

Nilavembu Herbs provides a safer alternative to modern medicine wherever possible and to offer simple, effective and safe remedies for common problems. In this project, its application deployment using Microsoft Azure Cloud PaaS and IaaS solution is explored and implemented.

Tejinder Singh

**2021**

**Microsoft Azure Project**

Table of Contents

[Case Study: 3](#_Toc76988427)

[Business Requirement 3](#_Toc76988428)

[Technical Requirement: 3](#_Toc76988429)

[Creating Jump Server in East US Region 4](#_Toc76988430)

[Create jump server from Azure portal 4](#_Toc76988431)

[Installing terraform in jump Server 5](#_Toc76988432)

[Pre-requisite Tools 7](#_Toc76988433)

[Vi main.tf 7](#_Toc76988434)

[Vi terraform.tfvars 13](#_Toc76988435)

[Vi variables.tf 14](#_Toc76988436)

[Due to Limitation in SEA (South East Asia), Changing Region to Central US 15](#_Toc76988437)

[Successful Deployment using Terraform apply 15](#_Toc76988438)

[Review of Network Diagram after Terraform apply 17](#_Toc76988439)

[Details of Resources after Running Terraform Apply 17](#_Toc76988440)

[Load Balancer 18](#_Toc76988441)

[After infrastructure built by terraform, Preparing Ansible for WebServer Installation 18](#_Toc76988442)

[Installing Ansible Software 19](#_Toc76988443)

[Creating Directory for Ansible Scripts 21](#_Toc76988444)

[Creating hosts file for Ansible 22](#_Toc76988445)

[Creating Yaml file for IIS installation using Ansible 22](#_Toc76988446)

[Running Ansible process to install IIS on host VM machines 22](#_Toc76988447)

[Accessing Load Balancer IP : http://40.122.47.65/ 23](#_Toc76988448)

[Storage Account Creation 24](#_Toc76988449)

[Storage Account Creation 24](#_Toc76988450)

[Allowing Access to select IP to Storage Account 24](#_Toc76988451)

[Creation of VMADMIN user id to manage all resources 26](#_Toc76988452)

[Create vmadmin user who can manage all VM in the subscription 26](#_Toc76988453)

[Create Backup\_admin user who can manage backup only in EUS servers in EURG 27](#_Toc76988454)

[Backup Admin has access to backup Vault 28](#_Toc76988455)

[Requirement Traceability Matrix along with Test Results 28](#_Toc76988456)

[Scripts 29](#_Toc76988457)

Nilavembu Herbs:

PaaS, IaaS Implementation using MS Azure

# Case Study:

Nilavembu Herbs provides a safer alternative to modern medicine wherever possible and to offer simple, effective and safe remedies for common problems. spreads awareness about the medicinal uses of these natural and safe herbs all over the world and to make it easily available through their online store for all those who want to enjoy its benefits

## Business Requirement

1. A low cost solution based on demand of dynamic business conditions.
2. As the business expands across EastUS and SEA, they would like to have their DataCenter virtualised using cloud computing.
3. Critical Data should be made available in case of disaster.
4. Sales manager should access his resource from windows explorer.

## Technical Requirement:

1. SEA region
   1. 2 web servers with 99.95% high availability
   2. These web services has to be utilised with proper balance with client affinity with Public IP
   3. Selected web servers should be reachable via RDP from internet
   4. A jump port should accessible from internet to upload contents to web servers.
   5. Protect web server traffic restricted to allowed based on ip addresses which will be updated as warranted
   6. Enable backup for WebServers
   7. Have alert generated in case of 80% above cpu usage
2. EastUS
   1. EastUS server (Server11) should be accessible from internet via public IP
   2. Establish secure Connection to SEA-EUS Azure sites
   3. All servers should be reachable with internal ip addresses
3. STORAGE Requirement
   1. EUS based resources should provide data resiliency in case of azure datacentre failure.
   2. The storage should be accessible by applications with secure access. provide access urls and keys.
   3. Sales manager should access his resource from windows explorer.
   4. SEA data resources must provide high resiliency in case of even multiple azure data center failures
4. Azure resource
   1. Create Vmadmin user who can manage all VM in the subscription
   2. Create Backup\_admin user who can manage backup only in EUS servers in EURG

# Creating Jump Server in East US Region

|  |
| --- |
| Create jump server from Azure portal |
|  |
|  |
|  |
| Installing terraform in jump Server |
| installing terraform  step 1  sudo apt-get update && sudo apt-get install -y gnupg software-properties-common curl  step 2  curl -fsSL https://apt.releases.hashicorp.com/gpg | sudo apt-key add -  step 3  sudo apt-add-repository "deb [arch=amd64] https://apt.releases.hashicorp.com $(lsb\_release -cs) main"  step 4  sudo apt-get update && sudo apt-get install terraform |
| curl -sL https://aka.ms/InstallAzureCLIDeb | sudo bash |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |

|  |
| --- |
| Pre-requisite Tools |
|  |
|  |
|  |
|  |
| Vi main.tf |
| # Configure the Azure provider  terraform {  required\_providers {  azurerm = {  source = "hashicorp/azurerm"  version = "~> 2.65"  }  }  required\_version = ">= 0.14.9"  }  provider "azurerm" {  features {}  }  ## Create an Azure resource group ##  resource "azurerm\_resource\_group" "nilavembuRG" {  name = var.resource\_group  location = var.location  }  ## Create an availability set ##  resource "azurerm\_availability\_set" "nilavembu-as" {  name = "nilavembu-as"  location = azurerm\_resource\_group.nilavembuRG.location  resource\_group\_name = azurerm\_resource\_group.nilavembuRG.name  }  ## Create an Azure NSG ##  resource "azurerm\_network\_security\_group" "nilavembunsg" {  name = "nsg"  location = azurerm\_resource\_group.nilavembuRG.location  resource\_group\_name = azurerm\_resource\_group.nilavembuRG.name  ## Create a rule to allow Ansible to connect to each VM ##  security\_rule {  name = "allowWinRm"  priority = 101  direction = "Inbound"  access = "Allow"  protocol = "Tcp"  source\_port\_range = "\*"  destination\_port\_range = "5986"  source\_address\_prefix = var.cloud\_shell\_source  destination\_address\_prefix = "\*"  }        ## Create a rule to allow your local machine ##  security\_rule {  name = "allowWebDeploy"  priority = 102  direction = "Inbound"  access = "Allow"  protocol = "Tcp"  source\_port\_range = "\*"  destination\_port\_range = "8172"  source\_address\_prefix = var.management\_ip  destination\_address\_prefix = "\*"  }        ## Create a rule to allow web clients to connect to the web app ##  security\_rule {  name = "allowPublicWeb"  priority = 103  direction = "Inbound"  access = "Allow"  protocol = "Tcp"  source\_port\_range = "\*"  destination\_port\_range = "80"  source\_address\_prefix = "\*"  destination\_address\_prefix = "\*"  }    ## Create a rule to allow RDP to the VMs ##  security\_rule {  name = "allowRDP"  priority = 104  direction = "Inbound"  access = "Allow"  protocol = "Tcp"  source\_port\_range = "\*"  destination\_port\_range = "3389"  source\_address\_prefix = var.management\_ip  destination\_address\_prefix = "\*"  }    ## Create a rule to allow SFTP to the VMs ##  security\_rule {  name = "allowSFTP"  priority = 105  direction = "Inbound"  access = "Allow"  protocol = "SFTP"  source\_port\_range = "\*"  destination\_port\_range = "22"  source\_address\_prefix = var.cloud\_shell\_source  destination\_address\_prefix = "\*"  }    }  ## Create Nilavembu Corporate vNet ##  resource "azurerm\_virtual\_network" "nilavembuCorpNet" {  name = "nilavembu-network"  address\_space = ["10.0.0.0/16"]  location = azurerm\_resource\_group.nilavembuRG.location  resource\_group\_name = azurerm\_resource\_group.nilavembuRG.name  }  ## Create a web subnet inside the vNet ##  resource "azurerm\_subnet" "internal" {  name = "internal"  resource\_group\_name = azurerm\_resource\_group.nilavembuRG.name  virtual\_network\_name = azurerm\_virtual\_network.nilavembuCorpNet.name  address\_prefixes = ["10.0.2.0/24"]  depends\_on = [  azurerm\_virtual\_network.nilavembuCorpNet  ]  }  ## Create public IP to assign to the load balancer ##  resource "azurerm\_public\_ip" "lbIp" {  name = "publicLbIp"  location = azurerm\_resource\_group.nilavembuRG.location  resource\_group\_name = azurerm\_resource\_group.nilavembuRG.name  allocation\_method = "Static"  }  ## Assign public IPs for each VM for Ansible to connect to and to deploy the web app ##  resource "azurerm\_public\_ip" "vmIps" {  count = 2  name = "publicVmIp-${count.index}"  location = azurerm\_resource\_group.nilavembuRG.location  resource\_group\_name = azurerm\_resource\_group.nilavembuRG.name  allocation\_method = "Dynamic"  domain\_name\_label = "${var.domain\_name\_prefix}-${count.index}"  }  ## Create a vNic for each VM ##    resource "azurerm\_network\_interface" "main" {  count = 2  name = "nilavembu-nic-${count.index}"  location = azurerm\_resource\_group.nilavembuRG.location  resource\_group\_name = azurerm\_resource\_group.nilavembuRG.name    ## Simple ip configuration for each vNic  ip\_configuration {  name = "ip\_config"  subnet\_id = azurerm\_subnet.internal.id  private\_ip\_address\_allocation = "Dynamic"  public\_ip\_address\_id = azurerm\_public\_ip.vmIps[count.index].id  }    ## Ensure the subnet is created first before creating these vNics.  depends\_on = [  azurerm\_subnet.internal  ]  }  ## Apply the NSG to each of the VMs' NICs ##  resource "azurerm\_network\_interface\_security\_group\_association" "nsg" {  count = 2  network\_interface\_id = azurerm\_network\_interface.main[count.index].id  network\_security\_group\_id = azurerm\_network\_security\_group.nilavembunsg.id  }  ## Create the load balancer with a frontend configuration using the public ##  resource "azurerm\_lb" "LB" {  name = "nobsloadbalancer"  location = azurerm\_resource\_group.nilavembuRG.location  resource\_group\_name = azurerm\_resource\_group.nilavembuRG.name  frontend\_ip\_configuration {  name = "lb\_frontend"  public\_ip\_address\_id = azurerm\_public\_ip.lbIp.id  }  }  ## Create backend address pool holding both VMs ##  resource "azurerm\_lb\_backend\_address\_pool" "be\_pool" {  resource\_group\_name = azurerm\_resource\_group.nilavembuRG.name  loadbalancer\_id = azurerm\_lb.LB.id  name = "BackEndAddressPool"  }  ## Assign both vNics on the VMs to the backend address pool ##  resource "azurerm\_network\_interface\_backend\_address\_pool\_association" "be\_assoc" {  count = 2  network\_interface\_id = azurerm\_network\_interface.main[count.index].id  ip\_configuration\_name = "ip\_config"  backend\_address\_pool\_id = azurerm\_lb\_backend\_address\_pool.be\_pool.id  }  ## Create a health probe which will periodically check for an open port 80 ##  resource "azurerm\_lb\_probe" "lbprobe" {  resource\_group\_name = azurerm\_resource\_group.nilavembuRG.name  loadbalancer\_id = azurerm\_lb.LB.id  name = "http-running-probe"  port = 80  }  ## Create a rule on the load balancer to forward all incoming traffic on port 80 using above health probe ##  resource "azurerm\_lb\_rule" "lbrule" {  resource\_group\_name = azurerm\_resource\_group.nilavembuRG.name  loadbalancer\_id = azurerm\_lb.LB.id  name = "LBRule"  probe\_id = azurerm\_lb\_probe.lbprobe.id  protocol = "Tcp"  frontend\_port = 80  backend\_port = 80  backend\_address\_pool\_id = azurerm\_lb\_backend\_address\_pool.be\_pool.id  frontend\_ip\_configuration\_name = "lb\_frontend"  }  ## Create the two Windows VMs associating the vNIcs created earlier ##  resource "azurerm\_windows\_virtual\_machine" "nilavembuVMs" {  count = 2  name = "nilavembuvm-${count.index}"  location = var.location  resource\_group\_name = azurerm\_resource\_group.nilavembuRG.name  size = "Standard\_DS1\_v2"  network\_interface\_ids = [azurerm\_network\_interface.main[count.index].id]  availability\_set\_id = azurerm\_availability\_set.nilavembu-as.id  computer\_name = "nilavembuvm-${count.index}"  admin\_username = "vmadmin"  admin\_password = "Vmadmin!2345"    source\_image\_reference {  publisher = "MicrosoftWindowsServer"  offer = "WindowsServer"  sku = "2019-Datacenter"  version = "latest"  }  os\_disk {  caching = "ReadWrite"  storage\_account\_type = "Standard\_LRS"  }  depends\_on = [  azurerm\_network\_interface.main  ]  }  output "VMIps" {  value = azurerm\_public\_ip.vmIps.\*.ip\_address  }  ## Get the load balancer's public IP address ##  output "Load\_Balancer\_IP" {  value = azurerm\_public\_ip.lbIp.ip\_address  } |
|  |
| Vi terraform.tfvars |
| location = "Central US"  resource\_group = "nilavembu-rg-corp"  cloud\_shell\_source = "20.106.152.225"  domain\_name\_prefix = "nilavembu"  management\_ip = "106.212.145.12" |
|  |
| Vi variables.tf |
|  |
|  |
| terraform apply -var-file="terraform.tfvars" |
|  |
|  |
| Error in Previous page shows, that South East Asia Region does not have 3 fault domains. |
|  |
| Due to Limitation in SEA (South East Asia), Changing Region to Central US As South East Asia as only 2 fault domains so changing the Location from South East Asia to **Central US** |
|  |
|  |
|  |

## Successful Deployment using Terraform apply

|  |
| --- |
| terraform apply -var-file="terraform.tfvars" |
|  |
|  |
|  |
|  |
| Review of Network Diagram after Terraform apply |
|  |
| Details of Resources after Running Terraform Apply |
|  |
|  |
|  |
| Load Balancer |
|  |
|  |
|  |

# After infrastructure built by terraform, Preparing Ansible for WebServer Installation

|  |
| --- |
|  |
|  |
| Installing Ansible Software |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
| Creating Directory for Ansible Scripts |
|  |
| Creating hosts file for Ansible |
|  |
| Creating Yaml file for IIS installation using Ansible |
|  |
| Running Ansible process to install IIS on host VM machines |
| ansible-playbook iis.yml -i hosts |
|  |

|  |
| --- |
| Accessing Load Balancer IP : http://40.122.47.65/ |
|  |

# Storage Account Creation

|  |
| --- |
| Storage Account Creation |
|  |
|  |
| Allowing Access to select IP to Storage Account |
|  |
|  |
|  |
|  |
|  |
|  |
|  |

|  |
| --- |
|  |
|  |
| Copy the content to mount\_fs.ps1 file and run in powershell |
| $connectTestResult = Test-NetConnection -ComputerName nilavembusa.file.core.windows.net -Port 445  if ($connectTestResult.TcpTestSucceeded) {  # Save the password so the drive will persist on reboot  cmd.exe /C "cmdkey /add:`"nilavembusa.file.core.windows.net`" /user:`"localhost\nilavembusa`" /pass:`"4TyfQ9cI0/aGdzpytDA8ZD9Ara78atHP3fxL8A/SpwtHyu4FTB/Ur9tLnwMNb6ZRLZ09W40/lt1dYufEgs+o0A==`""  # Mount the drive  New-PSDrive -Name Z -PSProvider FileSystem -Root "\\nilavembusa.file.core.windows.net\nilavembufs" -Persist  } else {  Write-Error -Message "Unable to reach the Azure storage account via port 445. Check to make sure your organization or ISP is not blocking port 445, or use Azure P2S VPN, Azure S2S VPN, or Express Route to tunnel SMB traffic over a different port."  } |
|  |
|  |

|  |
| --- |
| Creation of VMADMIN user id to manage all resources |
|  |
|  |
|  |
| Create vmadmin user who can manage all VM in the subscription |
|  |
|  |

|  |
| --- |
| Create Backup\_admin user who can manage backup only in EUS servers in EURG |
|  |
|  |
|  |
|  |
|  |
| Backup Admin has access to backup Vault |
|  |

# Requirement Traceability Matrix along with Test Results

Pls refer excel file “Nilavembu-Requirement Tracebility Matrix.xls”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sr.No. | ReqId | SubSection | Requirement Description | Technical Specification |
| 1 | R001 | ~~SEA~~ (Central US) | SEA-2 web servers with 99.95% high availability | [Two Web Server Details](file:///C:\Users\Tejinder\Desktop\MS%20Azure%20Tutorial\AzProject\Requirement%20Tracebility%20Matrix.xlsx#'Two Web Server Details'!A1) |
| 2 | R002 | ~~SEA~~ (Central US) | SEA-These web services has to be utilised with proper balance with client affinity with Public IP | [Mapping to LB](file:///C:\Users\Tejinder\Desktop\MS%20Azure%20Tutorial\AzProject\Requirement%20Tracebility%20Matrix.xlsx#'Mapping to LB'!A1) |
| 3 | R003 | ~~SEA~~ (Central US) | SEA-Selected web servers should be reachable via RDP from internet | [Allow RDP Access](file:///C:\Users\Tejinder\Desktop\MS%20Azure%20Tutorial\AzProject\Requirement%20Tracebility%20Matrix.xlsx#'Allow RDP Access'!A1) |
| 4 | R004 | ~~SEA~~ (Central US) | SEA-A jump port should accessible from internet to upload contents to web servers. | [jump vm access from internet](file:///C:\Users\Tejinder\Desktop\MS%20Azure%20Tutorial\AzProject\Requirement%20Tracebility%20Matrix.xlsx#'jump vm access from internet'!A1) |
| 5 | R005 | ~~SEA~~ (Central US) | SEA-Protect web server traffic restricted to allowed based on ip addresses which will be updated as warranted | [Deny All other Traffic](file:///C:\Users\Tejinder\Desktop\MS%20Azure%20Tutorial\AzProject\Requirement%20Tracebility%20Matrix.xlsx#'Deny All other Traffic'!A1) |
| 6 | R006 | ~~SEA~~ (Central US) | SEA-Enable backup for WebServers | [Backup Status](file:///C:\Users\Tejinder\Desktop\MS%20Azure%20Tutorial\AzProject\Requirement%20Tracebility%20Matrix.xlsx#'Backup Status'!A1) |
| 7 | R007 | ~~SEA~~ (Central US) | SEA-Have alert generated in case of 80% above cpu usage | [Alert Config](file:///C:\Users\Tejinder\Desktop\MS%20Azure%20Tutorial\AzProject\Requirement%20Tracebility%20Matrix.xlsx#'Alert Config'!A1) |
| 8 | R008 | EastUS | EastUS-EastUS server (Server11) should be accessible from internet via public IP | [Accessible from Internet](file:///C:\Users\Tejinder\Desktop\MS%20Azure%20Tutorial\AzProject\Requirement%20Tracebility%20Matrix.xlsx#'Accessible from Internet'!A1) |
| 9 | R009 | EastUS | EastUS-Establish secure Connection to SEA-EUS Azure sites | [Secured Connection](file:///C:\Users\Tejinder\Desktop\MS%20Azure%20Tutorial\AzProject\Requirement%20Tracebility%20Matrix.xlsx#'Secured Connection'!A1) |
| 10 | R010 | EastUS | EastUS-All servers should be reachable with internal ip addresses | [Reachable using internal IP](file:///C:\Users\Tejinder\Desktop\MS%20Azure%20Tutorial\AzProject\Requirement%20Tracebility%20Matrix.xlsx#'Reachable using internal IP'!A1) |
| 11 | R011 | Storage | EUS based resources should provide data resiliency in case of azure datacentre failure. | [Storage Account](file:///C:\Users\Tejinder\Desktop\MS%20Azure%20Tutorial\AzProject\Requirement%20Tracebility%20Matrix.xlsx#'Storage Account'!A1) |
| 12 | R012 | Storage | The storage should be accessible by applications with secure access. provide access urls and keys. | [Storage Account URLs](file:///C:\Users\Tejinder\Desktop\MS%20Azure%20Tutorial\AzProject\Requirement%20Tracebility%20Matrix.xlsx#'Storage Account URLs'!A1) |
| 13 | R013 | Storage | Sales manager should access his resource from windows explorer. | [Sales Manager Access](file:///C:\Users\Tejinder\Desktop\MS%20Azure%20Tutorial\AzProject\Requirement%20Tracebility%20Matrix.xlsx#'Sales Manager Access'!A1) |
| 14 | R014 | Storage | SEA data resources must provide high resiliency in case of even multiple azure data center failures | [Data Resiliency](file:///C:\Users\Tejinder\Desktop\MS%20Azure%20Tutorial\AzProject\Requirement%20Tracebility%20Matrix.xlsx#'Data Resiliency'!A1) |
| 15 | R015 | AzureResources | Create vmadmin user who can manage all VM in the subscription | [VMADMIN Access](file:///C:\Users\Tejinder\Desktop\MS%20Azure%20Tutorial\AzProject\Requirement%20Tracebility%20Matrix.xlsx#'VMADMIN Access'!A1) |
| 16 | R016 | AzureResources | Create Backup\_admin user who can manage backup only in EUS servers in EURG | [Backup Admin Access](file:///C:\Users\Tejinder\Desktop\MS%20Azure%20Tutorial\AzProject\Requirement%20Tracebility%20Matrix.xlsx#'Backup Admin Access'!A1) |

# Scripts

