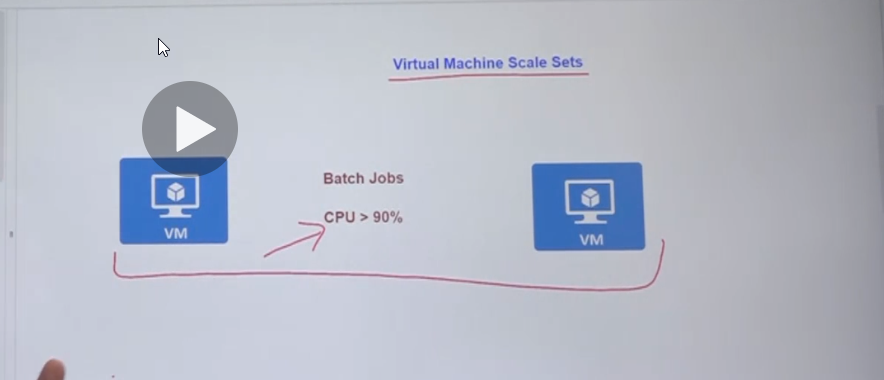
**Azure virtual machine scale sets**

A scale sets is a n identical pool of VMs running some application. You have application running some batch job within the VM , more and more jobs accumulated within the application and CPU capacity going beyond > 90%. To solve the issue, upgrade the size, ram and so on but there is some limitation due to application perspective. The company normally does to deploy another VM and distributed the batch jobs within the both VMs so the load of the jobs balanced between the VMs**./ this is manual process need to set alarm or metric in vms where administrator will get alarm if goes above 90 and spping the batch jobs to next vm/--------------------Company wats everything automated for that azure provide scale sets to solve the manual problem.**



1.Scale set allows to create and manage a group of identical VMs

2.you can also set the scale sets behind the load balancer to distribute the traffic across the VMs

3.vm instance automatically increase and decrease on demand.

4.VMs scale sets helps provide better redundancy and improve performance for your applications

Virtual Machine scale sets advantages:

1.Automatically Scale Up on demand - if goes above 75% vm will spine up and share the jobs.

2.Automatically scale down on demand – if goes below 25% will remove the vm coz there is cost.

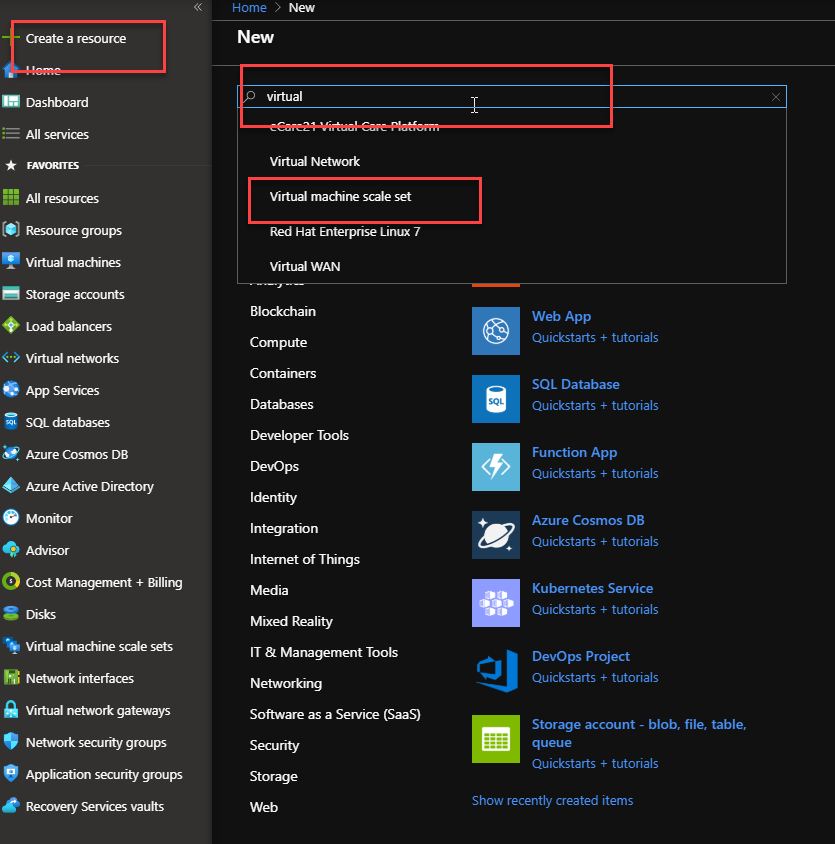
3.Define the configuration of the VM that world be part of scale set.

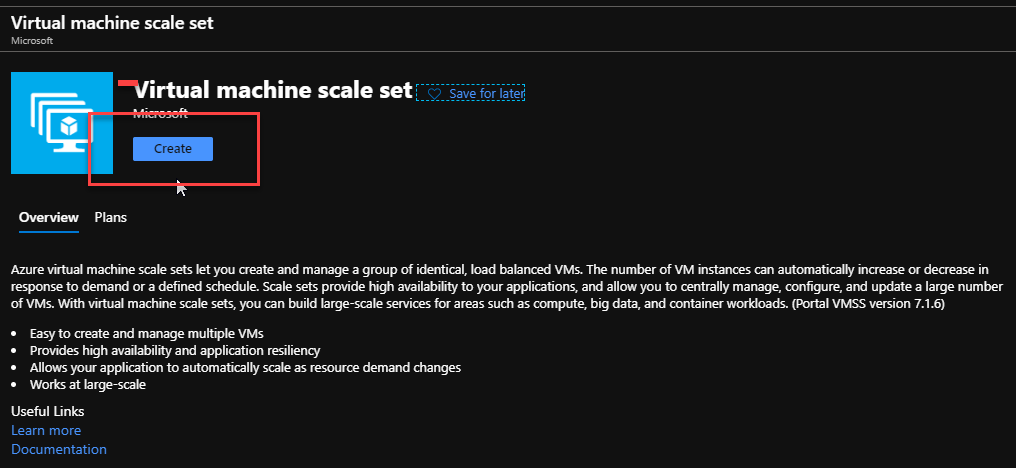
4.define the scaling conditions.

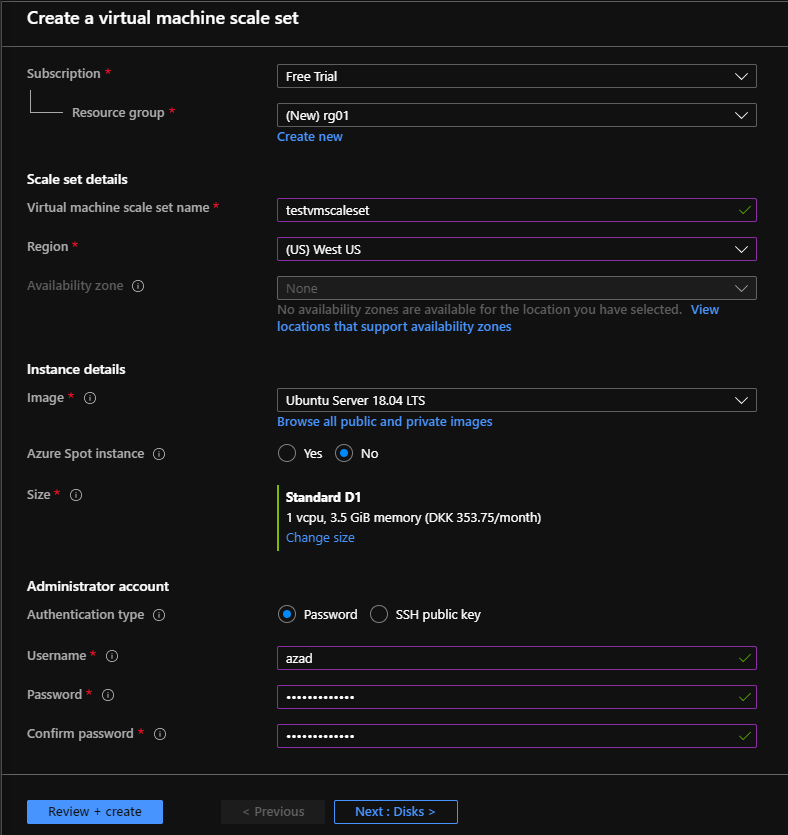
**(vertical scaling-scaleup and down -horizontal scaling-scale in and out)**

**(scale set support 1000VM instance, if you create your own custom VM images-the limit is 600vms)**

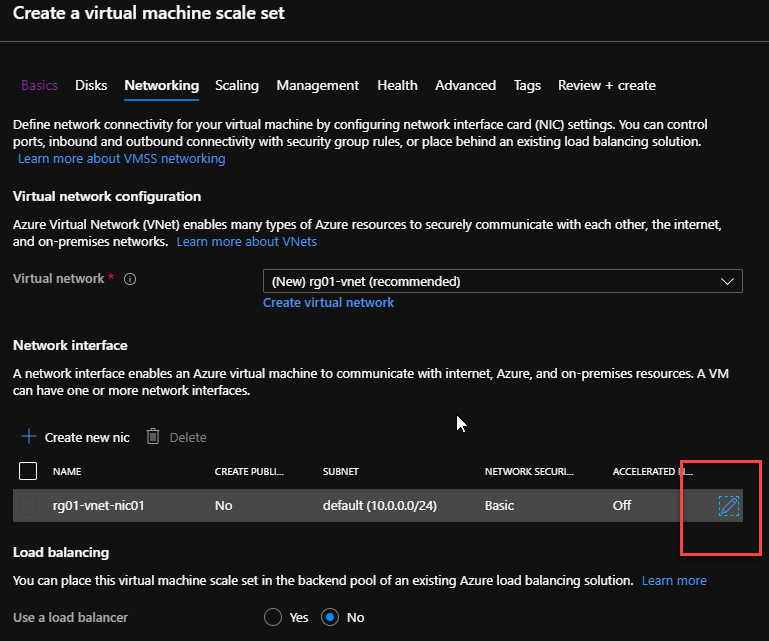
**1.**

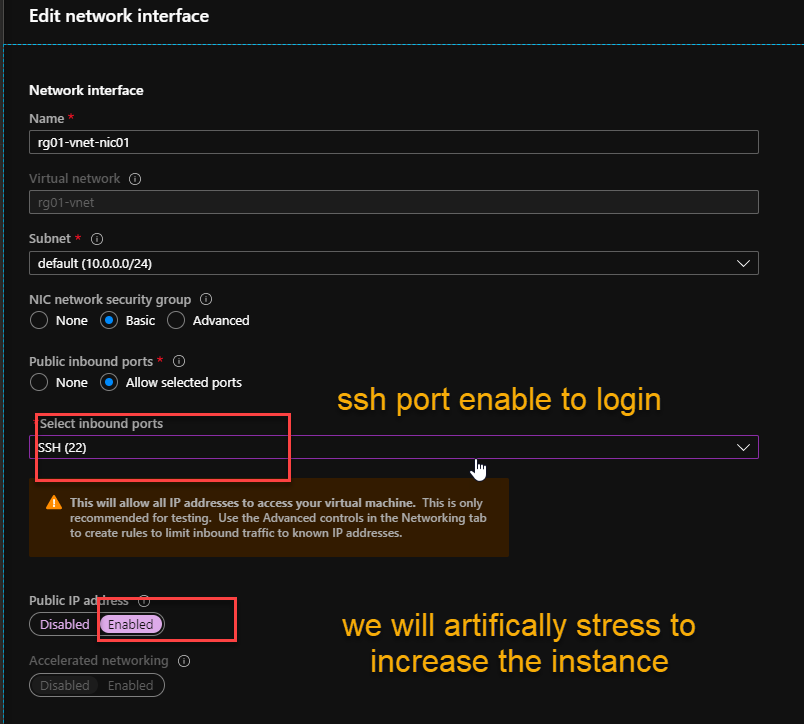


**2.** 

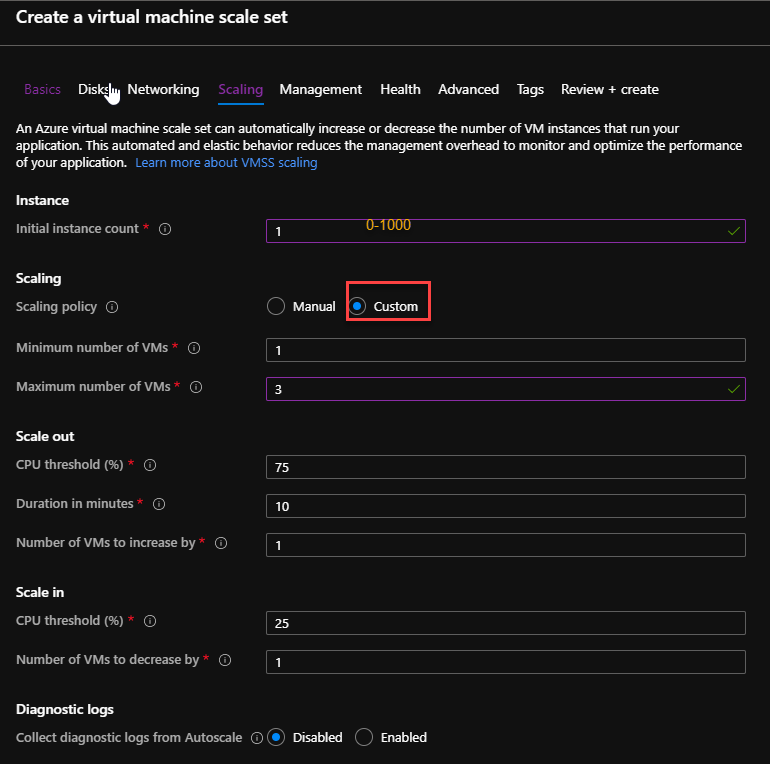
3. 

4.

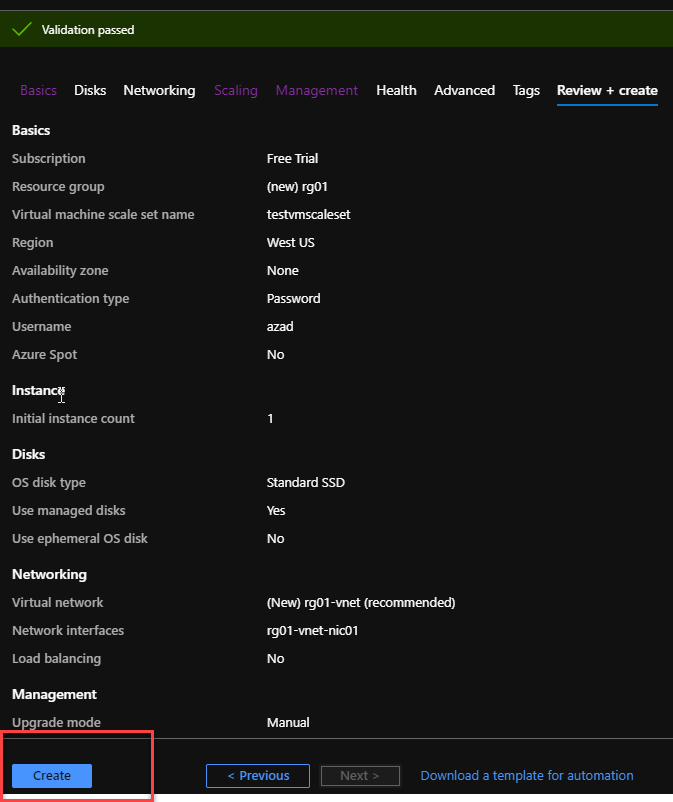


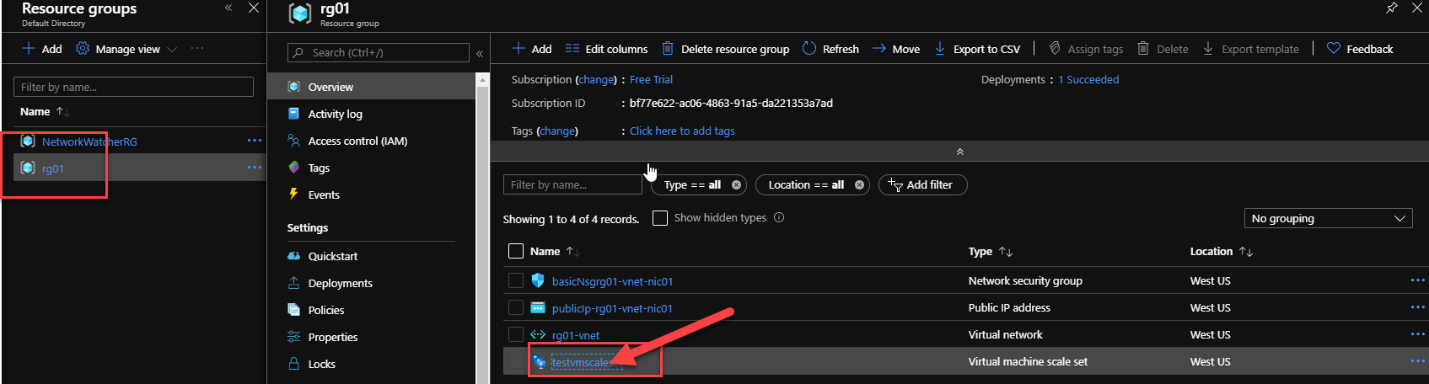


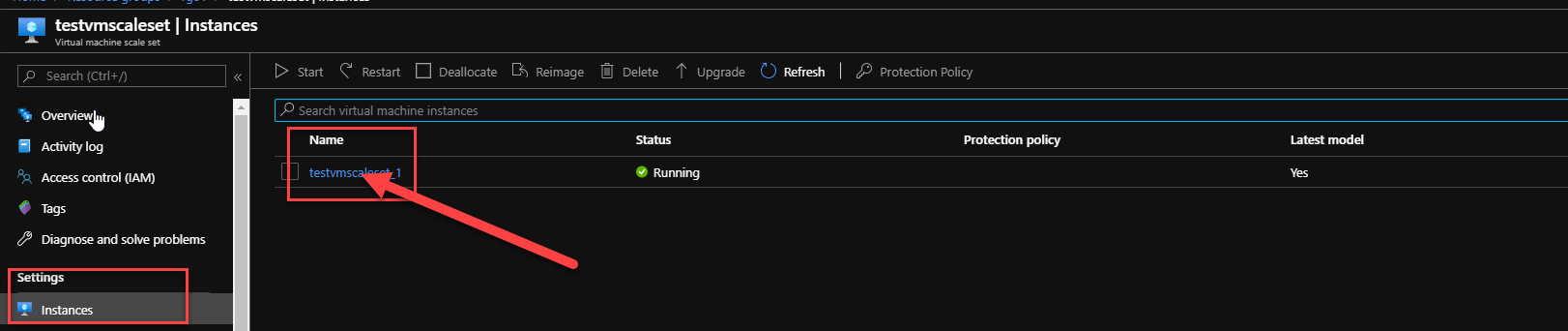
**5.**

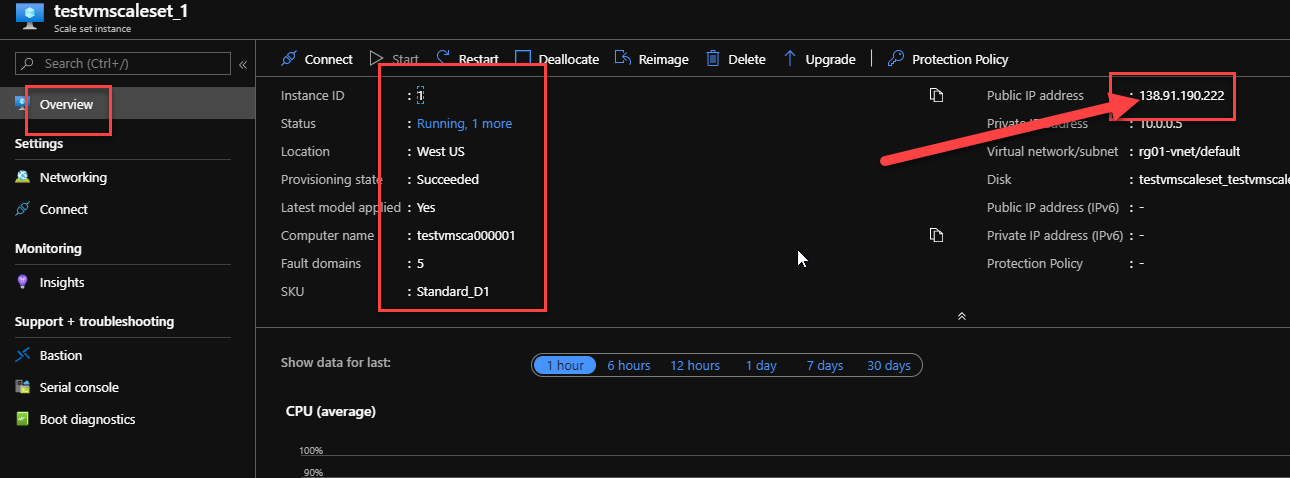


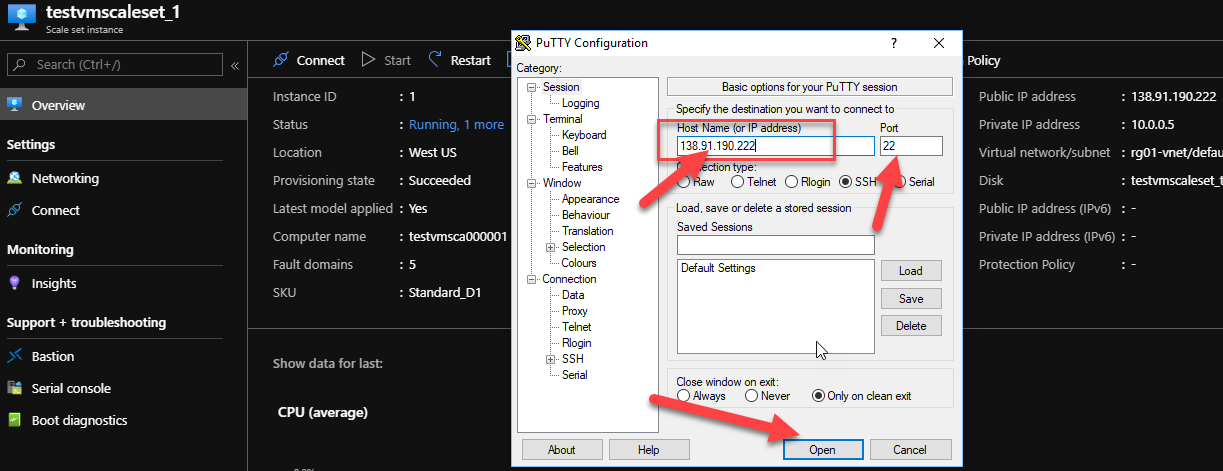
**6.**

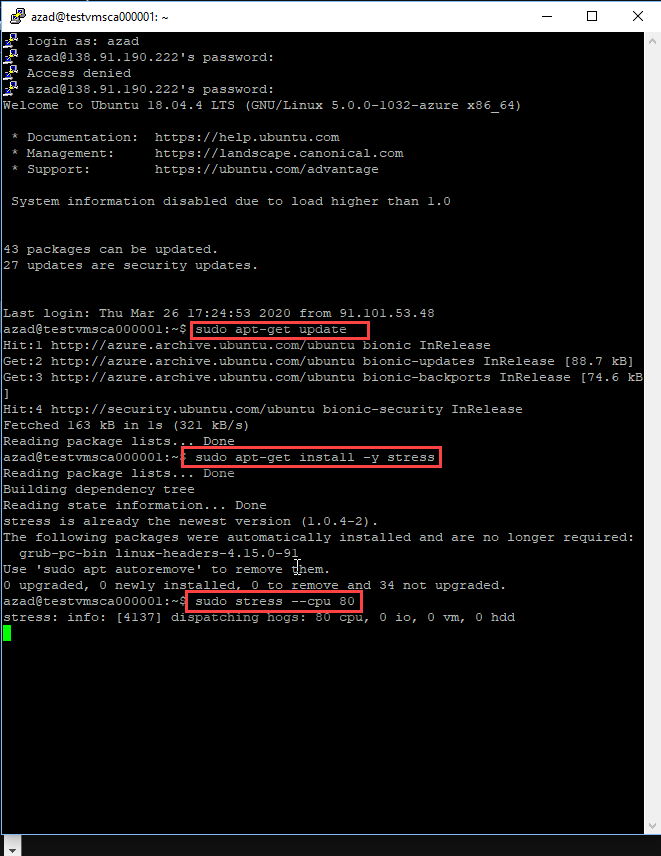












**Lab - Azure virtual machine scale sets - Command Line - Practice commands**

**// Creating a new virtual machine scale set**

az vmss create -n appvmset -g azuredemo --instance-count 1 --image Win2016Datacenter --data-disk-sizes-gb 10 --vnet-name azuredemo-vnet --subnet default --public-ip-per-vm --admin-username demousr

**// Using custom script extensions for a scale set**

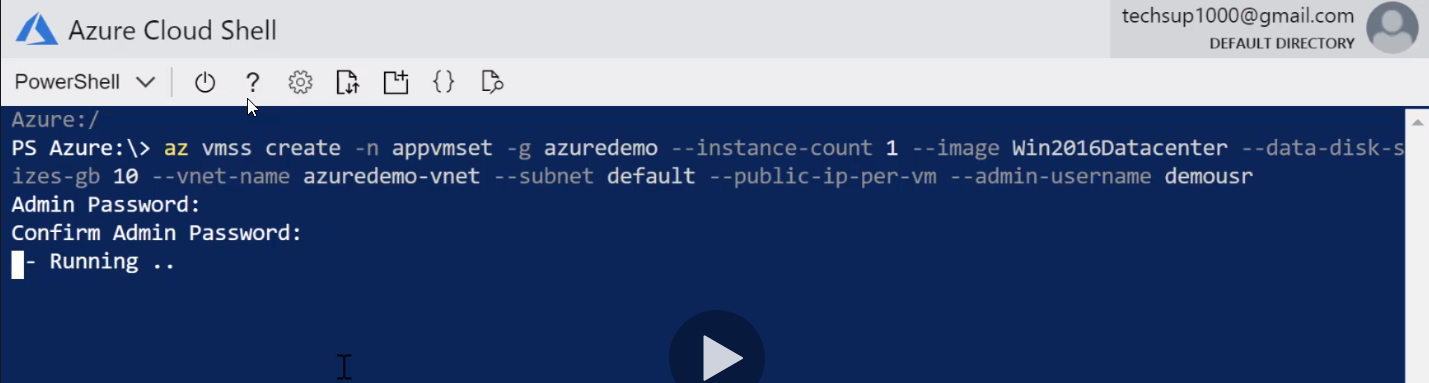
az vmss extension set --publisher "Microsoft.Compute" --version 1.10 --resource-group azuredemo --vmss-name appvmset --settings "appconfig.json" --name CustomScriptExtension

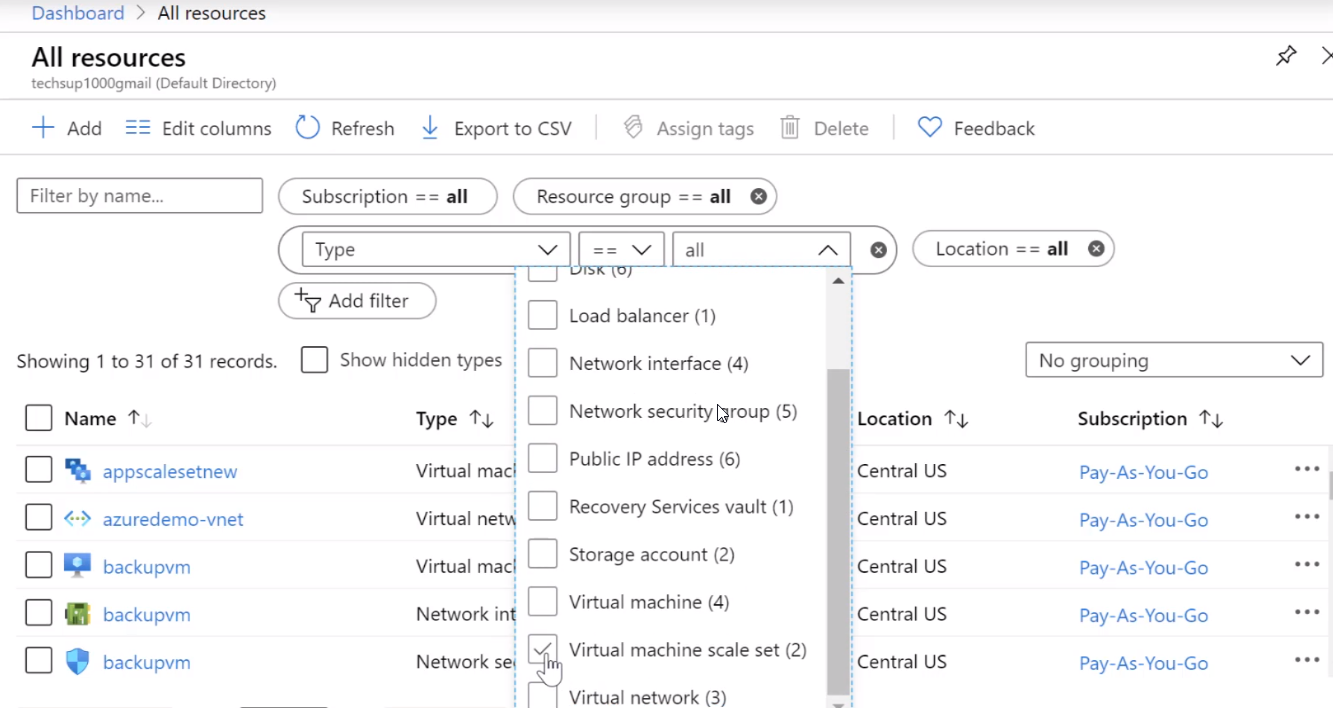
**This chapter has the configuration file attached as a resource. Please ensure you have a GitHub account and attach the URL for your account.**

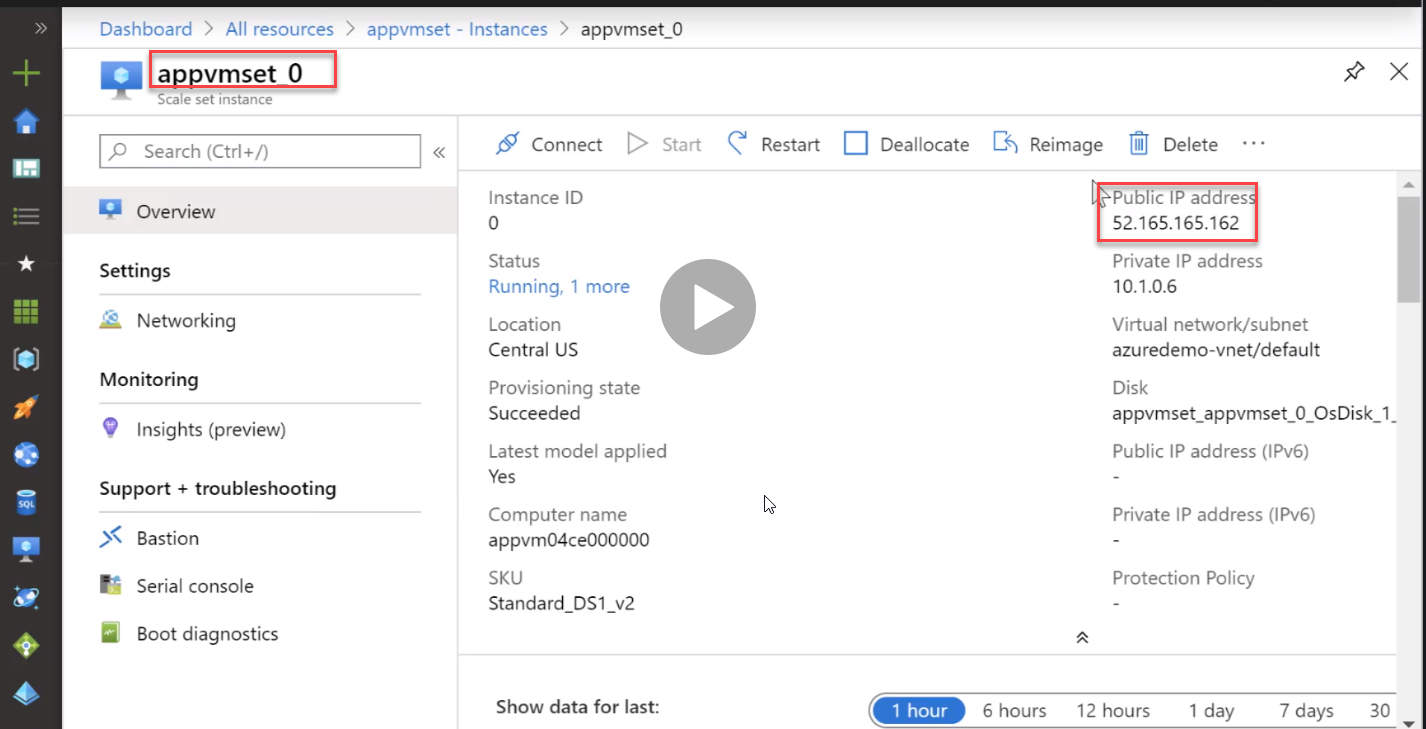
In the GitHub account , create a repository and add a file named InstallIIS.ps1 that has the following contents

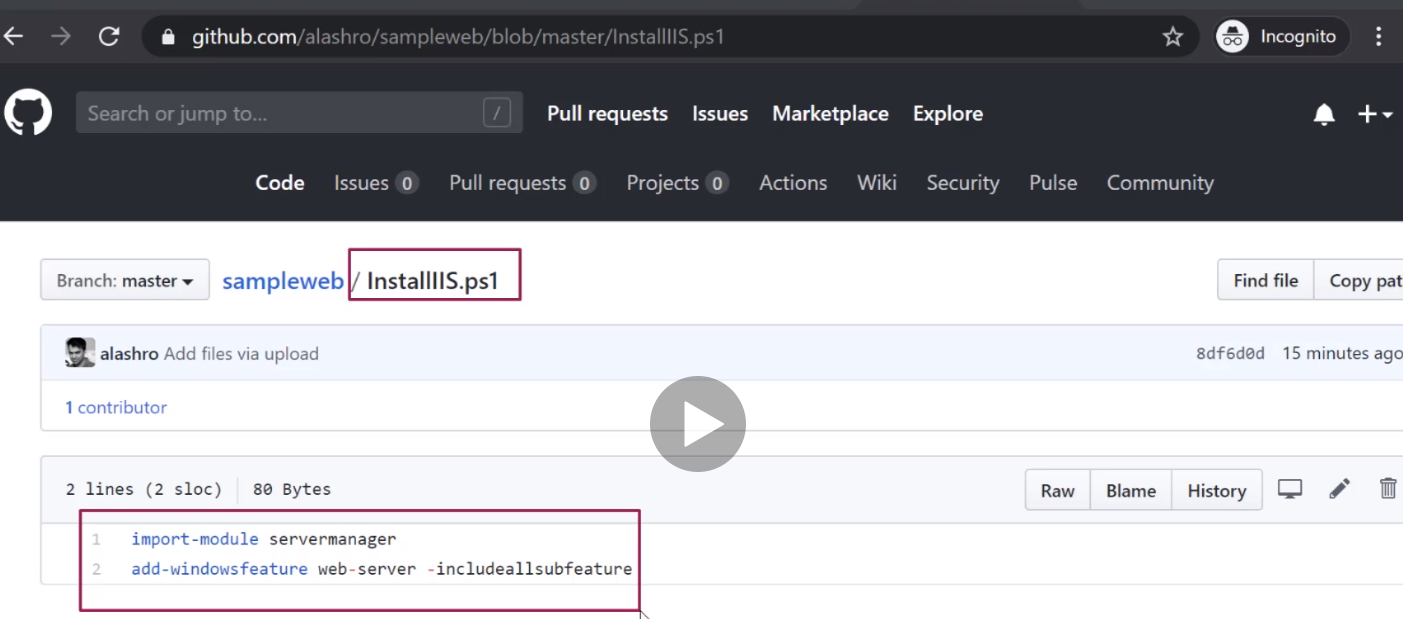
import-module servermanager

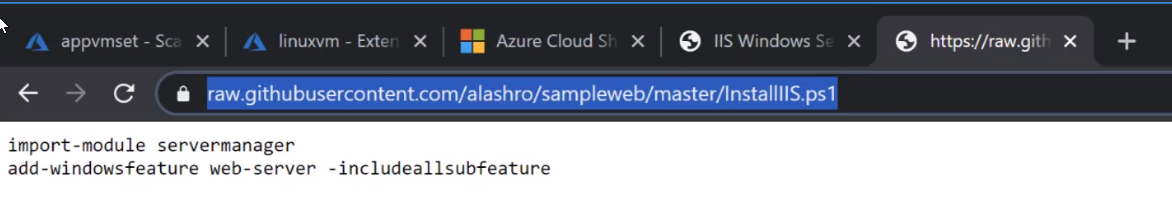
add-windowsfeature web-server -includeallsubfeature

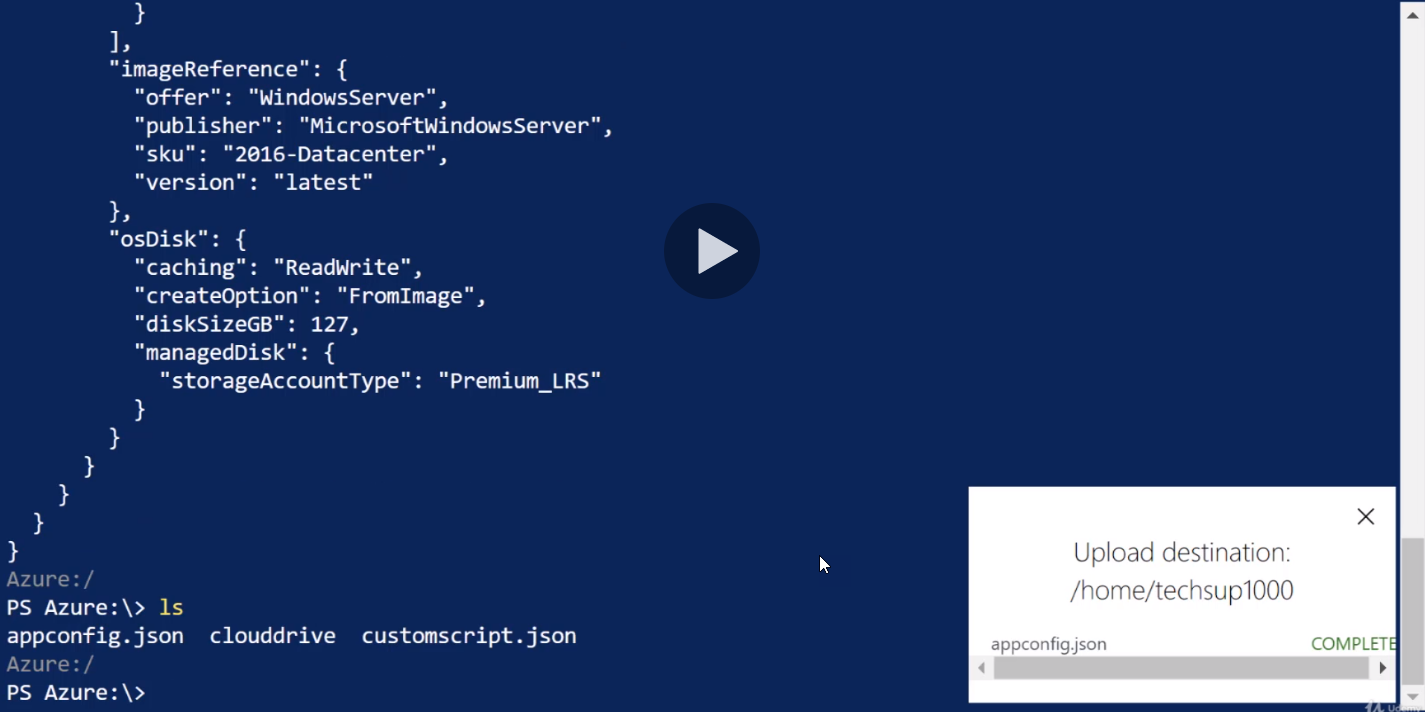


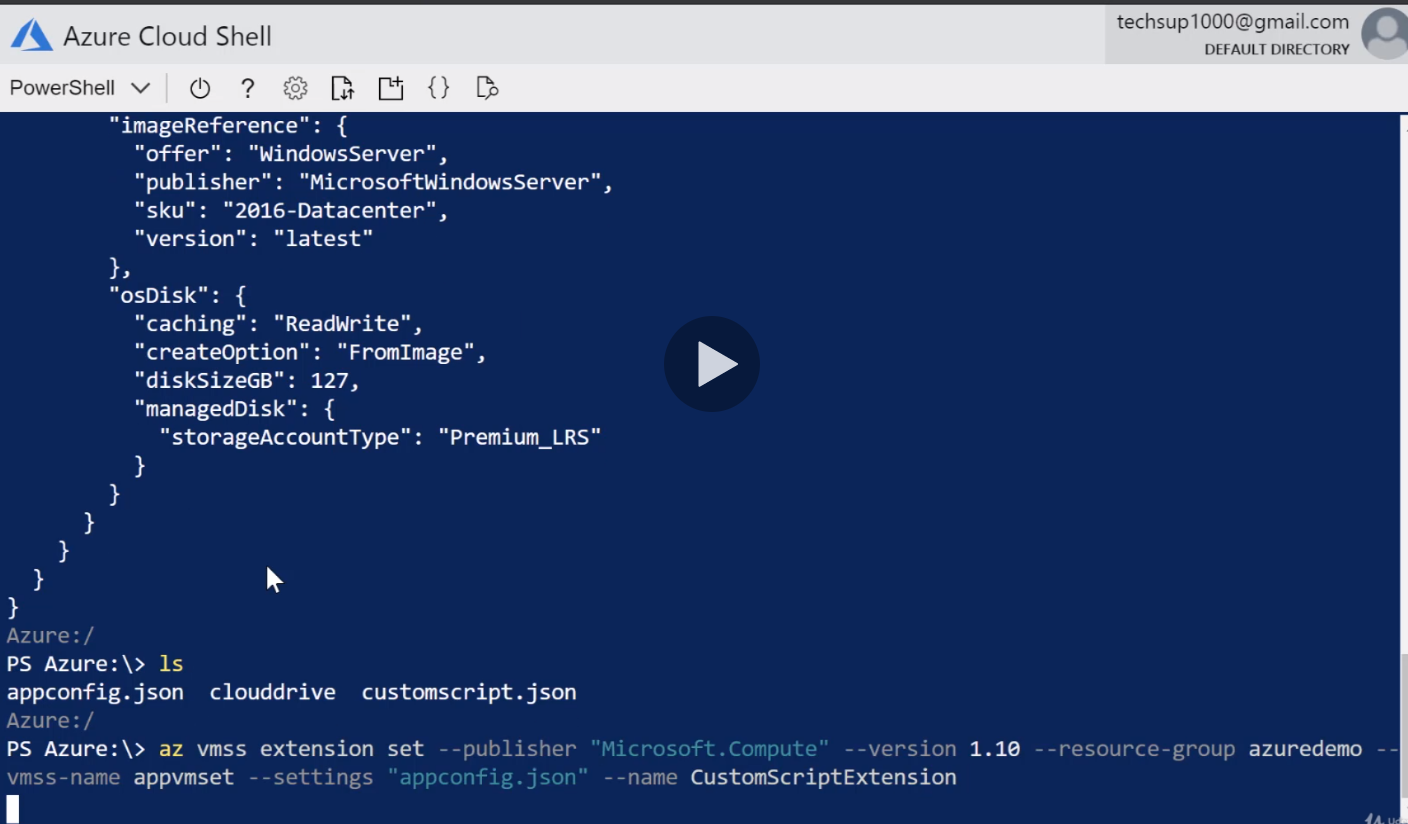


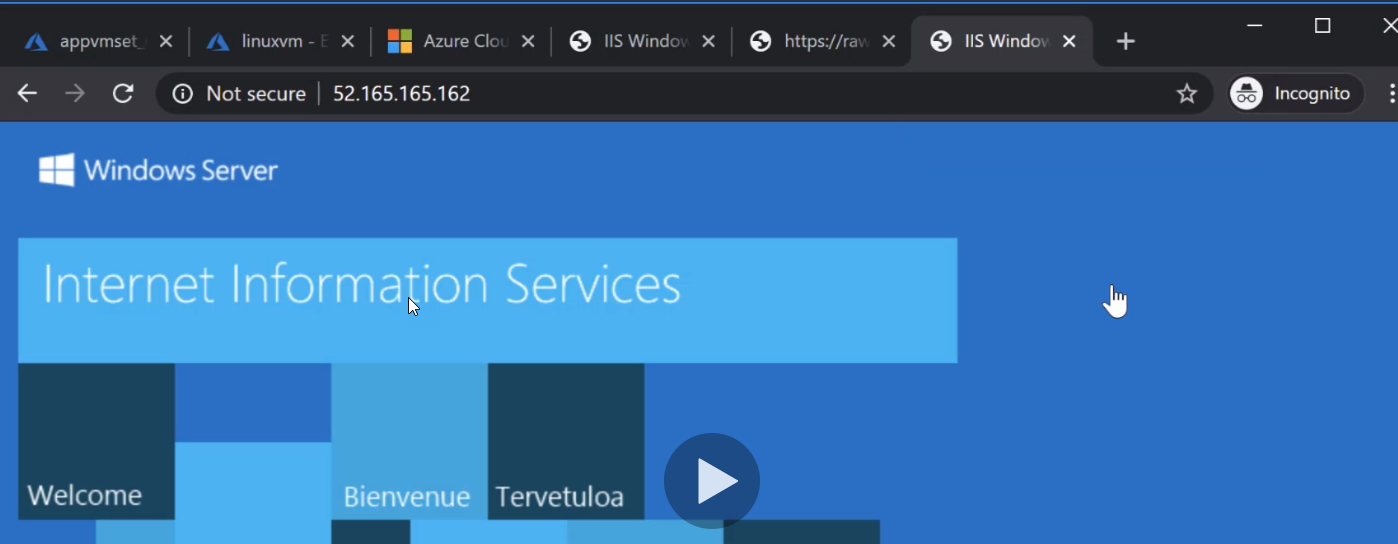












**Scaling conditions:**

After configured the all the threshold in scale set vms , we still can re-configure the rules and all the threshold according to you r instances.

Azure gave lots of flexibility in their portal to change the instances and thresholds and all metric values.

