# Prerequisite

You will need to have finished and still have your session18RG Resource Group

# Session’s Network Diagram

This is a network diagram for session 19



Outline of session

1. Create Load Balancer
2. Create Virtual Machine Scale Set with 1 VM
3. Create auto scaling rules
4. Simulate 1,000s of request - via Apache Benchmark
5. Watch automatic scale up and down of number of instances from 1->10 VMs

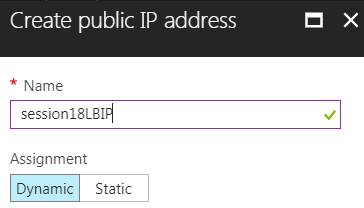
# Create Load Balancer

In this section you will create a Load Balancer and

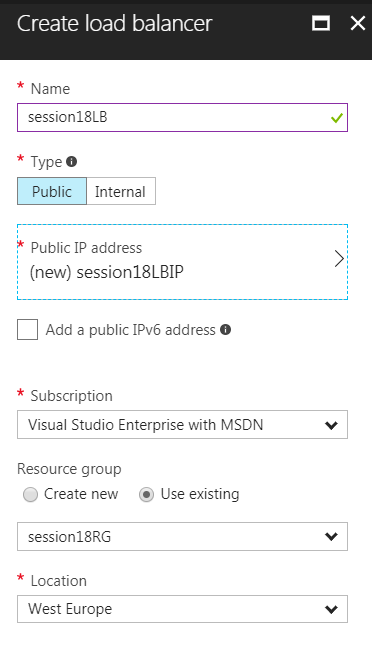
1. Assign a public address to it (assigned to Frontend IP of VMSS)
2. Assign LB to the Backend pools
3. Create its health probes
4. Create load balancing rule
5. Login to portal using

https://portal.azure.com

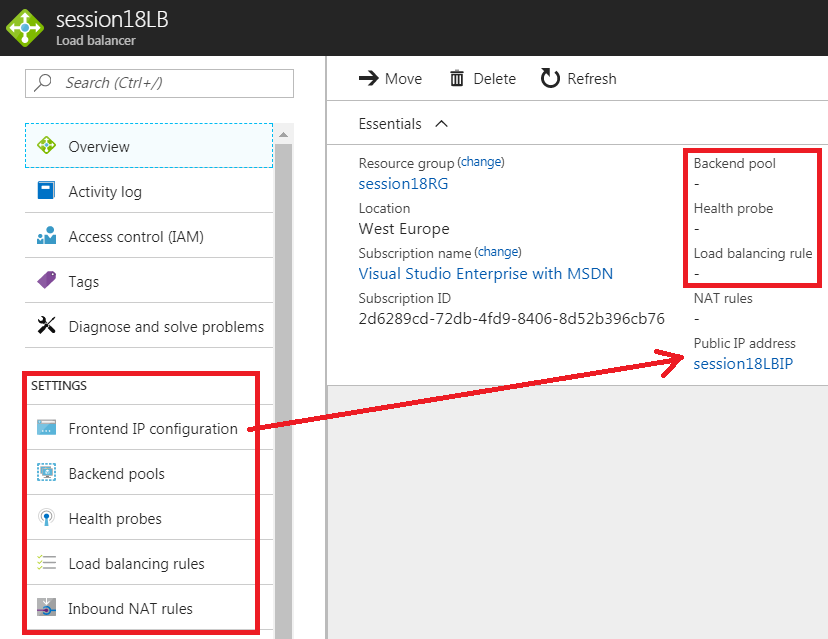
1. In favourites menu select ‘Load balancers’ then ‘Add’
2. In new blade select ‘*Choose a public IP address*’.
3. Select ‘Create new’ and name it session18LBIP then OK



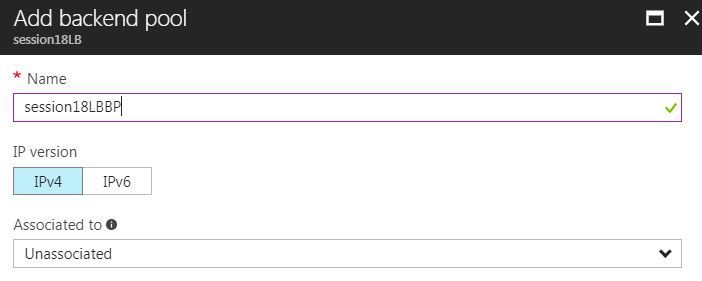
1. Complete the ‘Create load balancer’ configuration details as shown below then ‘OK’



1. Select the newly created load balancer



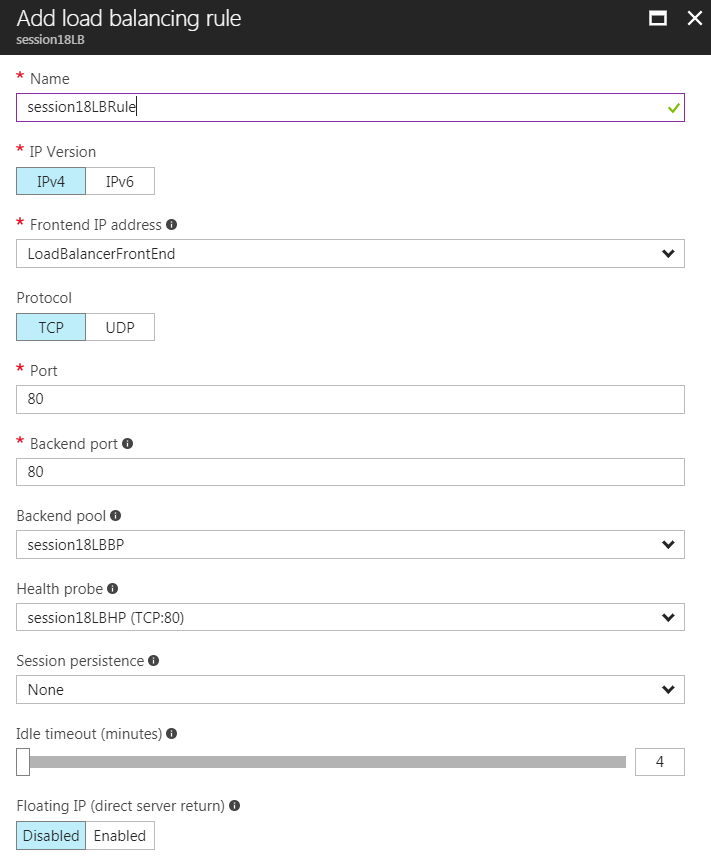
1. Select ‘Backend pools’ then ‘Add’ and name the pool and leave it ‘Unassociated’ then OK (when VMSS created LB’s backend pool will be automatically associated with VMSS)



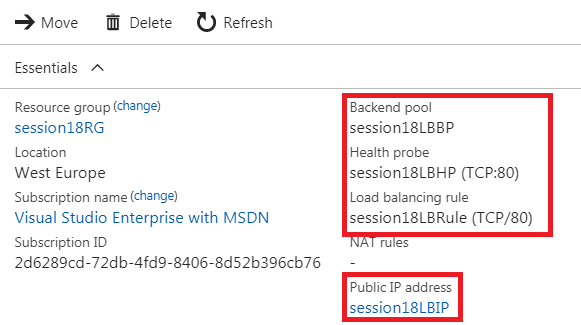
1. Now select ‘Health probes’ then ‘Add’ and fill as below then OK



1. Finally select ‘Load balancing rules’ then ‘Add’ and complete as below then OK



1. This completes the configuration of the load balancer, with the ‘Overview’ showing

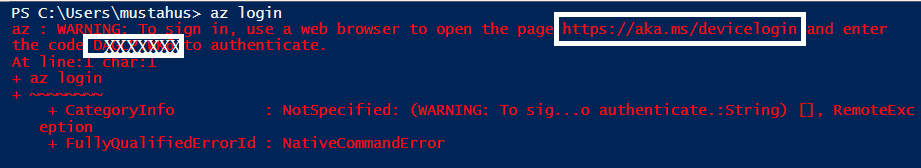


# Create Virtual Machine Scale Set

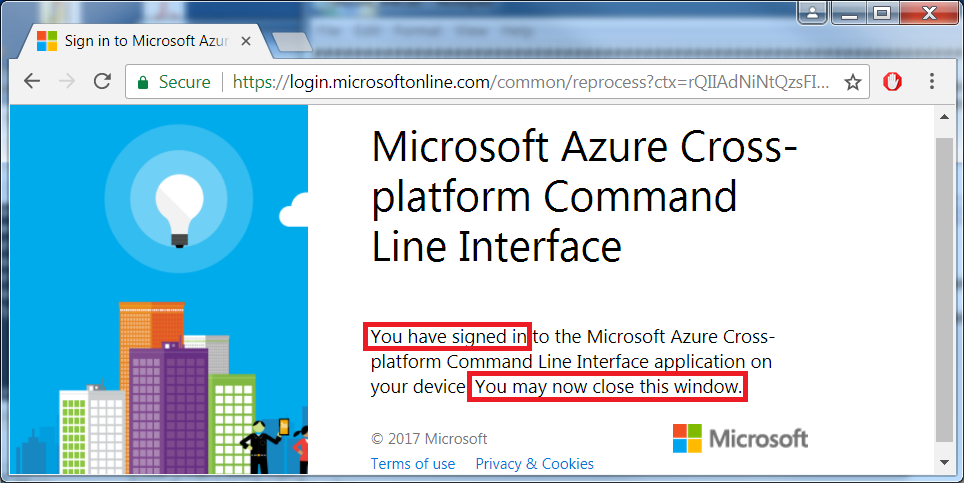
Via the portal Virtual Machine Scale Sets (VMSS) cannot be associated with user defined images (maybe in the future) but you can via Command Line Interface (CLI) or PowserShell cmdlets.

1. On your laptop, click Windows Start then type ise in search button.
2. Select Windows PowerShell ISE
3. Login to Azure, at the PowerShell prompt enter

az login



1. In a browser open the webpage in warning message <https://aka.ms/devicelogin>, enter your code, ‘Sign In’ then close browser as requested



1. Back in PowerShell setup some variables (which should match those just created in this session via the portal) by typing:

$myResourceGroup = "session18RG"

$location = "West Europe"

$vmSKU = "Standard\_A0"

$vmInstances = 1

$myVnet = "session18RG-vnet"

$mySubnet = "webTier-sn"

$myNSG = "webTier-nsg"

$myImage = "myGoldImagev1.1.0"

$adminuser = "dwpadmin"

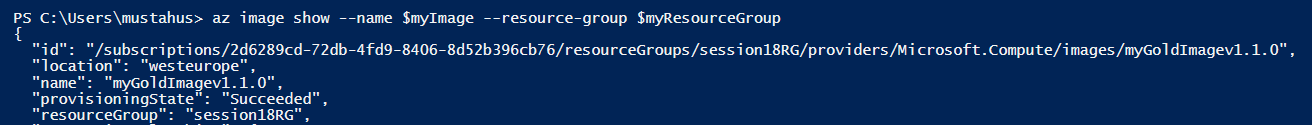
$adminpassword = "Pass@word011"

$vmScaleSetName = "session18vmss"

$myLB = "session18LB"

1. Get the “id” of the gold build image you created in the previous session

az image show --name $myImage --resource-group $myResourceGroup



1. Copy the “id” into a new variable $myImageID

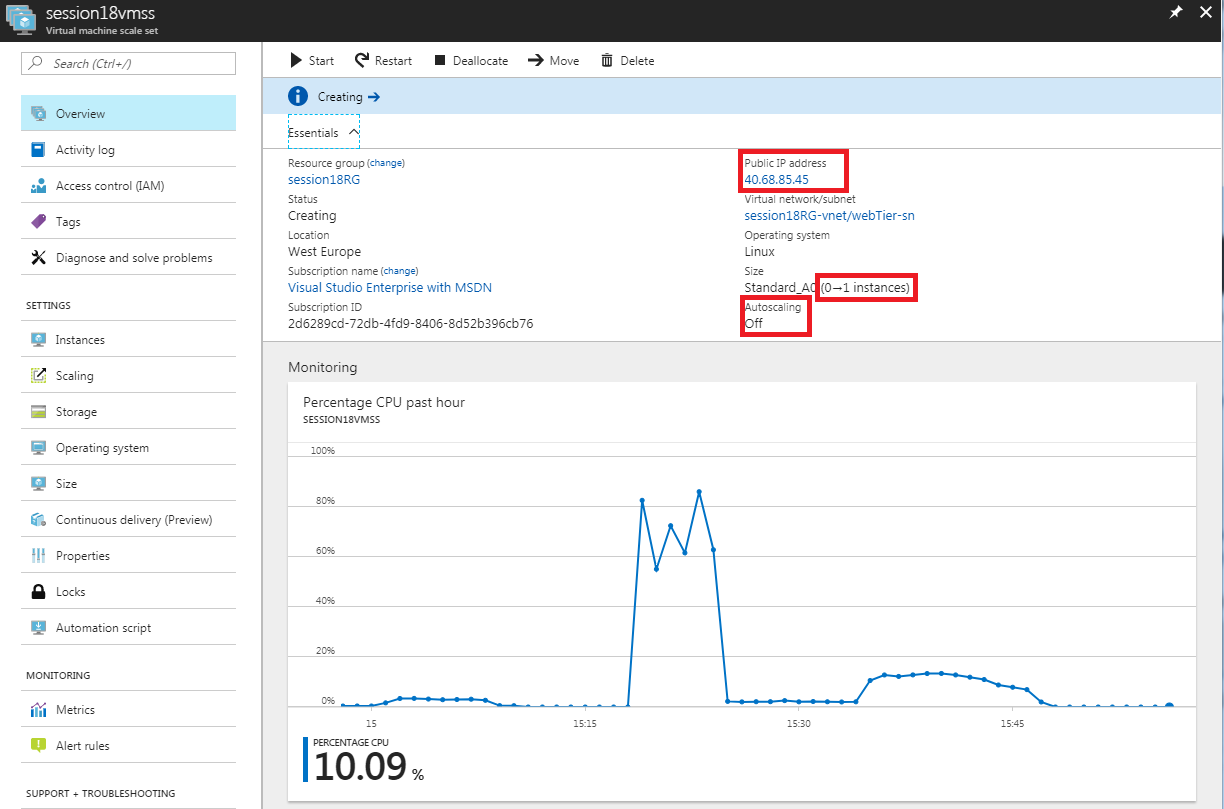
$myImageID = "/subscriptions/…../providers/Microsoft.Compute/images/myGoldImagev1.1.0"



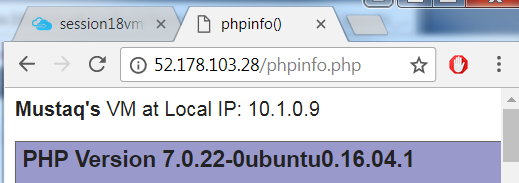
1. Now create a VMSS with 1 VM in scale set from your image with a load balancer:

az vmss create --resource-group $myResourceGroup --name $vmScaleSetName --image $myImageID --authentication-type password --admin-username $adminuser --admin-password $adminpassword --lb $myLB --instance-count $vmInstances --vm-sku $vmSKU

1. Back in your portal in the ‘Resource groups’ select ‘session18RG’ and select ‘session18vmss’ in listed items to show 1 instance being commissioned, Public IP address (which is the Load Balancer’s) and the ‘Autoscaling Off’



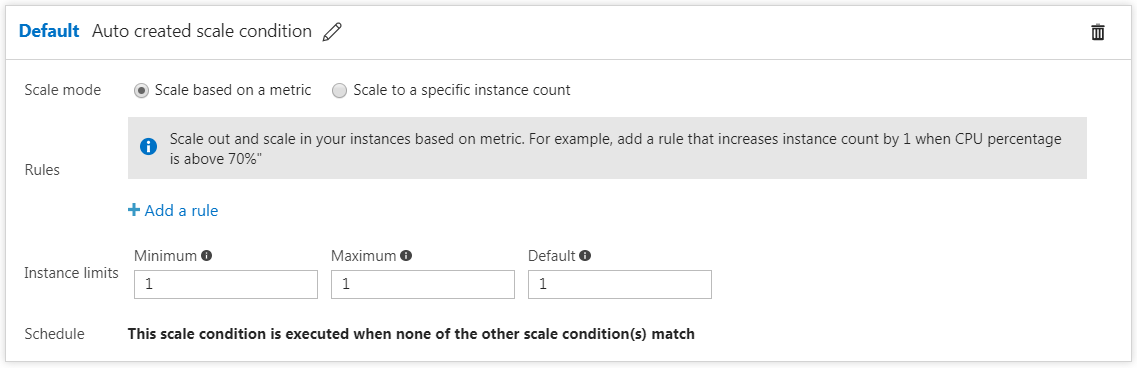
1. Open a browser and enter its ‘Public IP address’ – you should see the default Apache page
2. Change browser address to ‘Public IP address’/phpinfo.php – you should see PHP script with a local IP



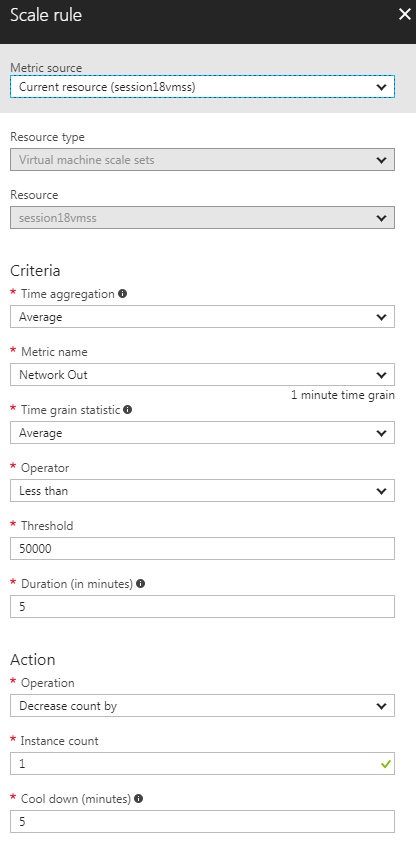
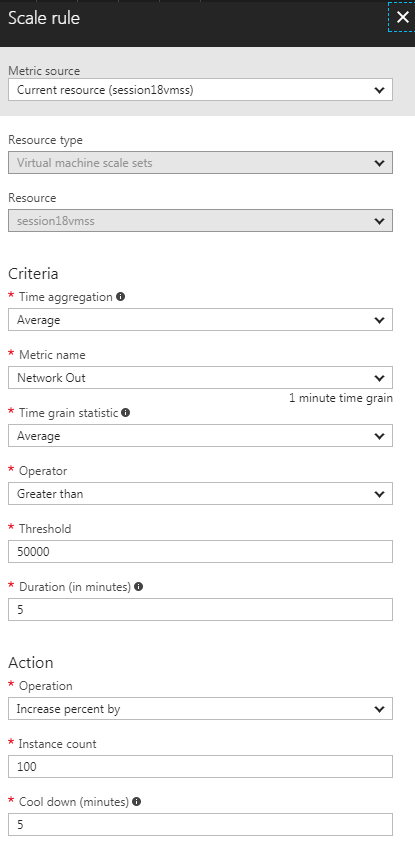
# Creating Auto Scaling Rules

Here you will be creating the following scaling rules:

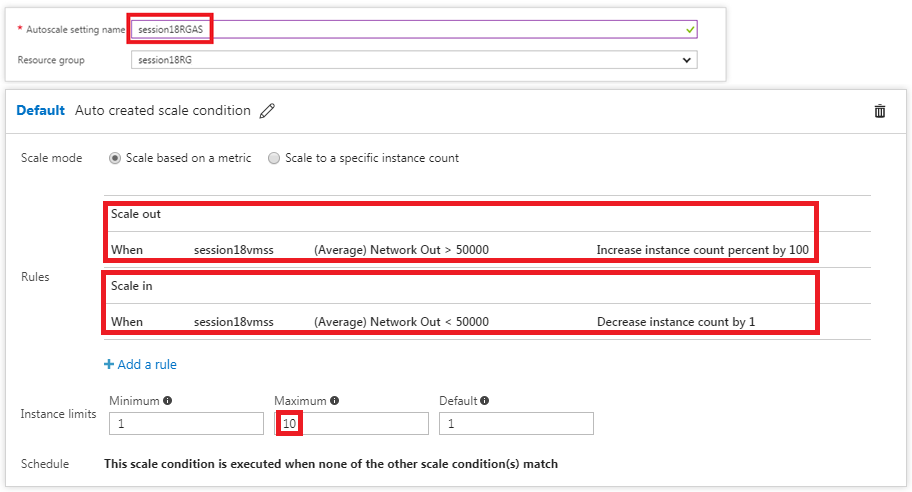
1. Scaling out rule : If avg network out > 50,000 bytes increase by 100% number of instances
2. Scaling in rule : If avg network out < 50,000 bytes decrease by 1 instance
3. From VMSS blade select ‘Scaling’, ‘Enable autoscale’ and ‘Add a default scale condition’ to show



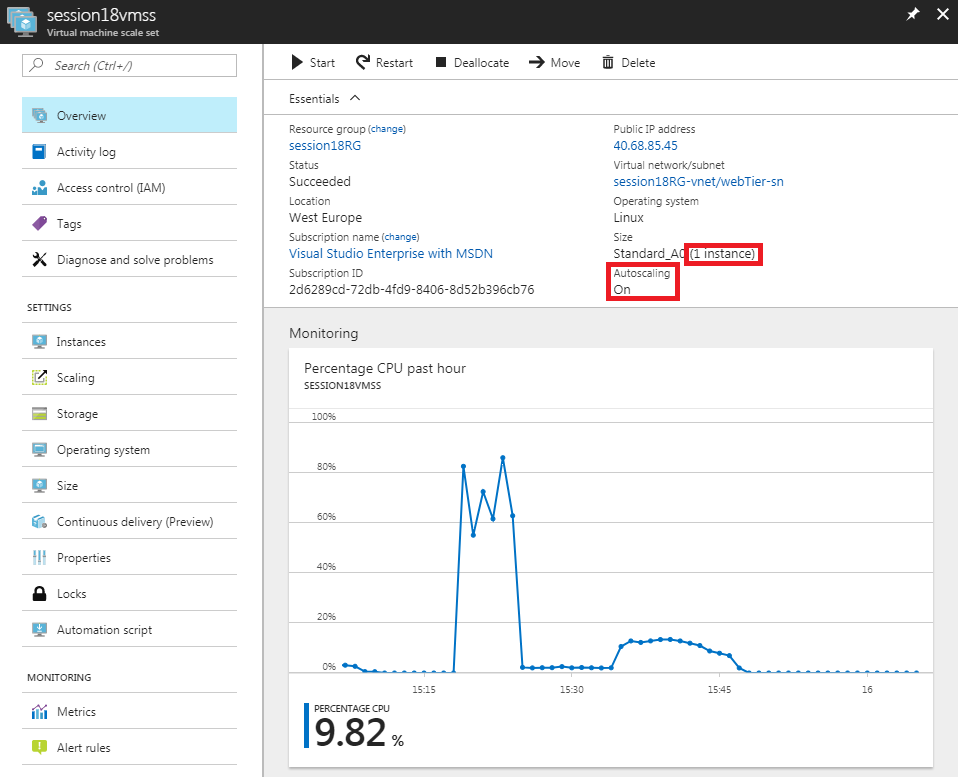
1. Create a Scale out rule by selecting ‘Add a rule’ and fill-in as shown below then ‘Add’; then create a Scale in rule by selecting ‘Add a rule’ fill-in as shown below then ‘Add’



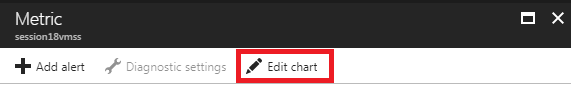
1. Finally set name and ‘Instance limits’ as below then ‘Save’.



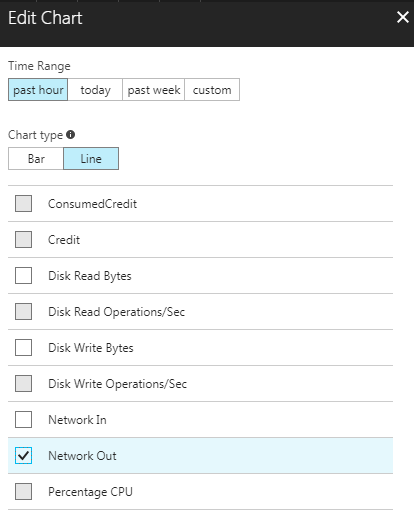
1. Back at the ‘Overview’, Autoscaling is now On



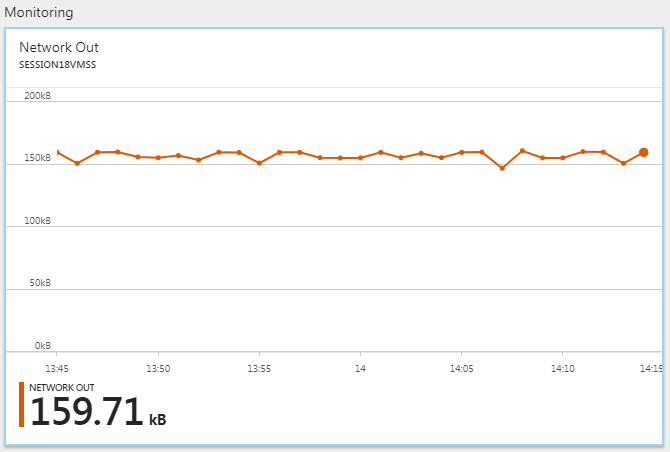
1. To see the Network Out metric change the ‘Monitoring’ metric view by double clicking inside the graph then inside ‘Metric’ blade click ‘Edit chart’



1. Unselect CPU then select ‘Network Out’



1. Click ‘OK’ then close ‘Metric’ blade to get back to ‘Overview’ showing the ‘Network Out’ metric



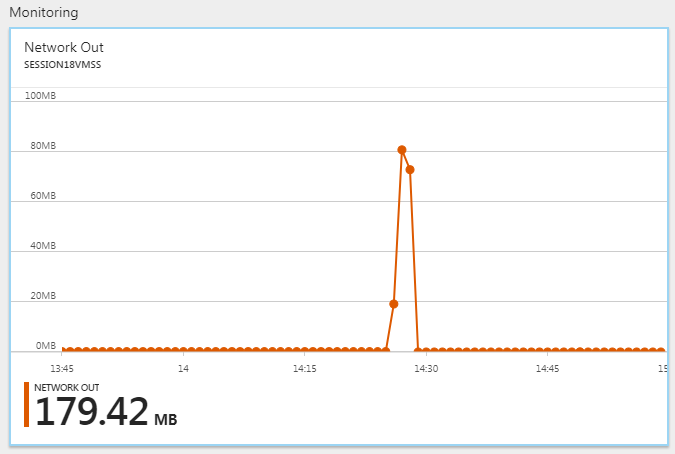
# Automatically Scale Up and Down Number of Instances

To test the scaling you need to send multiple http request. This can be done using Apache Benchmark tool ab.exe

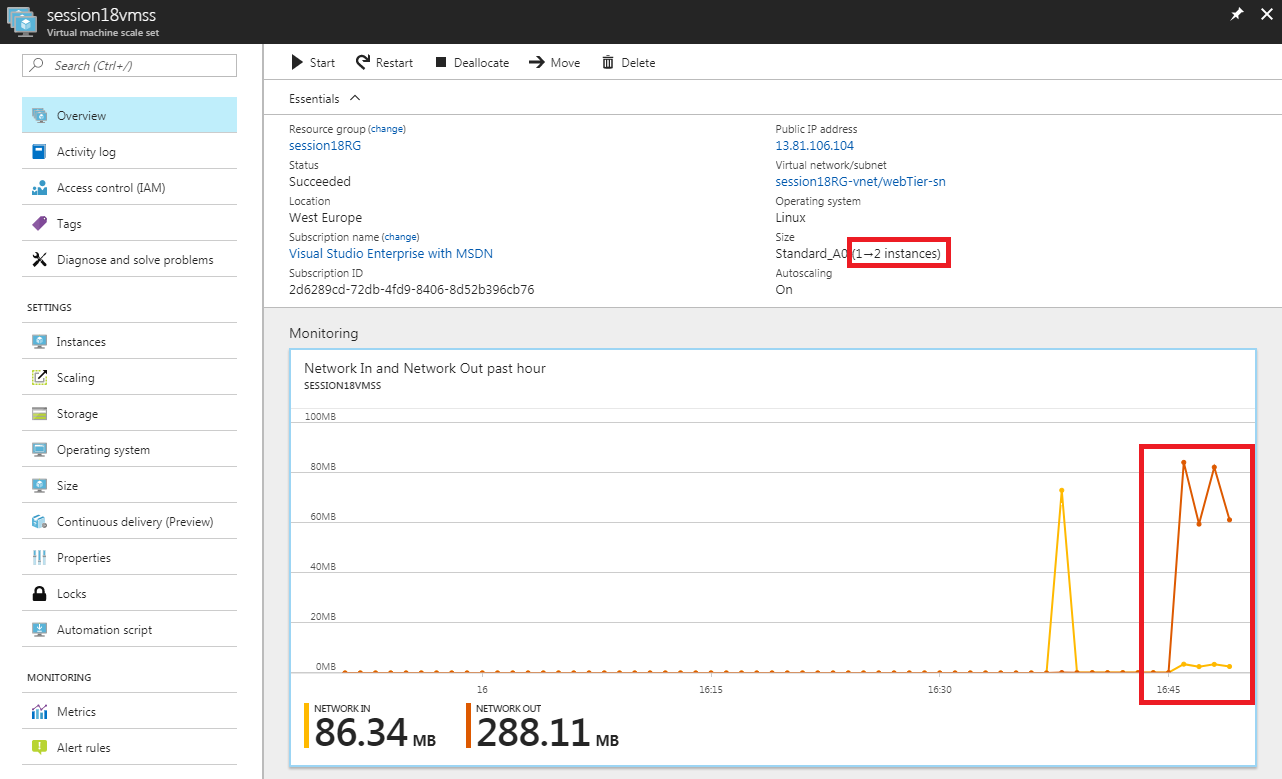
1. Open a command window and change directory where ab.exe installed (by default C:\xampp\apache\bin).
2. Type the following command (-n numberOfRequest, -c numberOfSimultaneousRequest)

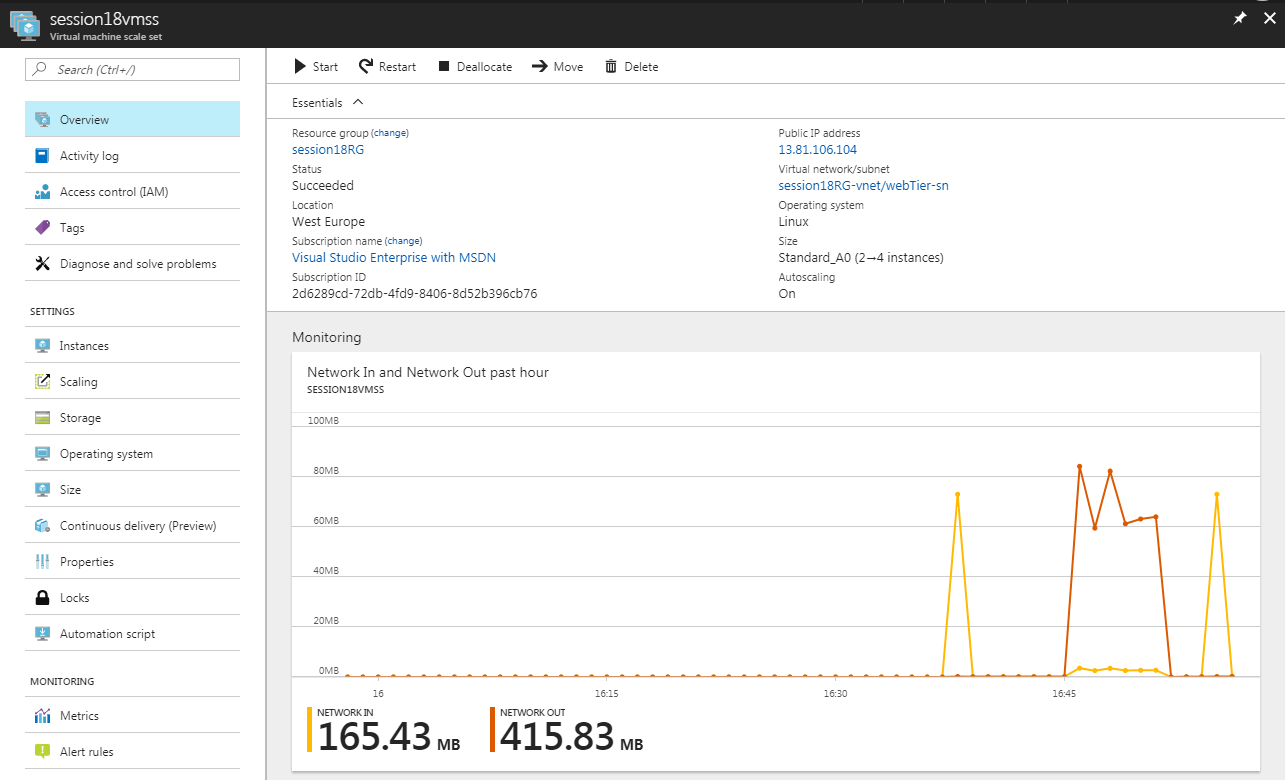
ab -n 1000 -c 50 http://<public IP>/phpinfo.php/

1. After each ‘ab’ statement has completed repeat approximately 10 times (~10minutes). You should see the ‘Network Out’ start to spike in a few minutes (lag in monitor refresh)

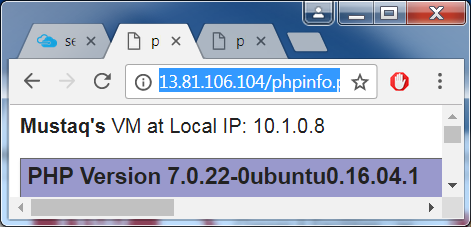
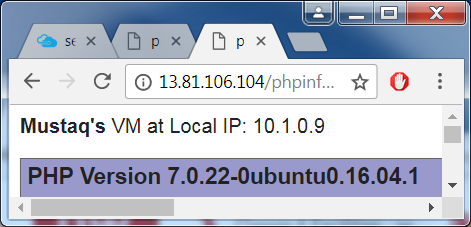


1. Within 10minutes you should see the Instance count doubling to 2 then 4 as system is scaling – up





1. In two browse tabs navigate to http://<Load Balancer IP>phpinfo.php - you should see two difference local IP addresses:

1. After 5minutes inactivity the instance count will decrease by 1 as system is scaling – down

