

Norbert Obiekwe. 24 Dec 2020

MNP Infrastructure – Web Server Iac Build Tech Challenge.

Project Requirement Considerations:

Here are some factors/ requirements I considered before implementing this Iac project;

- The Communication and Network requirement
- VM Name
- Location of VM and Data Residency
- Size of VM
- Pricing model
- VM Storage
- Selecting OS
- Technology: Azure DevOps (CICD), GitHub, Visual studio, ARM Template.
- Automation services: Microsoft Endpoint Configuration Manger and Azure Automation for update management

1. The Communication and Network planning Requirement

For Comms between AZ resources, traffic routes and segmentation considerations, design and planning out the Network layout are generally first step before the embarking on the actual infrastructure deployment. VNets are used in Azure to provide private connectivity between Azure VMs and other AZ services. VMs and services that are part of the same virtual network can access one another.

For this project challenge, the Web server would be interacting or accessing other Azure services. As such, it make sense to have these services on the same Virtual Network. Nevertheless, the network can also be configured to allow access to the external service, including on- premises servers through point to site VPN gateway. As such, its very import to plan for the network configuration as the net address and subnets are not very flexible to change once they are set up.

Another consideration is planning and set up **Network security group** (NSG) to secure the network. With NSG, I can control the traffic flow to and from subnets and to and from the web server.

2. The VM Name:

Once implemented, VM names are not trivial to change. For best practices, I choose VM name that would make it easier to identify what the VM does. A general good convention is to include as much meaningful info such as env, location, and instance.

MNP-devcc-webvm -representing first MNP dev webserver hosted in Canada Central Region

3. Resources Deployed via Iac - ARM Template

To meet the project requirement, the following resources would be created/deployed using the ARM Template;

- VM that provides the CPU and memory
- Storage Account to hold virtual hard disks, boot diagnostics.
- Virtual Network to connect VM to other Az services.
- Network Interface to communicate with the VNet.
- Network Security Group to secure the network traffic.
- Public ip address to access the VM.
- Custom Script for automated install of IIS.

4. VM/Data Residency Requirements:

As per this project data residency requirement, the webserver would be located in Canada. Data center regions offered by Azure in Canada include Canada Central and Canada East region.

To meet this project's data residency requirement, the webserver and resources would be hosted in **Canada Central** region

5. VM Size:

As per this project requirement workload and capability to run on a free tier sub, I choose the **general purpose – A1 v2 series** VM size with 2 GiB Ram and temp storage of 10 GiB. The Av2 standard is the latest gen of A-series VMs with similar CPU performance but more Ram per VCPU and faster disks. Gen purpose Av2 are designed to have a balanced CPU – to memory ration. It's ideal for testing, development small to medium database and low to medium traffic web servers. Also, when I compare the Av2 to other series, it is the most cost effective option for this project, with cost starting at CAD\$0.0551/hr-PAY AS YOU GO OPTION.

More so, the Av2 Virtual Machines include load balancing and auto scaling at no additional charge, very suitable as per this Infrastructure project requirement.

Moreover, putting VM management and bootstrapping into consideration, the VM size, location, and disk can be updated and automated via the ARM template, Azure cli or Az PowerShell cmdlet. The VM size can be changed while the VM is running.

6. Pricing Model:

To ensure this Infrastructure project remains within budget, I considered some cost implications such as the compute cost and storage cost. To keep cost at minimum, I choose the **Pay as you go** payment option for compute cost for this project. With Pay as you go, there's no long term cost commitment or upfront payment. Compute capacity are calculated by the second. I can increase or decrease computes capacity on demand as well as start or stop at any time. This payment option is ideal for doing quick test, or developing App in a VM, just as the project entails.

7. Storage Consideration:

To meet the project requirement of less manual management's process and using best practices, I choose the **managed disk storage model- and Standard SSD** as the OS disk type for the VM storage. With managed disks, Azure creates and manages both the disk and the storage. The Cloud Architect don't have to worry about storage account limits, which makes managed disks easier to scale out and eliminates manual process of

managing own storage account that are used to host the VHDs that correspond to the VM disks.

8. OS Consideration:

I selected the windows platform for this project, as IIS would be easier to manage and bootstrap on a **Windows 2019 server datacenter**.

9. Server Update Management and Monitoring Consideration:

For ease of maintenance and VMs update management, I choose to implement server update requirement using the **Microsoft Endpoint configuration manager**. This endpoint would allow for server update tracking and for Cloud Architect to take action as required. I also enable update management on the VM server. With this, cloud architect can manage updates and patches for the server. With update management enabled from either the Az automation account or VM Pane in portal, Cloud Architect can assess the status of available updates, scheduled installation and review deployments results to verify updates were applied successfully. A scheduled recurring update can also be implemented. With scheduled deployments, I can define what target VMs receive the updates, either by explicitly specifying VMs or by selecting computer group that's based on log searches of a specific set of VMs. Also, update management can also be configured for single VM from the VMs pane in the Azure portal.

For more in-depth reporting and analytics, another consideration is to connect the VM server to Log analytics workspace. Enabling update management on the VM would need a **log analytics works space and a runbook automation account**.

To better understand the performance, operation, health of the server and get alerted on attack vectors, I implement server monitoring by enabling metric and **boot diagnostics** on the VM via ARM template declaration. Some metrics to monitor include the Avg server CPU usage, Disks, OS disk and Network performance.

10. Availability, backups And Fail Over Consideration

To ensure high availability of the Web server, the VM server can be placed in an availability set. When VMs are placed in availability set, Azure guarantees to spread them across Fault and Update Domains. MS offers 99.95% external connectivity SLA for multiple-instance VMs deployed in an availability set.

On grand scheme of things, Availability set, backup and site recovery services are best practices and nice to have services for a robust Azure Infrastructure redundancy, however, these services were not implemented for this project.

11. Reoccurring VM Server management

This can be done using Azure Cli, AZ PowerShell or updating the ARM template triggering a CD trigger process.

Please also see the Screen shot file.