# Prerequisite

Azure command-line interface     <https://azure.microsoft.com/en-gb/downloads/>

PowerShell                                        <https://azure.microsoft.com/en-gb/downloads/>

Putty (to remote onto Linux SSH)                                                   <https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>  (choose the 64-bit Windows Installer)

# Create Linux Virtual Machine (VM)

This is a network diagram representing where your first Linux VM in Azure will be deployed

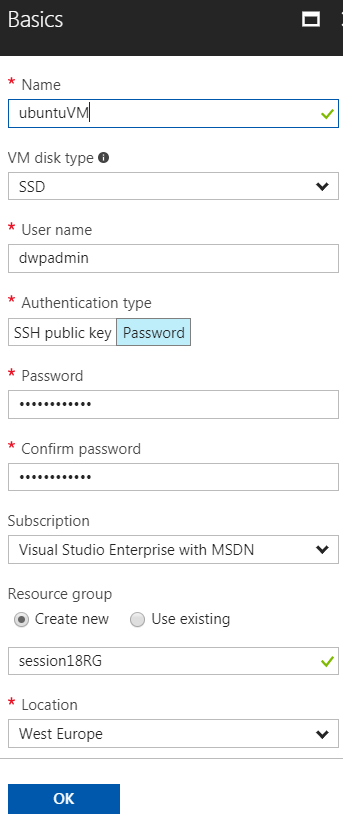
The VM will be Linux Ubuntu 16.04 with Apache as the web server so you can browse to it.



1. Login using

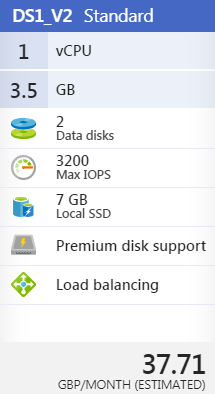
https://portal.azure.com

1. In favourites menu select ‘Virtual Machines’ then ‘Add’ then select ‘Ubuntu Server’
2. In new blade Select ‘Ubuntu Server 16.04 LTS’ then ‘Create’ (keep deployment model Resource Manager)
3. Then fill in basic configuration details as shown below then ‘OK’



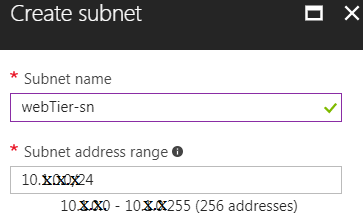
Note Use password: Pass@word011 -- pw length min 12

1. In the ‘Choose a size’ blade click ‘View all’ then scroll down to DS1\_V2 (1 vCPUs & 3.5GM at £37.71/month). Highlight it then click Select



1. In the ‘Settings’ blade select ‘Subnet’
2. Change ‘Subnet name’ from default to webTier-sn then OK

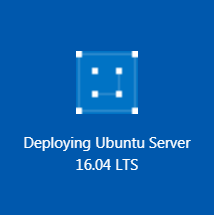
(Note: Your default local Subnet address range 10.1.0.0/24 may be different from screen shot, but you should have 256 address so leave as is)



1. Back in the ‘Settings’ blade select ‘Network security group (firewall)’
2. In the ‘Create network security group’ blade change Name to webTier-nsg
3. By default SSH (TCP/22) port allowed, select ‘+Add an inbound rule’
4. Modify as shown below to allow web traffic (HTTP) on port 80, then OK, OK, OK.

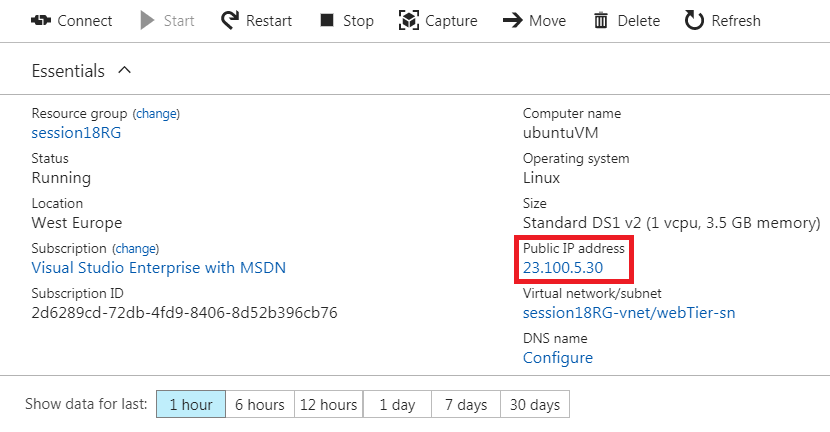


1. Finally at Purchase blade tick Terms of Use and select ‘Purchase’
2. Dashboard will now show ‘Deploying Ubuntu Server’



1. After 2-3minutes it will have finished deploying and opened up the ubuntuVM Virtual Machine blade automatically (or simply click on the ubuntuVM icon on the Dashboard)

Take a note of the ‘Public IP address’ (note this will change whenever you stop/restart VMs)

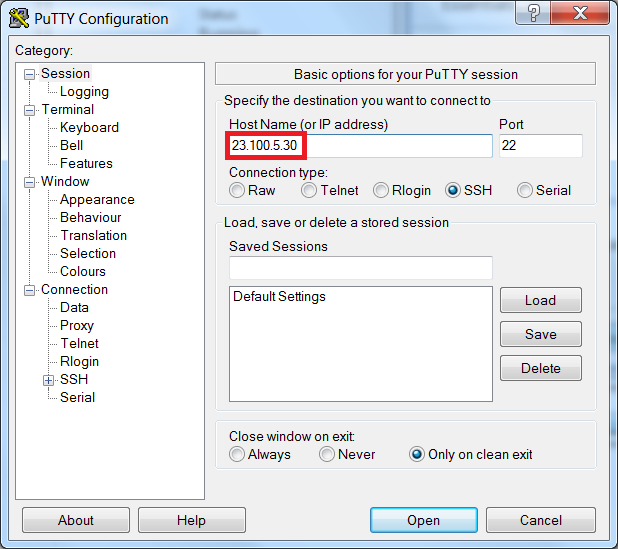


1. Open a browser and enter the ‘Public IP address’
2. Web page will not open because no Web Server is installed on ubuntuVM

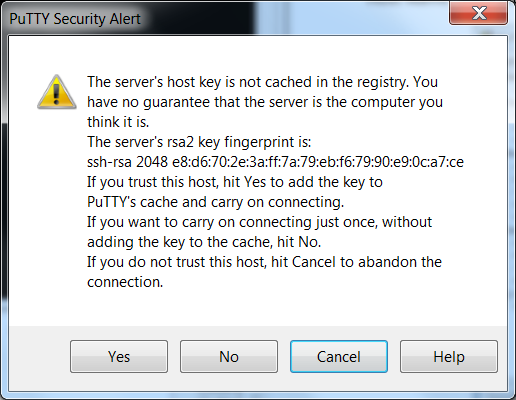
# Connect to Virtual Machine (VM)

First step to install a Web Server on VM is to logon to VM using SSH via Putty.

1. On your laptop click Windows Start then type putty in search button
2. Select PuTTY
3. In the PuTTY Configuration box enter the public IP address of the newly created VM in the ‘Host Name’ then press Open



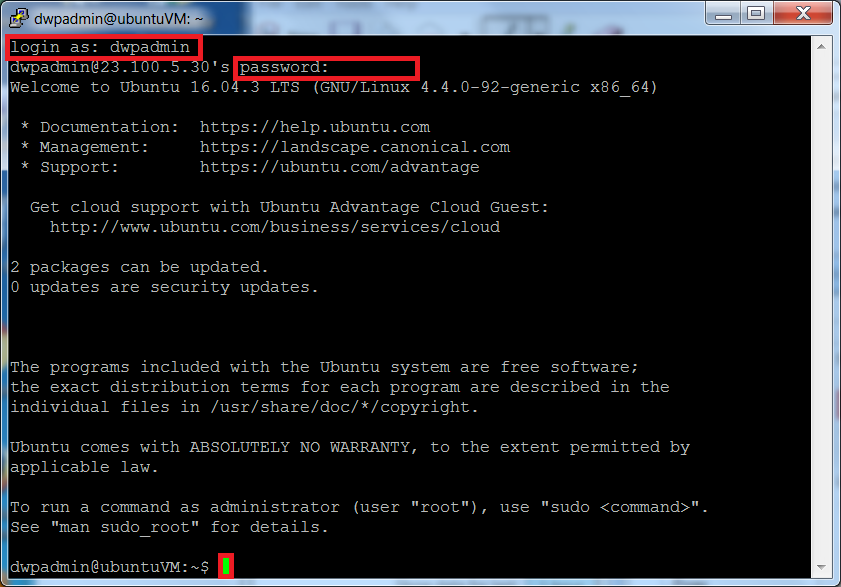
1. Select Yes to the PuTTY Security Alert



1. In the Putty Ubuntu command window enter the login and password

(Login : dwpadmin

Password: Pass@word011 -- NOTE password is not echoed)



1. You should now be logged into your Ubuntu VM server in a Datacentre in the ‘West Europe’ region

# Install Apache HTTP Server & PHP in VM

In this section, using apt (Advance Packing Tool), you will:

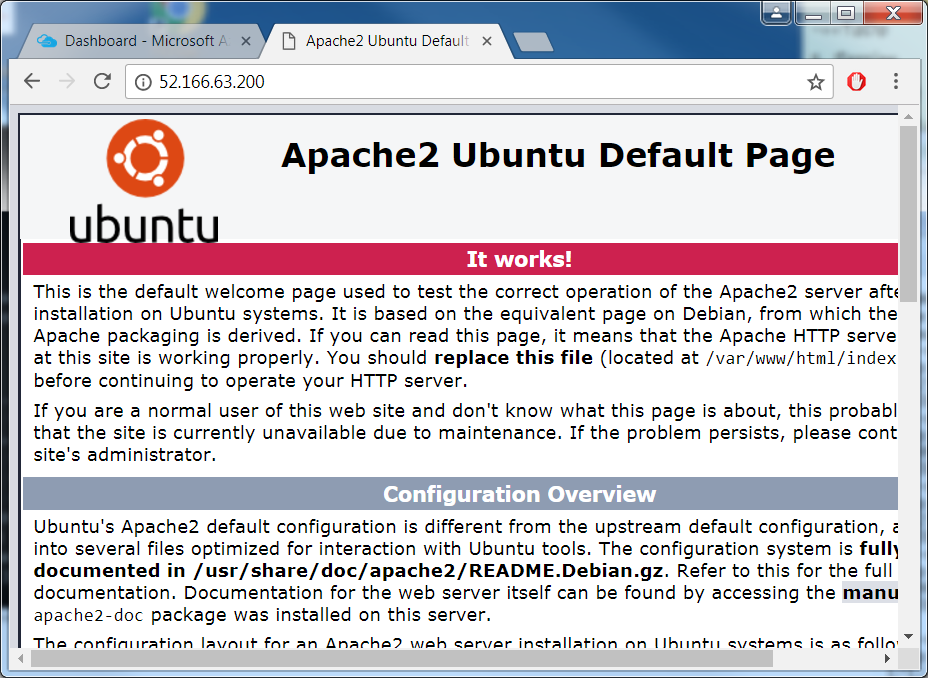
1. Update the server with the latest updates
2. Install Apache Web Server (similar to Microsoft’s Web Server IIS)
3. Install PHP (a server-side scripting language) and
4. Create a simple PHP page
5. At the Ubuntu prompt ‘~$’ update the server:



1. At the prompt ‘~$’ install Apache



1. Check Apache installed okay by opening a browser and entering the ‘Public IP address’



Apache default page should now appear (Web Server is installed successfully on VM)

1. Back at the prompt ‘~$’ install PHP, then restart Apache:



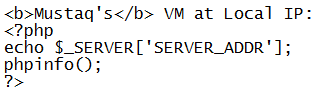
1. Change directory to Apache location:

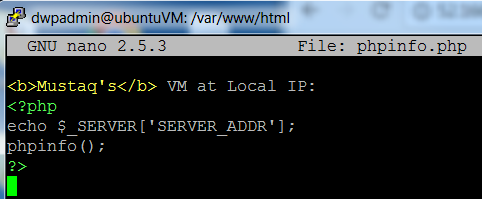


1. Open text editor:



1. Enter following PHP script (HTML with embedded PHP scripts between tags <?....?>)





but replace my name with your name so it can be identified as your VM later on

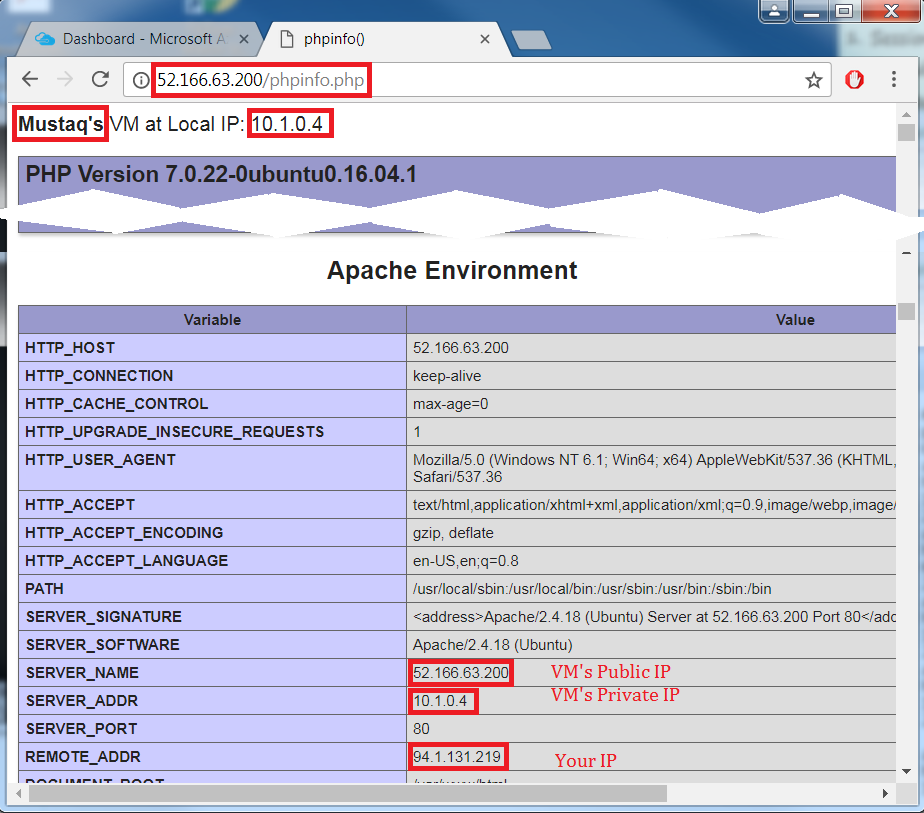
1. Save the file by entering (or following instructions in nano screen)

Ctrl x

y

Enter

1. Open a browser and enter ‘Public IP address’/phpinfo.php



You should now have a PHP page. Scroll down to Apache Environment to see the public & private IPs of your VM and your local IP.

1. Open another browser and enter the ‘Public IP address’/phpinfo.php of some else. You should see their website

You have been connecting successfully to your server deployed in the Azure Cloud via the internet

# Create a Gold Image of VM

To create multiple copies of your VM (including OS, installed software, machine size – number of cores, memory) in Azure you can capture an image of the VM.



Images cannot be created in the portal (many change in the future), only via the Command Line Interface (CLI) commands or PowerShell commands. In the following section you will:

1. Deprovision your VM
2. Connect to Azure via Command Line Interface (CLI) in Windows PowerShell ISE
3. Deallocate (stopped) VM
4. Create an image of your VM
5. Provision new VM from newly created image
6. Back in your Ubuntu server, at the prompt ‘$’ deprovision the VM which removes account specific details (so it can be deployed multiple times):

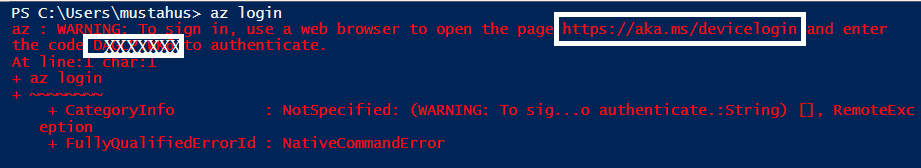
sudo waagent -deprovision -force

1. Close the console by entering:

exit

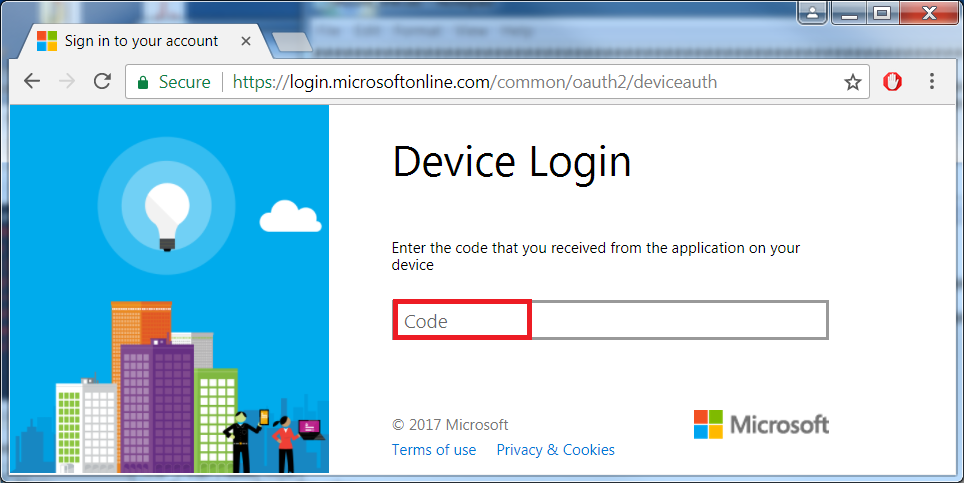
1. In your laptop, click Windows Start then type ise in search button.
2. Select Windows PowerShell ISE
3. To login to Azure, at the PowerShell prompt enter

az login

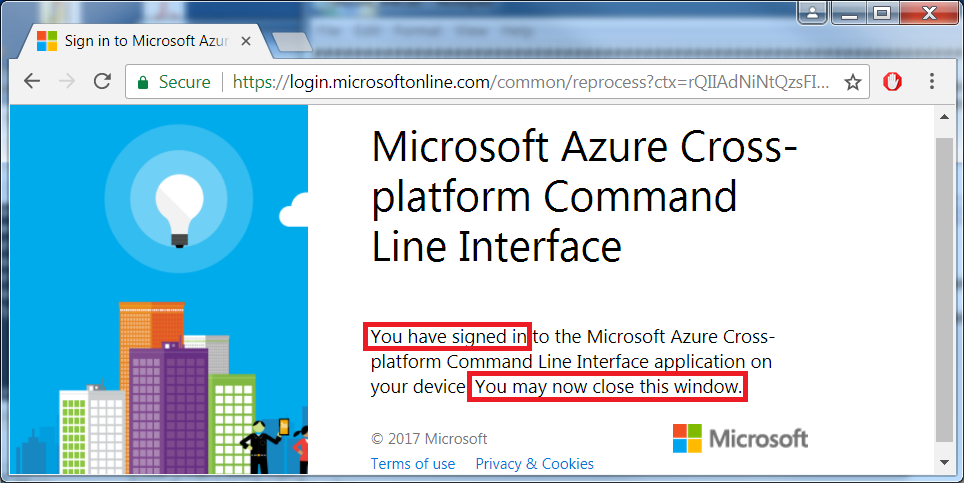


You should see an az: WARNING containing a web page address and your code for authentication

1. In a browser open the webpage in warning message <https://aka.ms/devicelogin> which will redirect you to a login page



1. Enter your code, click Continue, complete ‘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"

$myVM = "ubuntuVM"

$myVnet = "session18RG-vnet"

$mySubnet = "webTier-sn"

$myNSG = "webTier-nsg"

$myImage = "myGoldImagev1.1.0"

$myNewVMFromImage = "ubuntuFromImageVM"

$adminuser = "dwpadmin"

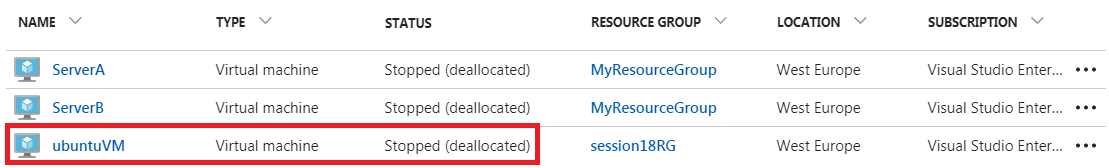
$adminpassword = "Pass@word011"

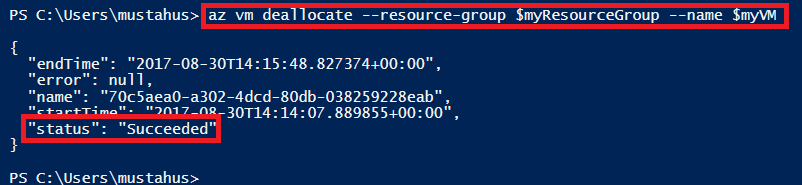
$myNewVMFromImage = "ubuntuVMFromImage"

1. ‘Stop (deallocated)’ your VM from its ‘Running’ state:

az vm deallocate --resource-group $myResourceGroup --name $myVM

1. Back in your Azure Portal under ‘Virtual Machines’ you should see your VM with a status of ‘Stopped (deallocated)’ in about 2-3 minutes, as well as a status of ‘Succeeded’ in PS (you will not be able to browse to your VM now)





1. Mark the VM as being a generalized VM:

az vm generalize --resource-group $myResourceGroup --name $myVM

1. Now create a managed image (aka Gold Build) of your VM

az image create --resource-group $myResourceGroup --name $myImage --source $myVM

1. Now create a new VM from the managed image just created

az vm create --resource-group $myResourceGroup --vnet-name $myVnet --subnet $mySubnet --nsg $myNSG --name $myNewVMFromImage --image $myImage --admin-username $adminuser --admin-password $adminpassword

1. Back in your portal you should see it being created



then finally running:



1. Click on ubuntuVMFromImage and take a note of its public IP
2. Open a browser and enter its ‘Public IP address’ – you should see the default Apache page
3. Change browser address to ‘Public IP address’/phpinfo.php – you should see PHP script
4. Try creating a 2nd/3rd VM from this image – just reset variable

$myNewVMFromImage = "ubuntuFromImage2ndVM"

then re-run az vm create command

1. Finally stop all your VMs (otherwise you will use up your allowance before too long)

References for creating images

<https://docs.microsoft.com/en-us/azure/virtual-machines/linux/capture-image>

<https://docs.microsoft.com/en-us/azure/virtual-machines/linux/tutorial-custom-images>

sudo apt-get update -y

sudo apt-get install -y apache2

sudo apt-get install -y php7.0 libapache2-mod-php

sudo /etc/init.d/apache2 restart

cd /var/www/html

sudo nano phpinfo.php

<b>Mustaq's</b> VM at Local IP:

<?php

echo $\_SERVER['SERVER\_ADDR'];

phpinfo();

?>

sudo waagent –deprovision

az login

$myResourceGroup = "session18RG"

$location = "West Europe"

$myVM = "ubuntuVM"

$myVnet = "session18RG-vnet"

$mySubnet = "webTier-sn"

$myNSG = "webTier-nsg"

$myImage = "myGoldImagev1.1.0"

$myNewVMFromImage = "ubuntuFromImageVM"

$adminuser = "dwpadmin"

$adminpassword = "Pass@word011"

$myNewVMFromImage = "ubuntuVMFromImage"

az vm deallocate --resource-group $myResourceGroup --name $myVM

az vm generalize --resource-group $myResourceGroup --name $myVM

az image create --resource-group $myResourceGroup --name $myImage --source $myVM

az vm create --resource-group $myResourceGroup --vnet-name $myVnet --subnet $mySubnet --nsg $myNSG --name $myNewVMFromImage --image $myImage --admin-username $adminuser --admin-password $adminpassword