Create Windows VMs behind a load balancer

- Overview
- Problem
- Solution
- Related articles

Overview

Platform:	Azure
Owner of this SOP:	Fully Managed POD A
Cloud Services:	Instance Management

Problem

When requesters are asking for creating Windows VM behind the load balancer.

tester	Reviewer
@ Ramkumar Samudram (Deactivated)	

Solution

- 1. Clone the Blue Print 8 repo https://gitlab.com/bhp-cloudfactory/azure-blueprints/azr-iaas-single-tier into a your repository under new project e.g. https://gitlab.com/bhp-cloudfactory/tooling-foundations/fm-sops/winvmwithlb

 2. Change all the variables in the tfvars file as per your requirement. Please go through the blue print read me file for more details

```
WINVMWITHLB
                     winvmwithlb > env_vars > * npe.auto.tfvars
 winvmwithlb
                                                                                                    > datado

√ .idea

                                            availability set name
                                                                         = "aue1winvmavs01"
  gitignore
                                                                         = "aue1winvmlb01"
                                            load balancer name
  > doc
                                                                         = "it-aue1-npe-arg-fmpodawin2019"
                                            resource_group_name
                                                                         = "Australia East"
                                                                                                           #Azure location
 env vars
                                            location
                                            network_resource_group_name = "it-aue1-npe-arg-network"
                                                                                                                   # Name
  mpe.auto.tfvars
                                            subnet_name
                                                                         = "sub-iaas" # THE SUBNET NAME WHERE THIS VM IS
  > scripts
                                                                         = "it-aue1-npe-vnt-10.125.121.0" # THE VNET NAM
                                            virtual_network_name
  > templates
                                                                         = "it-aue1-npe-arg-backup"
                                            rsv_resource_group_name
  ! .gitlab-ci.yml
                                            recovery_vault_name
                                                                         = "it-aue1-npe-rsv-grs"
  ! .pre-commit-config.yaml
                                            backup_policy_name
                                                                         = "AU Non-Production VM"
  ! .terraform-docs.yaml
 ata.tf
                                            secondary availability set name = "ausewinvmavs01"

™ locals.tf

                                            secondary load balancer name
                                                                               = "ausewinvmlb01"
 meta.tf
                                            secondary_resource_group_name
                                                                               = "it-aue1-npe-arg-fmpodawin2019-secondary
 w outputs.tf
                                            secondary_location
                                                                               = "Australia Southeast"
                                                                                                                #Azure loca
 primary_site.tf
                                            secondary_subnet_name
                                                                               = "sub-iaas" # THE SUBNET NAME WHERE THIS
 y providers.tf
                                            secondary_virtual_network_name
                                                                               = "it-ause-npe-vnt-10.125.201.0" # THE VNE
 (i) README.md
                                            secondary_rsv_resource_group_name = "it-aue1-npe-arg-backup"
                                            secondary recovery vault name
                                                                               = "it-ause-npe-rsv-grs"
 resource-group.tf
                                            secondary_backup_policy_name
                                                                               = "Non-Production-VM"
 y secondary_site.tf
 variables.tf
                                            #datadog enabled = 0
                                                                                        = "YOUR SHARED SUBSCRIPITON RESOUR
                                            #shared landscape resource group
                                            #shared_landscape_secondary_resource_group = "YOUR SHARED SUBSCRIPITON SECOND
                                            domain enabled = true # mark true if VMs need to join Domain
                                            enable_backup = false # Backup ### Uncomment if backup needs to be disabled
                                                                                        = "Basic"
                                            frontend_ip_primary ={
OUTLINE
                                               "sample1" = {
> TIMELINE
                                                               = "frontendip001"
                                                                                                                   ### NAME
                                                name
```

3. Change the variables in the env.sh file to match your environment

```
/ WINVMWITHLB
                        다 다 강 회 winvmwithlb > scripts > 를 env.sh
winymwithlb
                                                                                                                  > datado
                                                                                                                                              Aa Abl ■* No results ↑

√ idea

   gitignore
                                                  export FNVTRONMENT="nne"

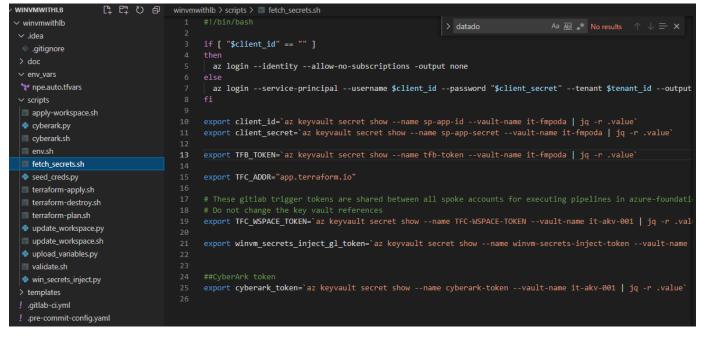
✓ env_vars

  npe.auto.tfvars
                                                  export SUBSCRIPTION ID="01818878-c294-4617-a15a-8b7934a433c4"
                                                  # NPE
  apply-workspace.sh
                                                  export TFB ORG="BHPEG-Dev"
  cyberark.py
 g cyberark.sh
   env.sh
                                                  #export TFB ORG="BHPEG"
  fetch secrets.sh
  seed creds.py
                                                  export CLOUD_SERVICE_PROVIDER="m" # 'a' for Amazon Web Service, 'm' for Microsoft Azure
export ASSET="it" # it (IT), bma (BMA), bmc (BMC), nswec (NSWEC), niw (Nickel West), od (Olympic Dam

    ■ terraform-apply.sh

  terraform-destroy.sh
  terraform-plan.sh
  update workspace.py
  update_workspace.sh
  upload variables.py
   validate.sh
  win_secrets_inject.py
                                                  export REGION="aue1"
  > templates
 ! .gitlab-ci.vml
                                                  export ENV=${ENVIRONMENT}
 ! .pre-commit-config.yaml
                                                  export TFC_ADDR="app.terraform.io"
 ! .terraform-docs.yaml
 🚏 data.tf
                                                  export TFC_WORKSPACE_NAME="a_azr_toolingdev_npe_fmpodalb_01818878-c294-4617-a15a-8b7934a433c4"
```

4. in fetch_secrets.sh update your key vault name in the required field



- 5. Commit and push your changes
- 6. Verify that the pipeline executes as expected and that a terraform workspace is created and run
- 7. Verify the terraform plan in your TF workspace
- 8. If you are satisfied with the plan output, Initiate the apply stage in pipeline

 a_azr_toolingdev_npe_fmpodalb_01818878-c294-4617-a15a-8b7934a433c4

 No workspace description available. Add workspace description.

 Overview Runs States Variables Settings V

 Current Run

 Queued from GitLab CI/CD Pipeline CURRENT

 Queued from GitLab CI/CD Pipeline CURRENT

 VIpdated
 5 hours ago
 1.1.2

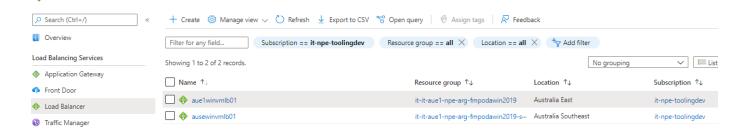
 Actions V

 Applied

5 hours ago

9. Logon to Azure portal and verify that whether the Load balancers are created as per requirement

-24mgUSYJoDMUxLtg | API Integration triggered via API



Related articles

AWS VPC Endpoints configuration change

🏠 Load balancing | Load Balancer 🗳 …

- Renewal / Upload SSL certificate for existing Application
- Monitoring GitLab Access Audit Report

- Azure Storage Account Blueprint SOP BPAZR022
- Azure App Service FTP Credential Sharing and Reset Procedure