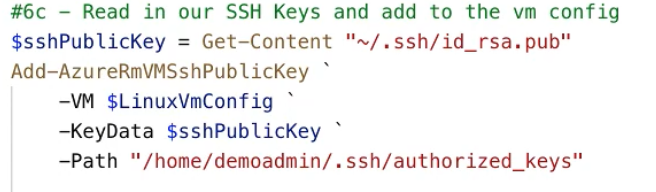


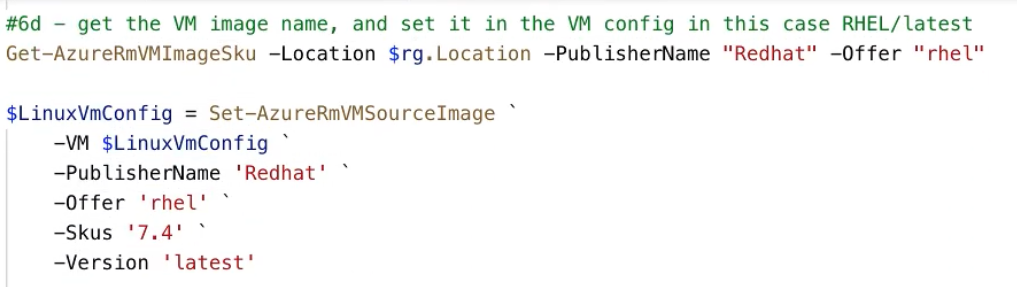


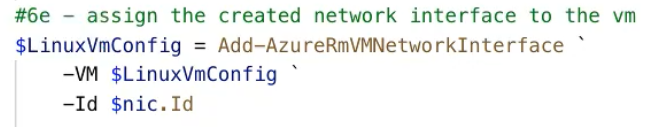


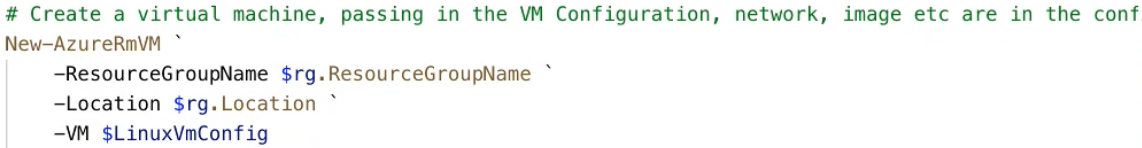


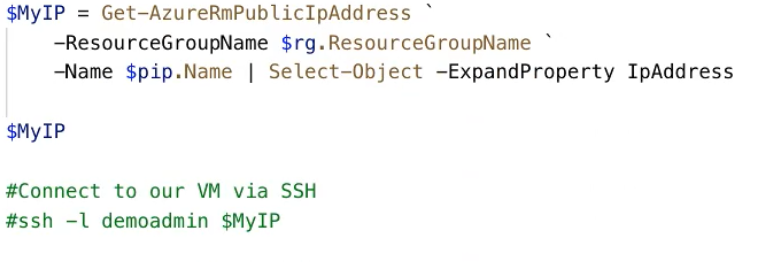












# Virtual Machine states

* Starting
* Running
* Stopping
* Stopped (still being charged, because it is allocated)
* Deallocating
* Deallocated

-AsJob parameter: PowerShell job, which is unblocking in the console

-Force: refresh state (some things are cached in the CLI/PowerShell/CloudShell)

**Get-AzureRmVm**

**Stop-AzureRmVm** –Force –AsJob *–StayProvidisioned*

...if StayProvisioned parameter is given, then VM is only stopped, NOT deallocated

**Start-AzureRmVm**

**Remove-AzureRmVm** (deletes VM all together, gone for good)