Instructions for Provisioning and Operating a Morpheus-Lumeric Node on a Virtual Host (Azure)

**Prerequisites**

1. **Azure Account Setup**
   * Go to <https://portal.azure.com> and log in or create an account.
   * Install the [Microsoft Authenticator](https://support.microsoft.com/en-us/account-billing/set-up-the-microsoft-authenticator-app-as-your-verification-method-33452159-6af9-438f-8f82-63ce94cf3d29#:~:text=Select%20Security%20info%20in%20the,list%2C%20and%20then%20select%20Add.) for Two-Factor Authentication (2FA)
2. **Azure Command-Line Interface (CLI) Session**
   * On your Azure portal session, launch **Azure CloudShel**l

NOTE: When starting the CLI, select the **Bash** option

* + Instructions to Start Azure CLI session: <https://learn.microsoft.com/en-us/azure/cloud-shell/get-started/classic?tabs=azurecli>

1. **Azure CLI Configuration** - Run these commands to get parameter information required for cloud host provisioning and configuration by Terraform:
   1. mkdir projects/mln
   2. git clone <https://github.com/indrgun/terraform-azure-linux-vm.git>
   3. cd terraform-azure-linux-vm
   4. chmod +x mln\_env\_config.sh
   5. ./mln\_env\_config.sh
2. Generate an SSH key pair using this command. Accept all default options.

ssh-keygen -t rsa -m PEM -b 4096 -C mln\_azure@myserver

**Cloud Host Provisioning and Configuration**

1. **Get the Terraform repository**
   1. Create a local projects directory and close the repository:
      1. mkdir /home/${USER}/projects/mln/code
      2. cd /home/${USER}/projects/mln/code
      3. git clone <https://github.com/indrgun/terraform-azure-linux-vm.git>
      4. cd terraform-azure-linux-vm
2. **Provision Cloud Host on Azure**
   1. Run the following commands to provision the Linux VM in VPC, Subnet, with storage on Azure.
      1. terraform init
      2. terraform plan -var "name\_prefix=linux" -var "hostname=linux$(echo $RANDObM)" -var "ssh\_public\_key=~/.ssh/id\_rsa.pub"
      3. terraform apply -var "name\_prefix=linux" -var "hostname=linux$(echo $RANDOM)" -var "ssh\_public\_key=~/.ssh/id\_rsa.pub"

Type "yes" to confirm the apply.

The command “terraform apply” will provide output of the **admin username (admin\_username)** and the public-accessible **FQDN hostname (vm\_fqdn)** of the VM. Document this name for secure logon. An example hostname will result as follows:

i.e.

admin\_username = “vmadmin”

vm\_fqdn = linux37105.westus.cloudapp.azure.com

1. **Connect to the VM**
   1. SSH into the VM using the **admin username** and the public-accessible **FQDN hostname** of the VM discovered in step #7, as follows:
   * ssh vmadmin@<linux#####.westus.cloudapp.azure.com> -i $HOME/.ssh/id\_rsa

**Morpheus-Lumerin Node Setup**

1. **Docker Installation & Setup** 
   1. Install Docker and Docker Compose prerequisites on the VM:
      1. mkdir ${HOME}/projects & cd ${HOME}/projects
      2. git clone <https://github.com/indrgun/terraform-azure-linux-vm.git>
      3. cd terraform-azure-linux-vm
      4. chmod +x mln\_docker\_setup.sh
      5. ./mln\_docker\_setup.sh
2. **Verify Docker Installation and config admin groups**
   1. Verify that Docker is installed and online. From the same directory used in #8:
      1. chmod +x mln\_docker\_verify\_config.sh
      2. ./mln\_docker\_verify\_config.sh
   2. Log off the VM
      1. exit
3. **Node Runtime**
   1. SSH into the VM. Run the following commands to start the Morpheus-Lumerin-Node on the VM host:
      * mkdir projects && cd projects
      * git clone <https://github.com/MorpheusAIs/Morpheus-Lumerin-Node.git>
      * cd Morpheus-Lumerin-Node
      * mv proxy-router/.env.example proxy-router/.env
      * docker compose up -d
4. **Ollama LLM Runtime Verification**
   1. Verify the Ollama container started and check the port number it is using:
      * ps -ef | grep "ollama" | grep -v "grep"
      * sudo lsof -i -P -n | grep LISTEN | grep docker
      * docker ps
5. **Instance Teardown**
   1. Delete the entire setup by executing the following command:
   * terraform destroy -var "name\_prefix=linux" -var "hostname=linux$(echo $RANDOM)" -var "ssh\_public\_key=~/.ssh/id\_rsa.pub"