# AZURE- Install a VM for an Indy Node

Introduction: The following steps are one way to adhere to the Indy Node guidelines for installing an Azure VM server to host an Indy Node. For the hardware requirements applicable for your network, please refer to the steward technical requirements document or the network governance documents.

1. From the Azure portal ‘home’ click 'Create a resource'
2. Type “ubuntu server” in the search field, ‘Enter’, then select 'Ubuntu server 16.04 LTS'
3. Click 'Create' to deploy with Resource Manager
4. TIP: Throughout the process of going through all of the tabs in the next several steps do NOT click your browsers back button as this will remove all previous selections and cause you to have to start over. Instead, you should either click on each tab across the top of the interface, or use the ‘Previous’ and ‘Next’ buttons at the bottom for navigation between the steps.
5. Basics tab
   1. Project Details
      1. Subscription - Your choice
      2. Resource group - Suggest creating a new one of your choosing. For example: NODE
   2. Instance Details
      1. VM Name - Your Choice
      2. Region - Suggest selecting the region that your business resides in
      3. Availability options - select 'No infrastructure redundancy required'
      4. Image - Default (already filled in with Ubuntu Server 16.04 LTS)
      5. Azure Spot instance - select No
      6. Size - click 'change size', click the x by 'Size : Small(0-6)', and select a size with at least 8 vCPUs and 32G RAM or greater then click ‘select’. Minimum: **Standard B8ms,** 8 vcpus, 32 GiB memory
   3. Administrator Account
      1. Authentication type: SSH public key
      2. Username: <your choice>
      3. SSH public key:
         1. Create a SSH public key that you will use to access this VM. Click the link “[Learn more about creating and using SSH keys in Azure](http://go.microsoft.com/fwlink/?LinkId=2102401)” for detailed information. Be careful not to overwrite any old keys that you have and use.
         2. TIP: you can use the following command to create a new key pair on Linux or MAC that will work for this step (copied from AWS instructions)
            1. ssh-keygen -P "" -t rsa -b 4096 -m pem -f ~/pems/validatornode.pem
         3. Once a public key is created the following example can be used on MAC or Linux to display the public key and copy it to the form:
            1. cat ~/pems/validatornode.pem.pub
            2. Copy the results of the above and paste it into the space provided being careful NOT to copy any leading or trailing whitespace.
   4. Inbound port rules
      1. Public inbound ports - Select 'None'
6. Disks tab
   1. Disk options
      1. OS disk type - standard HDD is inexpensive and acceptable, but the choice is yours.
      2. Encryption type - (Default) is acceptable
   2. Data disks - click 'Create and attach a new disk' and a new entry screen appears
      1. Name - default is fine
      2. Source Type - default (None)
      3. Size - click 'Change size' -> Select storage type as Standard HDD (or better) and then select 1024 GiB (1TB) and click OK
      4. Encryption type - default
      5. Click ‘OK’
   3. Leave LUN as 0
   4. Leave advanced options at default (use managed disks - Yes) NOTE: changing managed disks to No is not supported and it resets all selections made in the Data Disks section!
   5. DO NOT click 'Review + create' yet. This is not for reviewing and creating the disk, it is for the whole VM and we have a few more tabs to go through before we are ready for that.
7. Networking tab
   1. Virtual network - default
   2. Subnet - default
   3. Public IP - click ‘Create new’
      1. SKU - standard (NOTE: this is critical! It must match what you do later on the Second IP address)
      2. Click ‘OK’
   4. NIC network security group - Advanced
   5. Configure network security group - click 'Create new'
      1. Change the name for ease of identification, because this is the Client NIC’s nsg and it will operate on port 9702 (i.e. ValidatorClient9702-nsg)
      2. Click on the provided default SSH entry to change it
         1. Source - IP Addresses
         2. Source IP addresses - add all the Node admins’ workstations IP addresses to allow them to access the machine for maintenance.
         3. Priority - 900
         4. Leave the rest of the values at default and click ‘Save’ (upper left).
      3. Click '+ Add an inbound rule' to setup this NIC as the “Client” NIC
         1. Source - Any
         2. Source port ranges - \* (Any)
         3. Destination - Any
         4. Destination port ranges - 9702
         5. Protocol - TCP
         6. Name - your choice (probably make it more appropriate than the default, i.e. ClientPort\_9702)
         7. Description - add if desired. For example, I added: “This is the rule that opens up port 9702 for all client connections to the Validator Node.”
         8. Click “Add”
      4. You should now have 2 inbound rules, 1 for admin SSH access and 1 for client access to the node on port 9702.
      5. Click 'OK'
   6. Accelerated networking - Off
   7. Place this virtual machine behind an existing load balancing solution? - No
8. Management tab
   1. Boot diagnostics - On
   2. OS guest diagnostics - On
   3. Diagnostics storage account - Default is fine
   4. System assigned managed identity - Off
   5. Enable auto-shutdown - Off (required)
   6. Enable backup - On (unless you have another backup solution. Some type of backup is required)
      1. Recovery Services vault - your choice (some of the steps below will change if you select ‘Use existing’)
      2. Resource group - your choice. I used the example from earlier (new) NODE
      3. Backup Policy - your choice. Suggested Backup policy setup follows:
         1. Click ‘Create new’
         2. Policy name - WeeklyPolicy
         3. Backup schedule - your choice
         4. Retain instant recovery snapshots for - 5 Day(s)
         5. Retention range - defaults are fine
         6. Click ‘OK’
9. Advanced tab
   1. Defaults are fine
10. Tags tab
    1. No tags needed
11. Click Review + create
    1. Check all values for accuracy
    2. Example:

PRODUCT DETAILS

Standard B8ms by Microsoft

Subscription credits apply

**0.3330 USD/hr**

**Basics**

Subscription - Corporate (pay as you go)

Resource group - (new) NODE

Virtual machine name - ValidatorNode1

Region - West US 2

Availability options - No infrastructure redundancy required

Authentication type - SSH public key

Key pair name - lynnbendixsen

Azure Spot - No

**Disks**

OS disk type - Standard HDD

Use managed disks - Yes

Data disks - 1

Use ephemeral OS disk - No

**Networking**

Virtual network - (new) NODE-vnet

Subnet - (new) default (10.0.1.0/24)

Public IP - (new) ValidatorNode1-ip

NIC network security group - (new) ValidatorClient9702-nsg

Accelerated networking - Off

Place this virtual machine behind an existing load balancing solution? - No

**Management**

Boot diagnostics - On

OS guest diagnostics - On

Azure Security Center - Basic (free)

Diagnostics storage account - (new) nodediag501

System assigned managed identity - Off

Auto-shutdown - Off

Backup - Enabled

Recovery Services vault - (new) vault203

Vault resource group - (new) NODE

Backup policy - (new) WeeklyPolicy

**Advanced**

Extensions - None

Cloud init - No

Proximity placement group - None

1. Click ‘Create’
2. Wait for the message “Your deployment is complete” (Hint: This might take a few minutes)
3. Click ‘Go to resource’
4. On the VM overview screen find and record the public and private(local) IP addresses for later use. These are the Client IP’s.
5. Stop your Virtual machine so that you can add a new NIC.
   1. From Azure Portal Home, select your virtual machine then select ‘overview’.
   2. From the menu bar across the top select ‘Stop’ then ‘Yes’ to stop the VM
   3. Wait for notification that states “Successfully stopped virtual machine” (Hint: This could take several minutes, you can create a new Node IP address while you wait, but you won’t be able to add the new NIC until the VM is stopped)
6. Add a subnet to your virtual network.
   1. Select ‘Virtual networks’ from the Azure Home screen. (Hint: You might need to click ‘More services’ and search for it if you do not see it on the main page.)
   2. Click on the new Virtual network that you made earlier as part of these instructions. (example: NODE-vnet)
   3. Click ‘Address space’ in the left menu
      1. Click in the ‘Add additional address range’ entry box
      2. Type in 10.0.2.0/24
      3. Click ‘Save’ in the top menu
   4. Click ‘Subnets’ in the left menu
      1. Click ‘+ Subnet’ on the top menu
      2. Name - your choice (i.e. nodeSubnet-9701)
      3. Address range - 10.0.2.0/24
      4. Defaults for the rest
      5. Click ‘OK’
7. Add a second NIC to ValidatorNode VM
   1. From Azure home, find and select your new VM
   2. Select ‘Networking’ from the side menu of the Azure portal Virtual machine interface.
   3. Select ‘Attach network interface’ from the top menu
   4. Click ‘Create network interface’
   5. Name - your choice (ValidatorNode9701-NIC)
   6. Subnet - Select the subnet created in the previous step of these instructions.
   7. Private IP address assignment - Static
   8. Private IP address - 10.0.2.5 (or other if preferred. Record this as private(local) ip address of node\_ip)
   9. (Skip to Resource group first so that you don’t forget it)
      1. Resource group - NODE (must be the same as the new Node VM)
   10. Network security group - Click the arrow to create a new group.
       1. Click ‘+ Create new’
       2. Name - ValidatorNode9701-nsg
       3. The following steps must be repeated for each Node in the Indy Network that you will be a part of. For a list of IPs and ports in your network, please ask your network administrator.  
           Note: This step can be done later and the “whitelist” that you begin during this step needs updated every time a new node is added to your network. LATER: To get your own list of nodes on your network, run the following command from your validator node after installation is complete and the node is added to the network:  
           > **sudo current\_validators --writeJson | node\_address\_list**
          1. Click ‘+ Add an inbound rule’
          2. Source - IP Addresses
             1. Enter the next IP address from the IP/port list.
          3. Source port ranges - \*
          4. Destination - IP addresses
             1. 10.0.2.5
          5. Destination port ranges - 9701 (Must match the port that you will set up later in the Node software configuration and must be the same for all rules added to the whitelist)
          6. Protocol - TCP
          7. Action - Allow
          8. Priority - your choice (default value should work here)
          9. Name - your choice. Recommended to add the name of the node allowed access for ease of future removal or IP/port change.
          10. Click ‘Add’
       4. Repeat steps iii.1-10 above until all nodes in the network have been added to your “whitelist”
       5. Click ‘OK’ to complete the Security Group creation
   11. Click ‘Create’ to create the new NIC
   12. Select ‘Attach network interface’ from the top menu (again)
   13. Select the new NIC form the dropdown list
   14. Click ‘OK’ to complete the addition of the new NIC to the VM
8. Add a static public IP address to the new NIC
   1. Click on the new Network Interface name then click on the name again (next to Network Interface) to open the ‘Overview’ view for the new NIC.
   2. Click ‘IP configurations’ in the left menu.
   3. Click ‘ipconfig1’ to open the settings for the configuration and add the IP address
   4. Public IP address - Associate
      1. Click the right arrow next to ‘IP address - Configure required settings’
      2. Click ‘+ Create new’
         1. Name - your choice (i.e. ValidatorNode9701-ip)
         2. SKU - Standard (HINT: This value must match what you used for the first IP address for your VM!)
         3. Click ‘OK’
      3. Assignment - Static
      4. IP address - 10.0.2.5
      5. Click ‘Save’
   5. Click the ‘X’ in the upper right of the active window twice to close the IP configuration windows.
   6. Refresh the screen (click on the first NIC and then back on the second NIC?) to view and copy the new Public IP just created. Save that value for future use.
   7. Click ‘Overview’ on the left bar to prepare for the next step.
9. Start your VM
10. Log in to your VM
    1. ssh -i <private rsa key file> <Client IP Address>
    2. Where rsa key file was the ssh key generated earlier (sample steps showed a pem file)
    3. And Client IP is the public address from Nic #1 (Client Public IP from your Node Installation Info spreadsheet)
11. Configure networking to the second NIC
    1. From your VM command prompt, run the command $ ip a and verify that you have 2 internal IP addresses that match what you have in your Node Installation Info spreadsheet. Note the names of the network interfaces. The remaining instructions in this section assume eth0 is your original primary NIC (Client NIC) and eth1 is the secondary NIC (Node NIC).
    2. Perform the following steps to allow routing of traffic to your second NIC.
       1. Create 2 files and fill them with the listed contents. In vim, press ‘i’ to enter editing mode, paste the content listed, press esc, then type :wq to save and quit.
       2. vim /etc/network/interfaces.d/51-eth1.cfg
       3. Contents:

auto eth1

iface eth1 inet static

address 10.0.2.5

netmask 255.255.255.0

# Gateway configuration

up ip route add default via 10.0.2.1 dev eth1 table 1000

# Routes and rules

up ip route add 10.0.2.5 dev eth1 table 1000

up ip rule add from 10.0.2.5 lookup 1000

* + 1. vim /etc/dhcp/dhclient-enter-hooks.d/restrict-default-gw
    2. Contents:

case ${interface} in

eth0)

;;

\*)

unset new\_routers

;;

esac

* + 1. systemctl restart networking
    2. to double check the routing run route -n

1. Configure and mount the data disk.
   1. Find the name of your data disk:
      1. sudo fdisk -l
      2. In most cases **/dev/sdc** will be the name of the 1T data disk created during Azure VM setup.
   2. The following steps assume that your disk size is less than 2 TiB, that your disk is /dev/sdc and that you will be using MBR partitioning.
   3. sudo fdisk /dev/sdc
      1. Create a new partition
         1. n
         2. p
         3. <defaults for the rest> TIP: press enter 3 times to accept the defaults and complete the process of creating a partition.
         4. Now, print and write the partition and exit.
         5. p
         6. w
   4. Update the kernel:
      1. partprobe
   5. Add a filesystem to your new disk partition:
      1. sudo mkfs -t ext4 /dev/sdc1
   6. Mount the disk to the directory where the Node software does the most writing (/var/lib/indy):
      1. sudo mkdir /var/lib/indy
      2. sudo mount /dev/sdc1 /var/lib/indy
   7. Add the drive to /etc/fstab so that it mounts at server startup.
      1. sudo blkid
      2. Record the UUID of /dev/sdc1 for use in the /etc/fstab file.
      3. sudo vim /etc/fstab
      4. Add the following line to the end of the fstab file:
         1. UUID=189d7b71-99b6-44fa-8b26-5ea4db3c9ffd /var/lib/indy ext4 defaults,nofail 1 2
         2. Vim Hint: In vim, arrow down to the last line of the file, press the ‘o’ key and then paste in the above line. As before, <esc> then :wq will write and exit the file.
2. Restart the VM to check for NIC and Disk persistence.
   1. From your Virtual Machine overview in Azure, click ‘Restart’, then ‘Yes’
   2. Login to your VM as before:
      1. ssh -i <public rsa key file> <Client IP Address>
   3. Check the NIC
      1. ip a
      2. The output of the above command should have 2 NICS with the correct IP addresses displayed.
      3. df -h
      4. The output of the above command should show /var/lib/indy mounted to the /dev/sdc1 disk with the correct size (1T).
      5. More NIC and disk verifications will occur during the Indy Node install process.
3. Optional: Add a temporary administrative user as a safety net during Two Factor Authentication (2FA) setup.
   1. sudo adduser tempadmin
      1. You can safely ignore messages like “sent invalidate(passwd) request, exiting“
   2. sudo usermod -aG sudo tempadmin
   3. Setup sshd\_config to temporarily allow password login for the tempadmin user.
      1. sudo vim /etc/ssh/sshd\_config
      2. Comment out the line containing ‘ChallengeResponseAuthentication’.
         1. #ChallengeResponseAuthentication no
      3. Make sure this line exists and is set to yes:
         1. PasswordAuthentication yes
      4. :wq to save and exit.
      5. sudo systemctl restart sshd
      6. The above lines will be altered again when you set up 2FA.
   4. To be able to login, you will also likely need to setup an ssh key
      1. sudo mkdir /home/tempadmin/.ssh
      2. sudo chown tempadmin:tempadmin /home/tempadmin/.ssh
      3. sudo vim /home/tempadmin/.ssh/authorized\_keys
      4. Paste the users public key into the open file and then save it (:wq) (You can use the same key as you used for the base admin user in this case, since it is a temporary user)
      5. sudo chown tempadmin:tempadmin /home/tempadmin/.ssh/authorized\_keys
4. Setup 2FA for SSH access to the Node for your base user.
   1. Optional: Login in a separate terminal as your tempadmin user (that has sudo privileges) to have a backup just in case something goes wrong during setup.
      1. ssh tempadmin@<Client IP Addr>
   2. Install Google Authenticator, Duo, or Authy on your phone.
   3. As your base user on the Node VM, run the following to install the authenticator:
      1. sudo apt-get install libpam-google-authenticator
   4. Configure the authenticator to allow both password and SSH key login with 2FA by changing 2 files:
      1. sudo vim /etc/pam.d/common-auth
      2. Add the following line as the first uncommented line in the file
         1. auth sufficient pam\_google\_authenticator.so
         2. :wq
      3. sudo vim /etc/ssh/sshd\_config
         1. add/configure the following lines:
            1. ChallengeResponseAuthentication yes
            2. UsePAM yes
            3. AuthenticationMethods publickey,keyboard-interactive
            4. PasswordAuthentication no
         2. If you see any of the above lines commented out, remove the # to uncomment them. If you don't see any of the above lines, make sure to add them. If you see those lines configured in any different way, edit them to reflect the above.
         3. :wq
      4. sudo systemctl restart sshd
   5. Setup your base user to use 2FA by running the following from a terminal:
      1. google-authenticator
      2. Answer ‘y’ to all questions asked during the setup
      3. Save the secret key, verification code and scratch codes in a safe place. These are all just for your user and can be used to login or to recover as needed.
   6. On your phone app add an account and then scan the barcode or enter the 16 character secret key from the previous steps output.
   7. You should now be able to login using 2FA. First, check that login still works for your base user in a new terminal. If that doesn’t work, double check all of the configuration steps above and then restart sshd again. If it still doesn’t work, it’s possible that a server restart is required to make 2FA work (NOTE: It is dangerous to restart at this point, because then all of your backup terminals that are logged in will be logged out and there is a chance that you will lose access. Please check that all other steps have been executed properly before restarting.)
5. Add other administrative users:
   1. Send the other new admin users the following instructions for generating their own SSH keys:
      1. ssh-keygen -P "" -t rsa -b 4096 -m pem -f ~/pems/validatornode.pem
      2. Have the new users send you their public key validatornode.pem.pub
      3. Also have them send you their Public IP address so that you can add it to the Azure firewall to allow them access.
   2. Add the IP addresses to the Azure firewall:
      1. From the Azure portal, select your VM name and click ‘Networking’ in the left menu.
      2. Select the Client NIC (default) and then click on the priority 900 rule allowing port 22 access to your Client IP.
      3. In the new window that pops up, add the new users' IP addresses to the ‘Source IP addresses’ field, separated by commas.(no spaces)
      4. Click ‘Save’ (Note: Restart is not needed. As soon as you save, they should have access.)
   3. Add the users to the server:
      1. Login to the Node as the base user.
      2. Run the following commands, substituting the username in for <newuser>
      3. sudo adduser <newuser>
         1. You can safely ignore messages like “sent invalidate(passwd) request, exiting“
      4. sudo usermod -aG sudo <newuser>
      5. Then create a file in the newusers home directory:
         1. sudo mkdir /home/<newuser>/.ssh
         2. sudo chown <newuser>:<newuser> /home/<newuser>/.ssh
         3. sudo vim /home/<newuser>/.ssh/authorized\_keys
         4. Paste the users public key into the open file and then save it (:wq)
         5. sudo chown <newuser>:<newuser> /home/<newuser>/.ssh/authorized\_keys
      6. Repeat the above for each new admin user you create.
   4. The new users are now able to login. Since 2FA is required, when you send the password to each of the new users, also send the following instructions (HINT: fill in the username, Client IP address, and password for them with the correct values):
      1. Thanks for agreeing to help with the administration of our Indy Validator Node. Please login to the node, change your password, and setup Two Factor Authentication (2FA) using the following instructions:
         1. ssh -i <your private SSH key file> <username>@<Client IP Addr>
         2. Type in <password> for your password
         3. On successful login, type in ‘passwd’ to change your password on the Validator Node. Please use a unique password of sufficient length and store it in a secure place (i.e. a password manager).
         4. To set up 2FA, type in ‘google-authenticator’
            1. Answer ‘y’ to all questions asked during the setup
            2. Save the secret key, verification code, and scratch codes in a safe place. These are all for your user and can be used to login or to recover as needed.
         5. Install Google Authenticator, Duo, Authy, or other google-authenticator compatible app on your phone or device.
         6. On your 2FA phone app, add an account, and then scan the barcode or enter the 16 character secret key from step 4’s output.
         7. Log out and then log back in to check and make sure it worked!
   5. All of your secondary admin users should be setup now.
6. You can now begin the Indy Node installation.

# Troubleshooting and Tips

1. **Removing all resources** created during the instructions (In case you make an error and need to start over).
   1. Delete Resource Group (ONLY if you created a new one during the above steps and do so with the knowledge that ALL resources in the group will be deleted, not just the ones you added)
      1. IF you delete the whole resource group, I think all subsequent steps are unneeded.
      2. You can also select the resource group and remove all unneeded items from the Resource Group list. Use the order of deletion shown below for efficiency (some items must be removed before others can be removed)
      3. If you choose not to delete from the resource group, then please proceed to delete each of the following, one by one.
   2. Delete the VM
      1. Must stop the VM before you can delete it.
      2. From the VM’s Overview page click ‘Delete’ from the top menu.
      3. Wait for deletion notification before proceeding.
   3. Delete the Network Interfaces
      1. From the ‘All services’ view, select ‘Network interfaces’
      2. For each interface needing deleted, click on the three dots to the far right and select ‘Delete’
   4. Delete Virtual networks/subnet
      1. From the ‘All services’ view, select ‘Virtual Networks’’
      2. For each Network needing deleted, click on the name then do the following
         1. Click ‘Delete’ in the top bar
         2. NOTE: if there are still Interfaces in the list, double check that they are needing deleted, then delete them by clicking on them and following above instructions
   5. Delete the Network Security Groups
      1. From the ‘All services’ view, select ‘Network Security Groups’
      2. For each NSG needing deleted, click on the three dots to the far right and select ‘Delete’
   6. Delete the new IPs created
      1. From the ‘All services’ view, select ‘Public IP addresses’
      2. For each IP needing deleted, click on the three dots to the far right and select ‘Delete’
   7. Delete the Disks
      1. From the ‘All services’ view, select ‘Disks’’
      2. For each Disk needing deleted, click on the name then do the following
         1. Click ‘Delete’ in the top bar
   8. Delete Storage account
      1. From the ‘All services’ view, select ‘Storage accounts’’
      2. For each Storage account needing deleted (I only had 1), click on the name then do the following
         1. Click ‘Delete’ in the top bar
         2. Type the name of the account
         3. Click ‘Delete’
   9. Delete Recovery Services vault
      1. From the ‘All services’ view, select ‘Recovery Services vaults’’
      2. For each Recovery Services vault account needing deleted (I only had 1), click on the name then do the following
         1. Click ‘Delete’ in the top bar
         2. Click ‘Yes’ to confirm
         3. THIS DELETION FAILED for me. It said I had some soft deleted stuff in the vault. If you want to, you can wait 2 weeks to delete it, or you can just reuse the vault if you plan to reinstall like I did.
2. **Steps to manually create a NIC** (These steps were written when I thought they were required. They are NOT required, but some of them might be useful for maintenance or recovery.)
   1. Create a new Network Security Group
      1. From the Azure portal ‘Home’ click on the ‘Network Security Groups’ service (HINT: you might need to click “more services” then “all services” then you will find it in the ‘Networking’ section of that page)
      2. Click +Add to add a new Security Group
         1. Subscription - your choice
         2. Resource group - NODE (select what was used for the new VM just created)
         3. Name - your choice (example: ValidatorNode9701-nsg)
         4. Region - must match VM’s region
         5. Click Review + create
         6. Click ‘Create’
   2. Install the Azure CLI
      1. To add another NIC, since the online Azure Portal does not support it, you must install the Azure CLI or run it from Azure Cloud Shell (incurs small monthly cost for storage requirements)
         1. To install Azure CLI follow instructions here (I used MAC instructions)<https://docs.microsoft.com/en-us/cli/azure/install-azure-cli?view=azure-cli-latest>
         2. To run Azure Cloud Shell online go to<https://shell.azure.com/bash>
   3. Run the following command in the Azure CLI to create a new NIC in the NODE resource group with name NODE\_NIC and subnet NODE-vnet/default
      1. az network nic create -g NODE --vnet-name NODE-vnet --subnet default -n NODE\_NIC --network-security-group ValidatorNode9701-nsg
3. Manually Create a new IP address
   1. Create a new IP address. (Was “for the second NIC” so adjust those parts of the instructions if you are doing this for another reason)
      1. From the Azure portal Home click on the ‘+ Create a resource’
      2. Search for ‘public’ then select ‘Public IP address’
      3. Click ‘Create’
         1. IP Version - IPv4
         2. SKU - Standard
         3. Name - your choice (example - ValidatorNode9701-ip)
         4. Idle timeout - 4m
         5. DNS name label <blank>
         6. Subscription - your choice (the extra IP address adds an additional cost)
         7. Resource group - NODE (must match group used for the VM)
         8. Location - <same as your Node VM>
         9. Availability zone - Zone-redundant
         10. Click ‘Create’
      4. From Azure Portal Home select Resource Groups -> NODE (the resource group you have been using for the VM)
      5. In the list of resources seen, find and Click on the name of the new ip address that you just created as part of this section of the instructions.
      6. Record the public IP address shown for later use. This is the node\_ip.
4. Expand your OS disk size to 1TB (Decided to create a separate data disk for the main instructions, but if you want to do it this way, here are the old instructions)
   1. From your Validator Node VM Overview, Click ‘Disks’ on the left menu.
   2. Click on the name of your OS disk.
   3. On the new screen that appears, click “Configuration” in the left menu.
   4. Click in the Size entry and change it to 1000.
   5. Click “save”
   6. Return to your VM’s main page by clicking the ‘X’ in the upper right, and then select ‘Overview’ from the left menu.
5. The following route command might be sufficient to allow routing of traffic to your second NIC. You might also have to add in the instructions from the main content regarding the creation of the restrict-default-gw file.
   1. sudo route add -net 10.0.0.0 netmask 255.255.240.0 gw 0.0.0.0 eth1
   2. This method is untested, attempt at your own risk.
6. Helpful links (and acknowledgements to the authors of the content on these sites).
   1. <https://docs.microsoft.com/en-us/azure/virtual-machines/linux/multiple-nics?toc=/azure/virtual-network/toc.json#configure-guest-os-for-multiple-nics>
   2. <https://aws.amazon.com/premiumsupport/knowledge-center/ec2-ubuntu-secondary-network-interface/>
   3. <https://docs.microsoft.com/en-us/azure/virtual-machines/linux/add-disk>
   4. <https://www.digitalocean.com/community/tutorials/how-to-add-and-delete-users-on-ubuntu-16-04>
   5. <https://www.digitalocean.com/community/tutorials/how-to-set-up-ssh-keys-on-ubuntu-1604>
   6. <https://www.techrepublic.com/article/how-to-combine-ssh-key-authentication-and-two-factor-authentication-on-linux/>
   7. For adding a second user’s 2FA for them<https://www.linux.com/topic/desktop/how-set-2-factor-authentication-login-and-sudo/>