

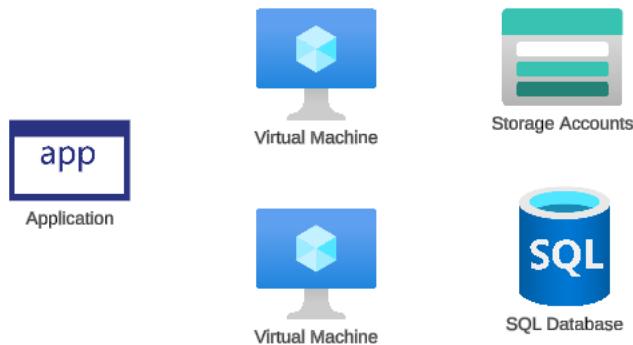
Introduction

What is Terraform and why do we use it

What is Terraform and why do we use it?

Terraform is an infrastructure as code tool.

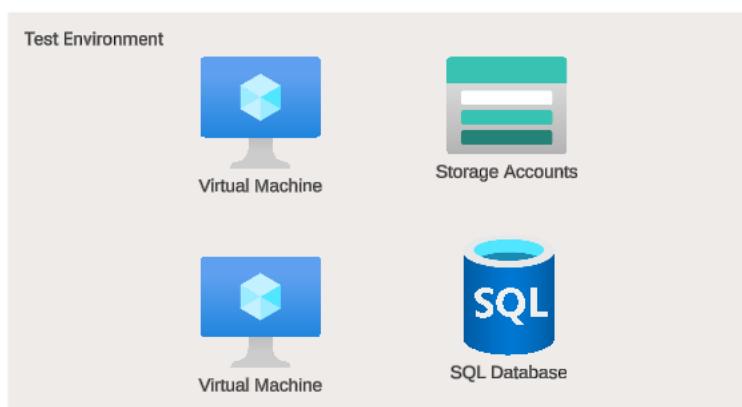
Using this tool you can define configuration files that allow you to deploy and configure your infrastructure-based resources.



Let's say a company has an application that needs to be hosted on a set of infrastructure-based resources.

You would normally provision these resources yourself and deploy the application.

You would normally provision these resources yourself and deploy the application.



Let's say that there is also a test environment for this application that also needs to have a set of resources.

Whenever the testing phase starts, you provision the infrastructure. And then delete the resources after testing is complete. You want to save on costs and keep capacity in place.



If you try to recreate the environment each and every time testing needs to be done, its first of all a headache of a manual task to recreate resources every time.



Secondly the repeated task of recreating resources is also prone to errors.



Code

We can avoid all of this by declaring a code configuration file.

This code configuration file would declare all of our infrastructure based resources.

We can then use Terraform to deploy the resources on demand based on the configuration file.

How does Terraform work

How does Terraform work?

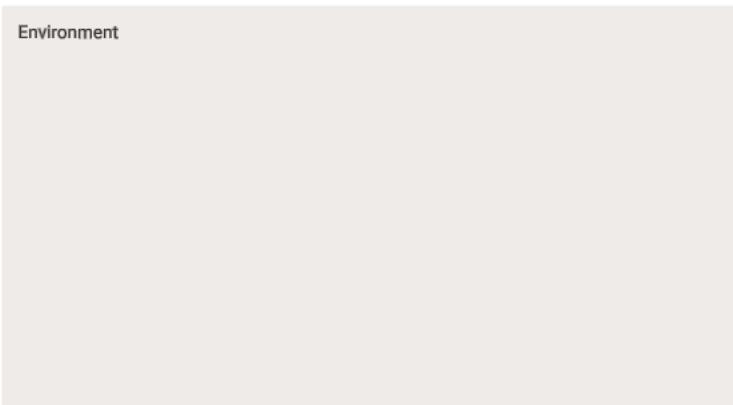


Code



First you write your infrastructure as code within a Terraform configuration file.

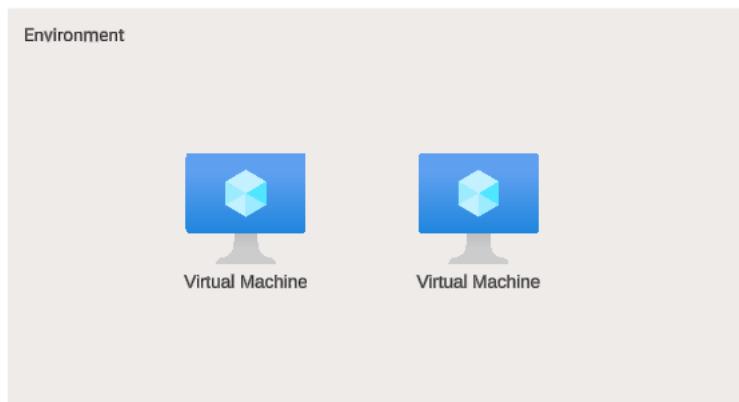
You tell Terraform where is your target environment.



You can then tell Terraform to compare the current infrastructure and the configuration file.

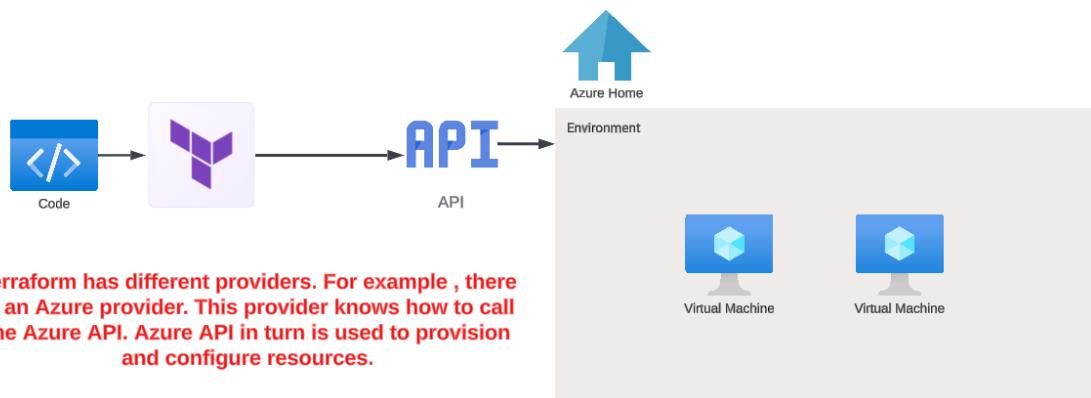
Let's say that the configuration file declares the presence of 2 virtual machines.

You then tell Terraform to apply the changes as per the configuration file.



The brilliance of Terraform is that it works with multiple cloud providers - Amazon Web Services , Microsoft Azure, Google Cloud Services.

It's not bound to any one cloud platform. Cloud providers do provide their own tools to manage infrastructure. But normally that is pertinent to that particular cloud provider.



Azure Storage Accounts and Virtual Machines

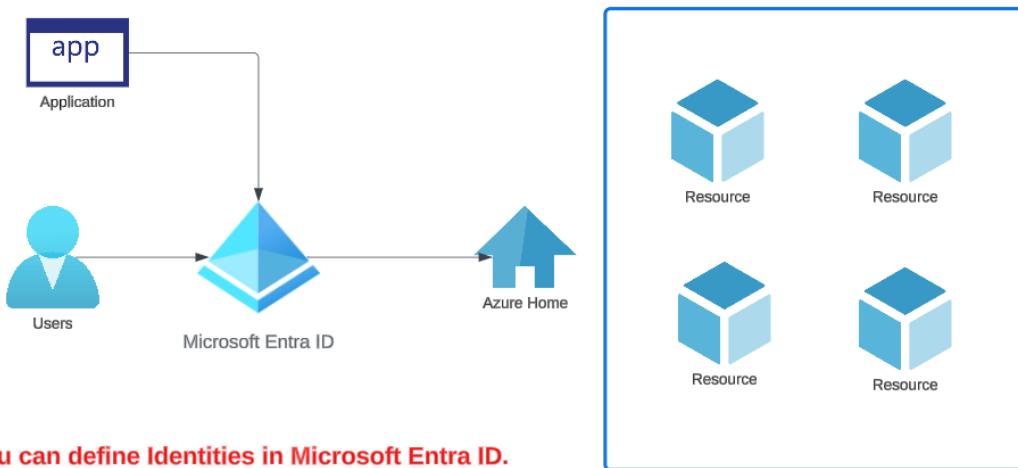
Authentication and Authorization onto Azure

Now using Terraform we want to be able to build our infrastructure on Azure.

In the Terraform configuration file we will use a provider for Azure that allows us to interact with Azure-based services.

But terraform needs to authenticate itself and have the right authorization to work with Azure-based services in our account.

Microsoft Entra ID is the identity-based provider that's available for Azure.



You can define Identities in Microsoft Entra ID.

Authentication - Here the identity of the users are verified.

Authorization - Here the permissions are checked for the users.

We need some way for Terraform to authenticate to our Azure account. There are several ways to achieve this.

One way is to define a user in Microsoft Entra ID. And then log in with that user using Azure CLI.

Another way is to use a Service Principal and a client secret with the terraform configuration file.

Lab - Write Terraform File - Azure resource group

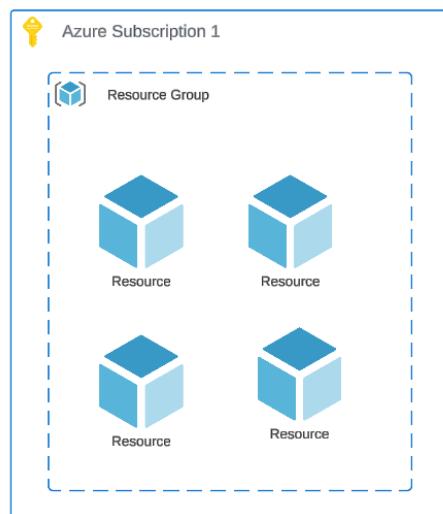
1. We first create a user in Microsoft Entra ID



2. Then we need to give access to our user. We will use Role-based access control to give Contributor access to the user over the subscription.

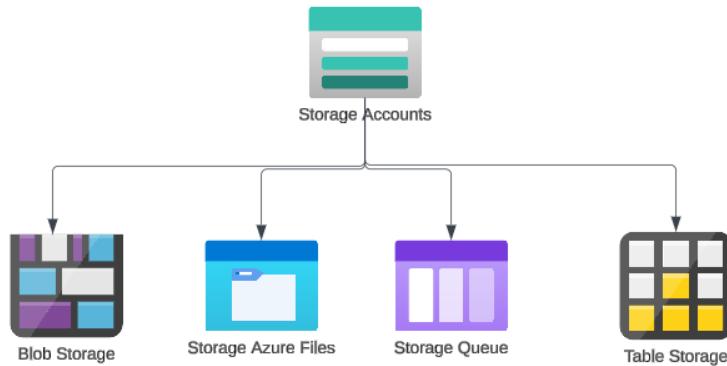
Note:- In actual production-based environments, we will not give such a broad level of access to the user.

Let's then add code to create a resource group. A resource group is a logical grouping of resources.



Lab - Creating an Azure Storage Account

Azure Storage Accounts - This is storage on the Azure cloud for your blob objects, files, queues and tables.



Azure Storage Accounts provides 4 services.

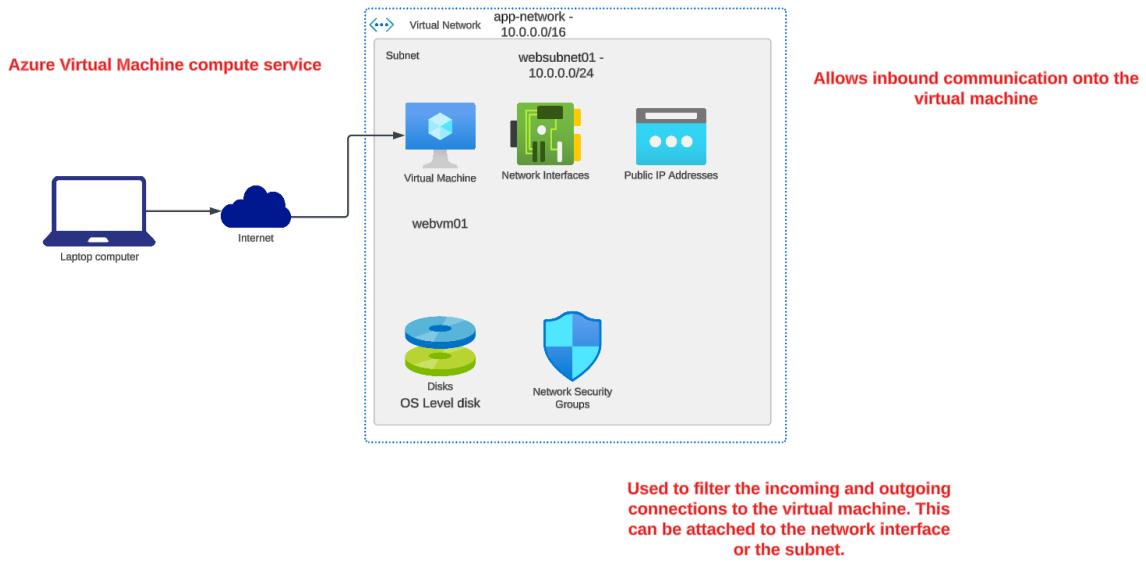
Commonly used service



This is used for storing a large amount of unstructured data. Suitable for storing images, documents, video and audio files.

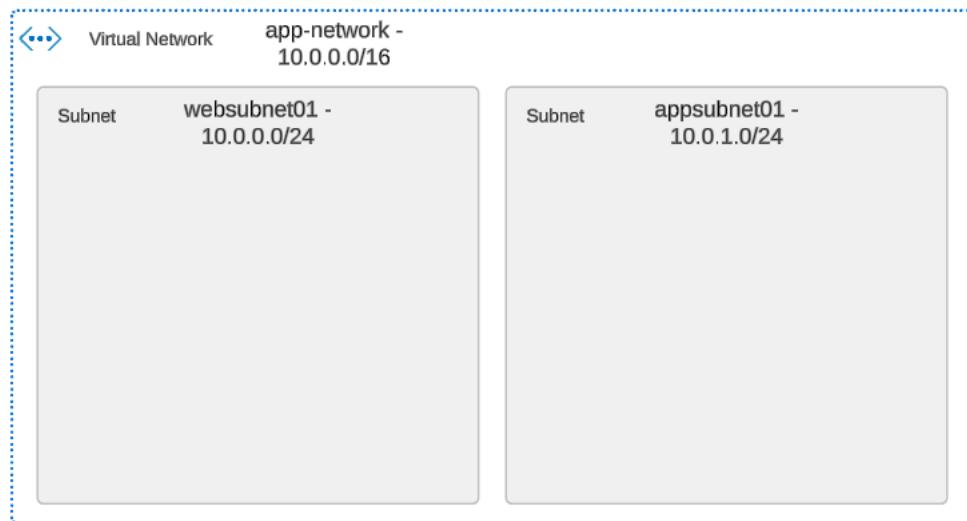
Within this service you create a container to hold objects. You can then upload files that are stored as binary-based objects.

Lab - Review the creation of a virtual machine

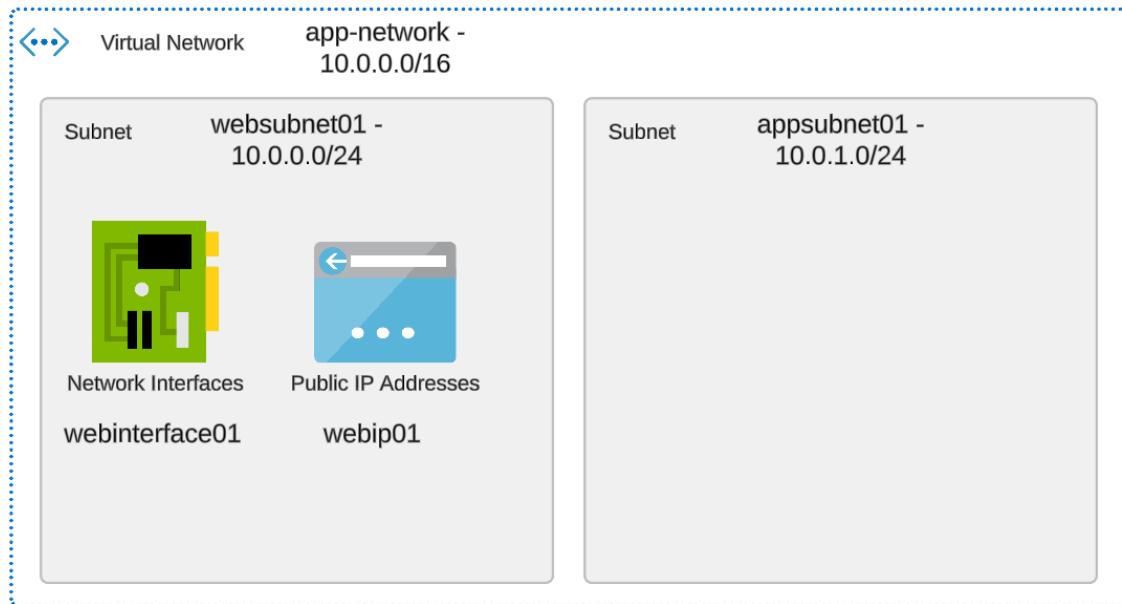


Lab - Creating an Azure Virtual Network - via Terraform

We just want to deploy an Azure virtual network with 2 subnets.



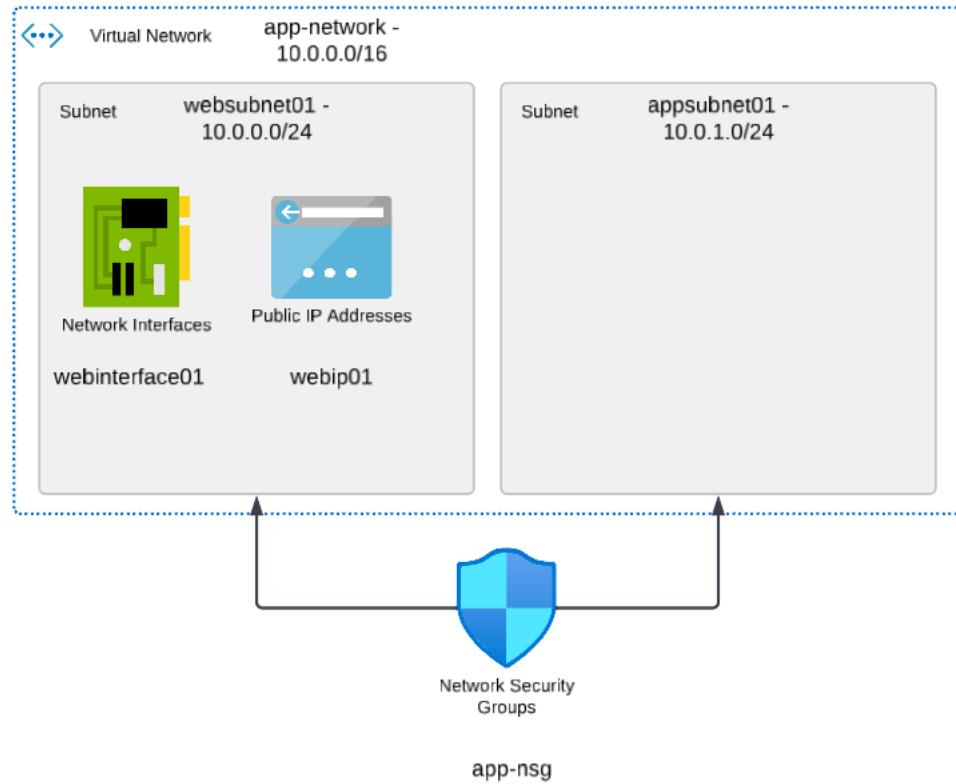
Lab - Creating a Public IP Address



Let's now create a Public IP address

We also have to ensure that the Public IP address is mapped to the network interface.

Lab - Creating a Network Security Group

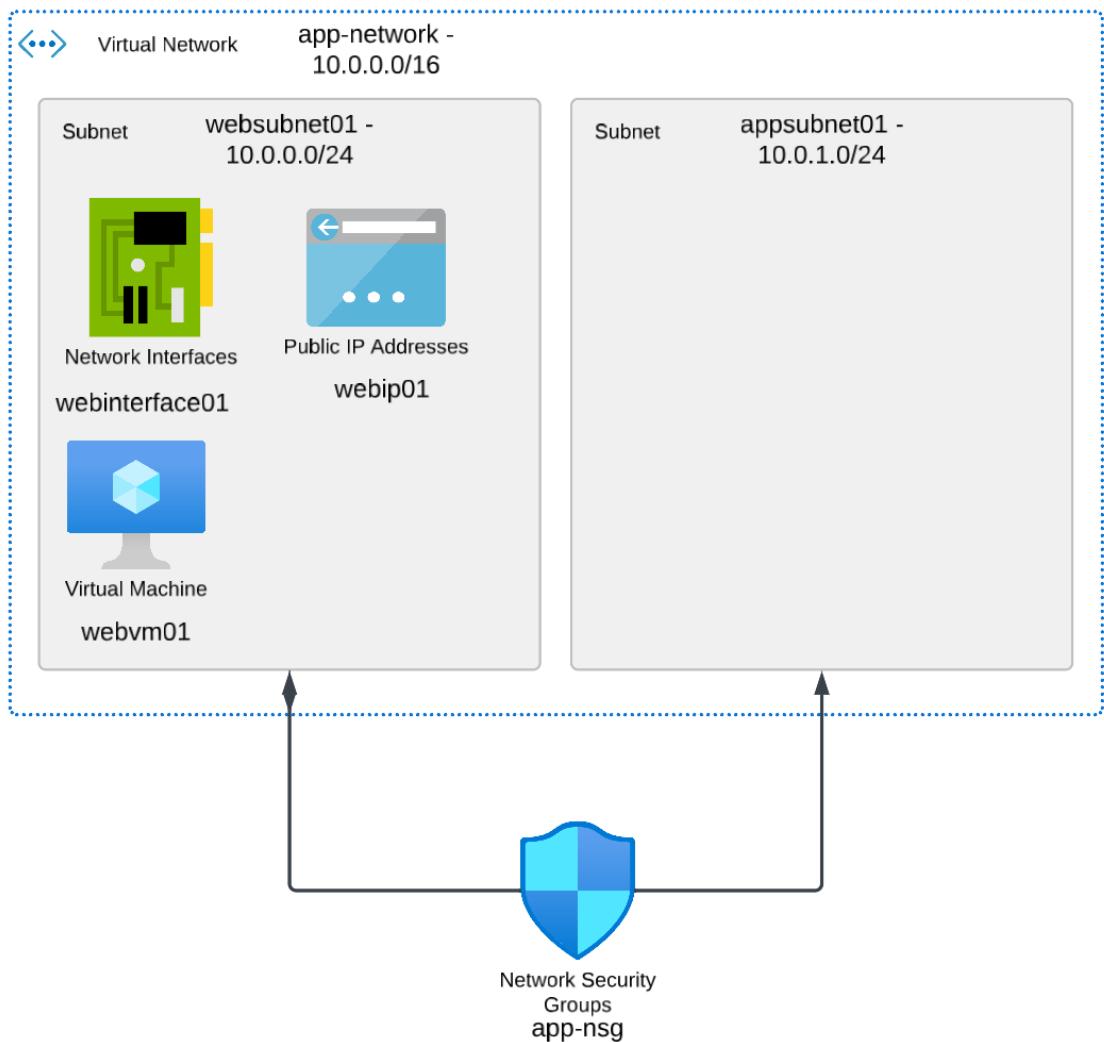


Let's create a network security group.

Let's assign the network security group to our subnets.

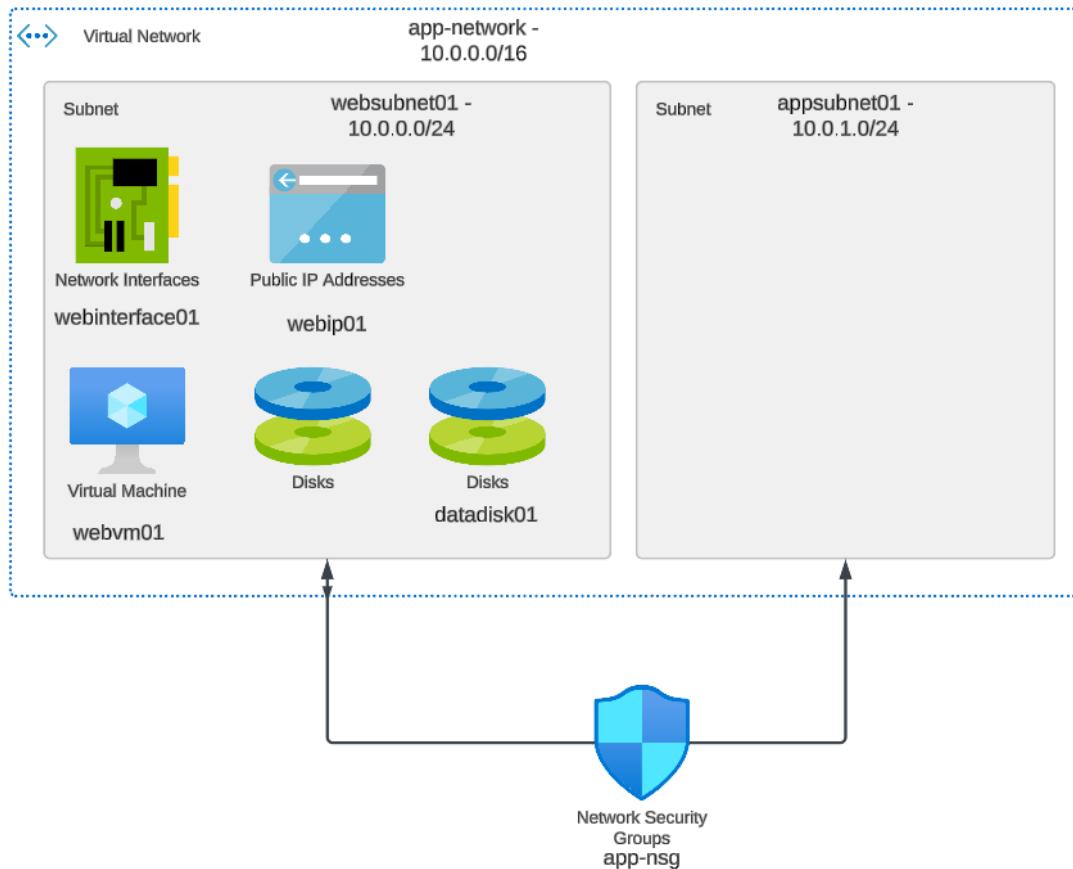
Lab - Creating the Azure Virtual Machine

Let's now deploy an Azure virtual machine based on Windows Server 2022.



Lab - Adding a data disk

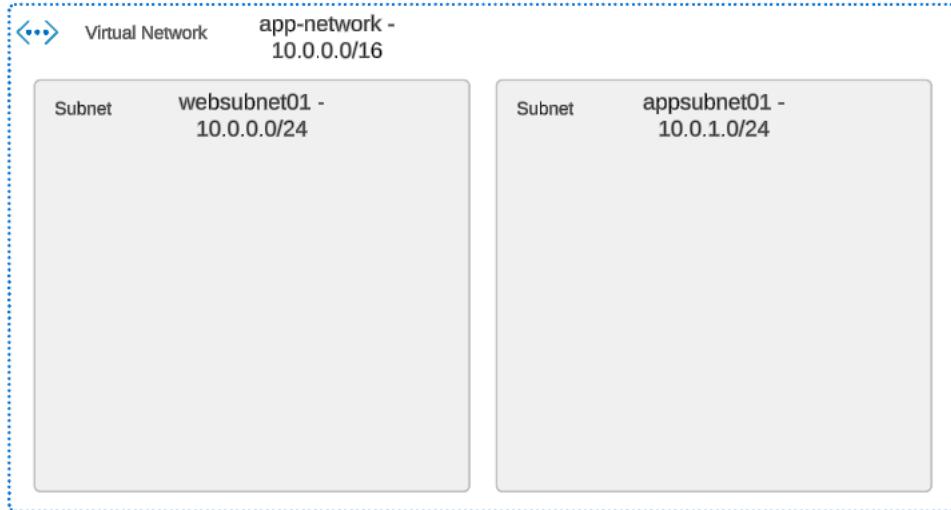
Let's add an additional data disk for the virtual machine.



Azure Storage Accounts and Virtual Machines - Further aspects

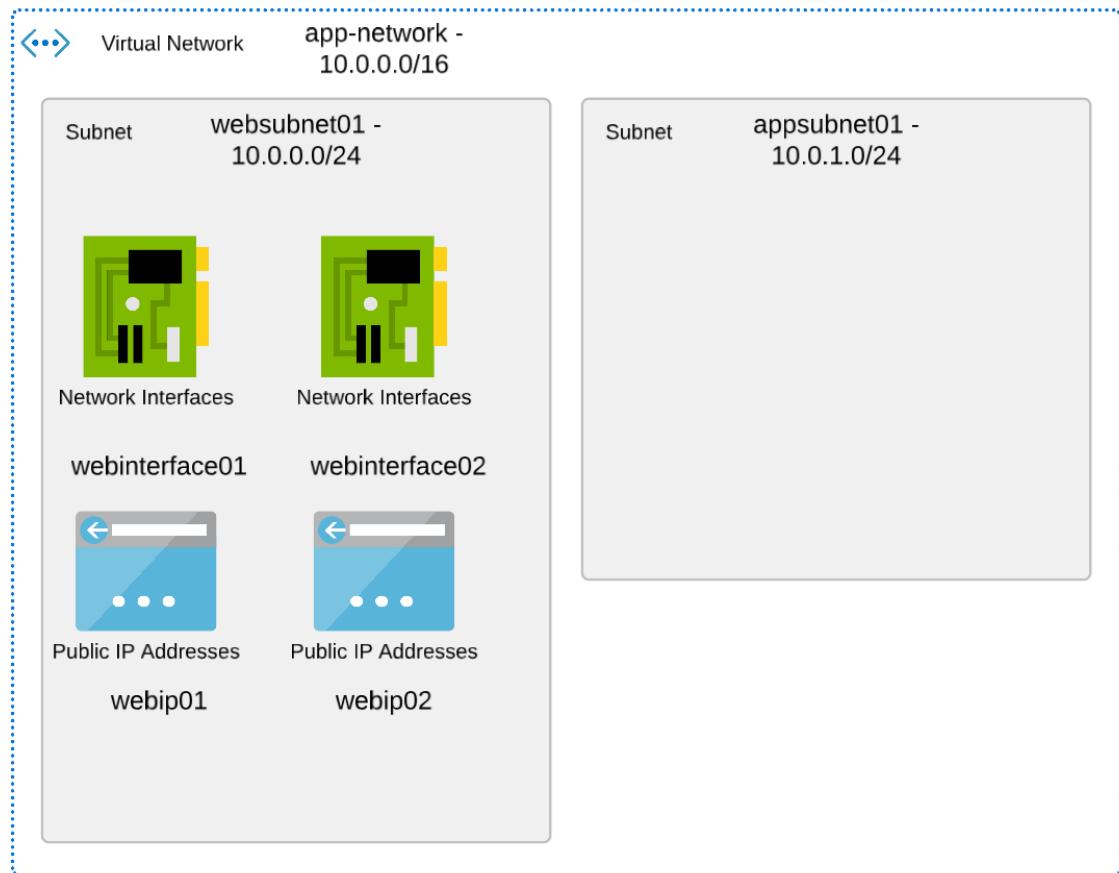
Lab - Creating multiple subnets

**Let's again deploy our virtual network and the subnets ,
but let's make use of Meta-Arguments**



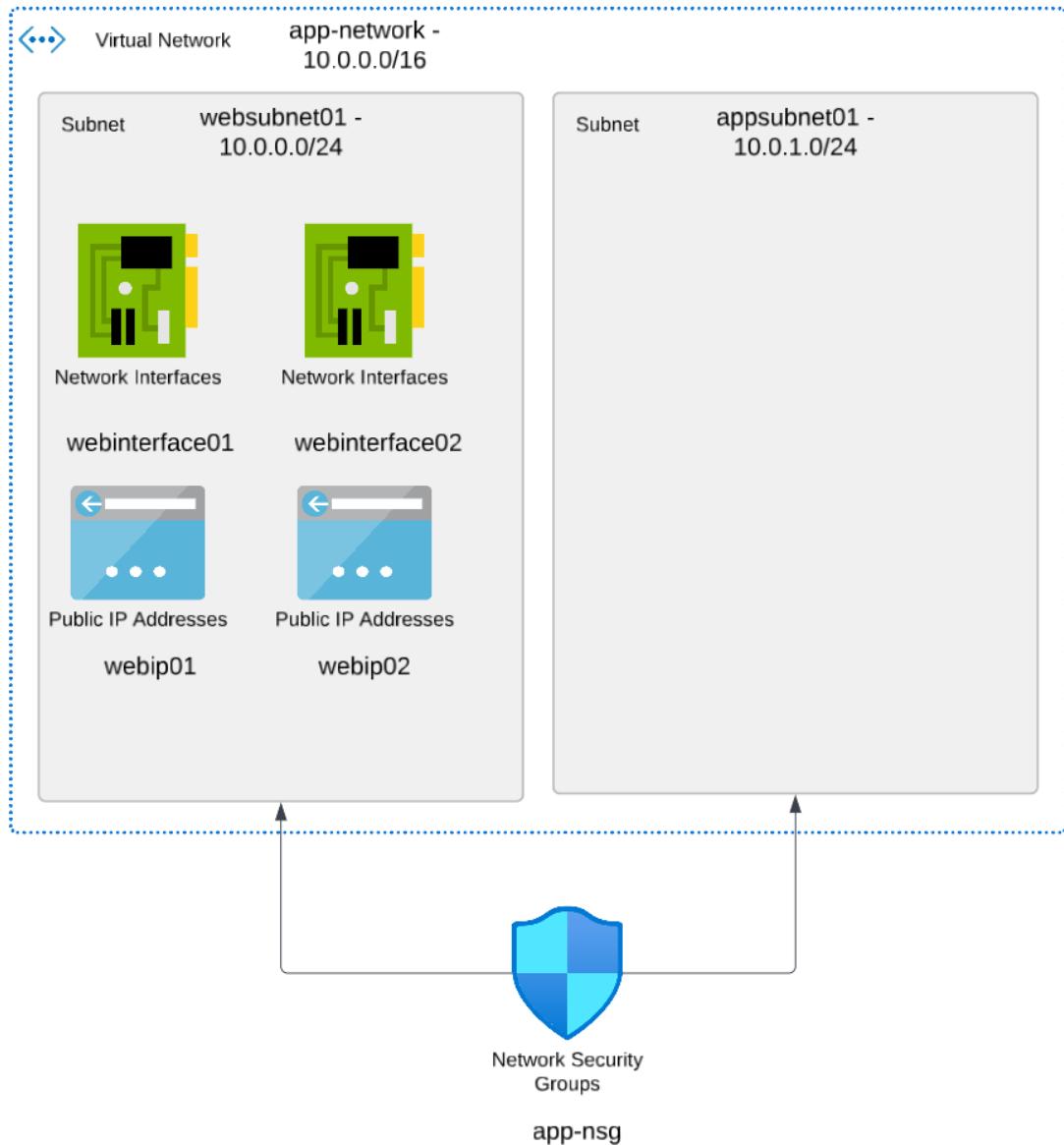
Lab - Creating multiple Public IP addresses

Let's now create multiple Public IP addresses for the multiple network interfaces



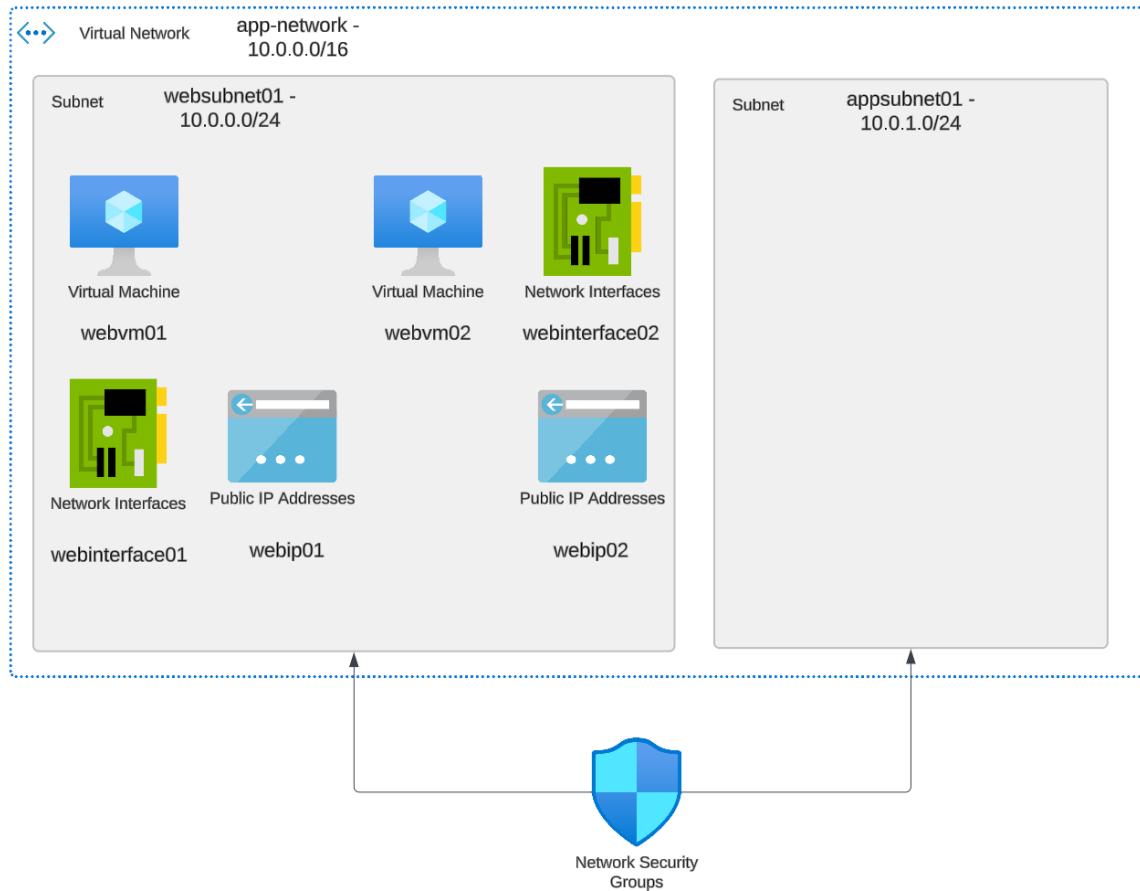
Lab - Network Security Groups

Let's again have one Network Security Group that is attached to both subnets.



Lab - Creating multiple virtual machines

Let's now create multiple Azure virtual machines



Lab - Availability Sets

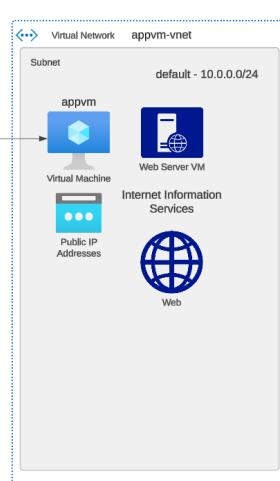
Your company is hosting a web application on an Azure virtual machine.



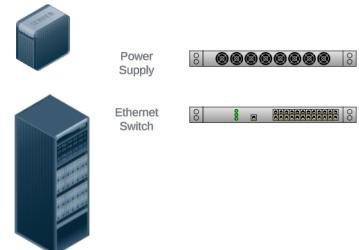
Users are accessing the web application from the Internet.

What happens if the Virtual machine goes down? Azure does everything to ensure that the virtual machine service, the underlying infrastructure is kept up and running.

But issues are bound to occur. In such a scenario your application would become unavailable.



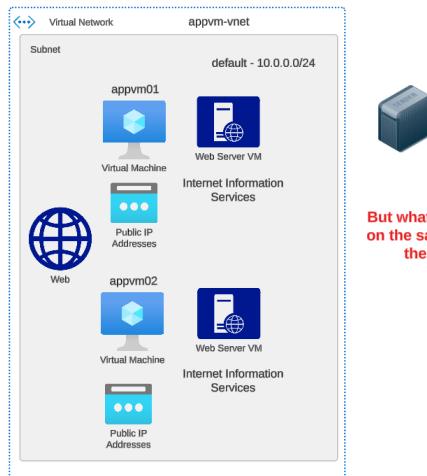
There could be faults in the underlying physical layer, maybe a network issue, a power issue.



Sometimes Microsoft also has to perform vital updates to the physical machines that host the virtual machines. At that point in time, a reboot of the physical server could be required. This would again cause a disruption to the application.

We need to look at increasing the availability of our system.

You can look towards hosting your application on multiple virtual machines.



But what happens if both virtual machines are launched on the same physical server. By default we can't specify the underlying physical servers for the virtual machines.

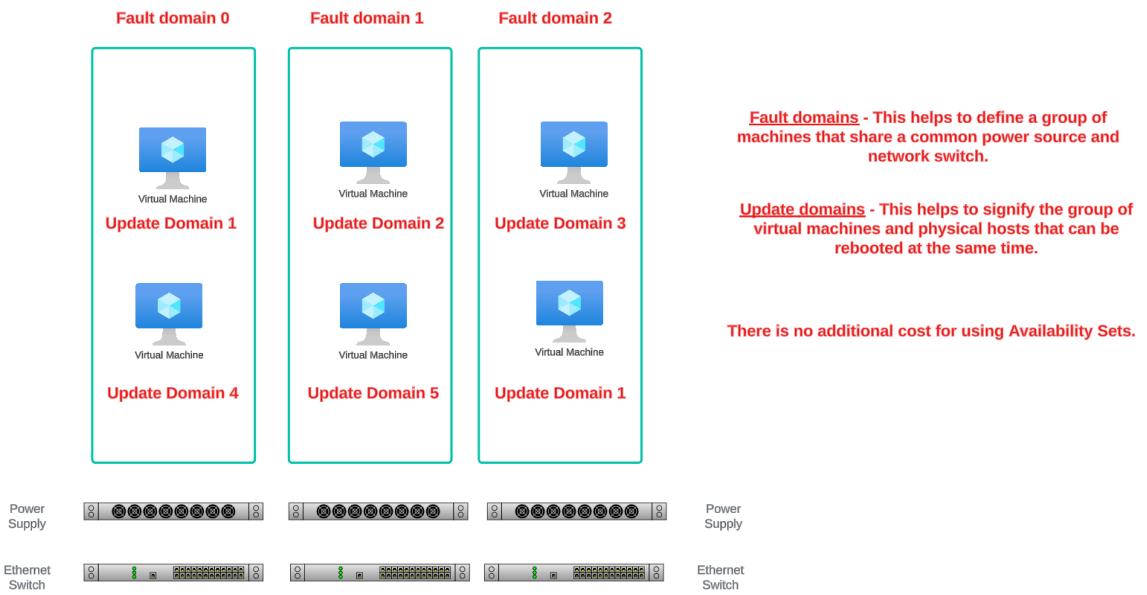
Availability sets

This is a logical grouping of machines that helps to reduce the chances of multiple VM's going down because of hardware issues.

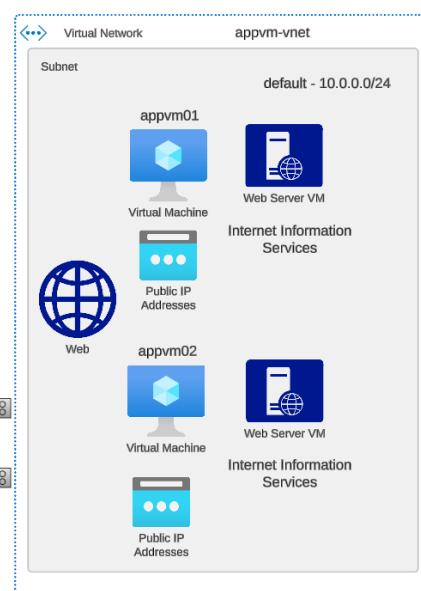
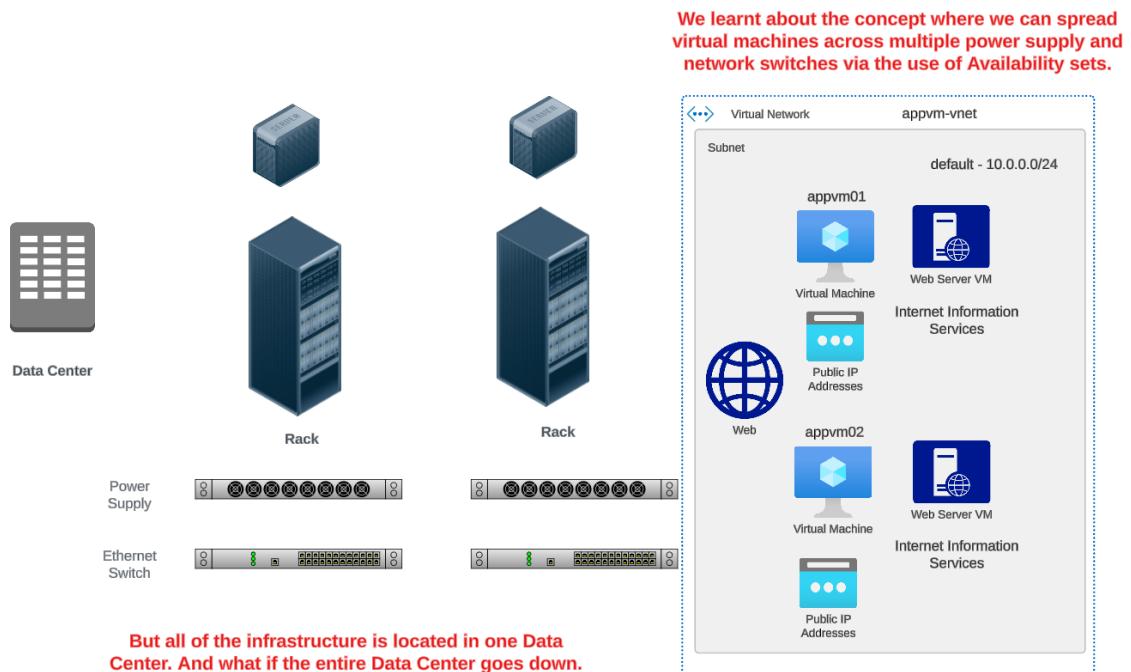
To make use of Availability sets, you need to deploy a Virtual Machine to an Availability Set. You can just create an Availability Set and deploy the machine to the set. The machine can only be part of a set when the machine is created.

The virtual machine is placed as part of a fault and update domain in the Availability set.





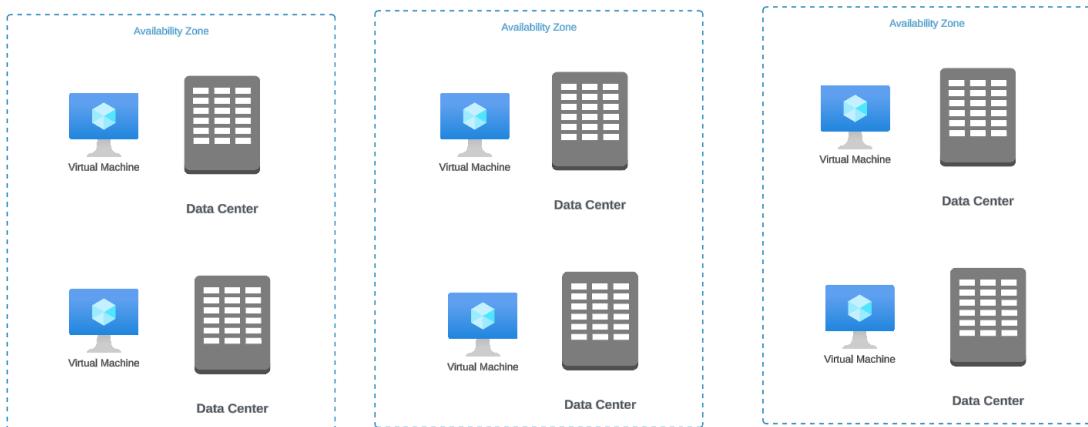
Lab - Availability Zones



You can spread your infrastructure across multiple Availability Zones.

An Availability zone is a group of data centers. There are fast links across Availability Zones to ensure low latency.

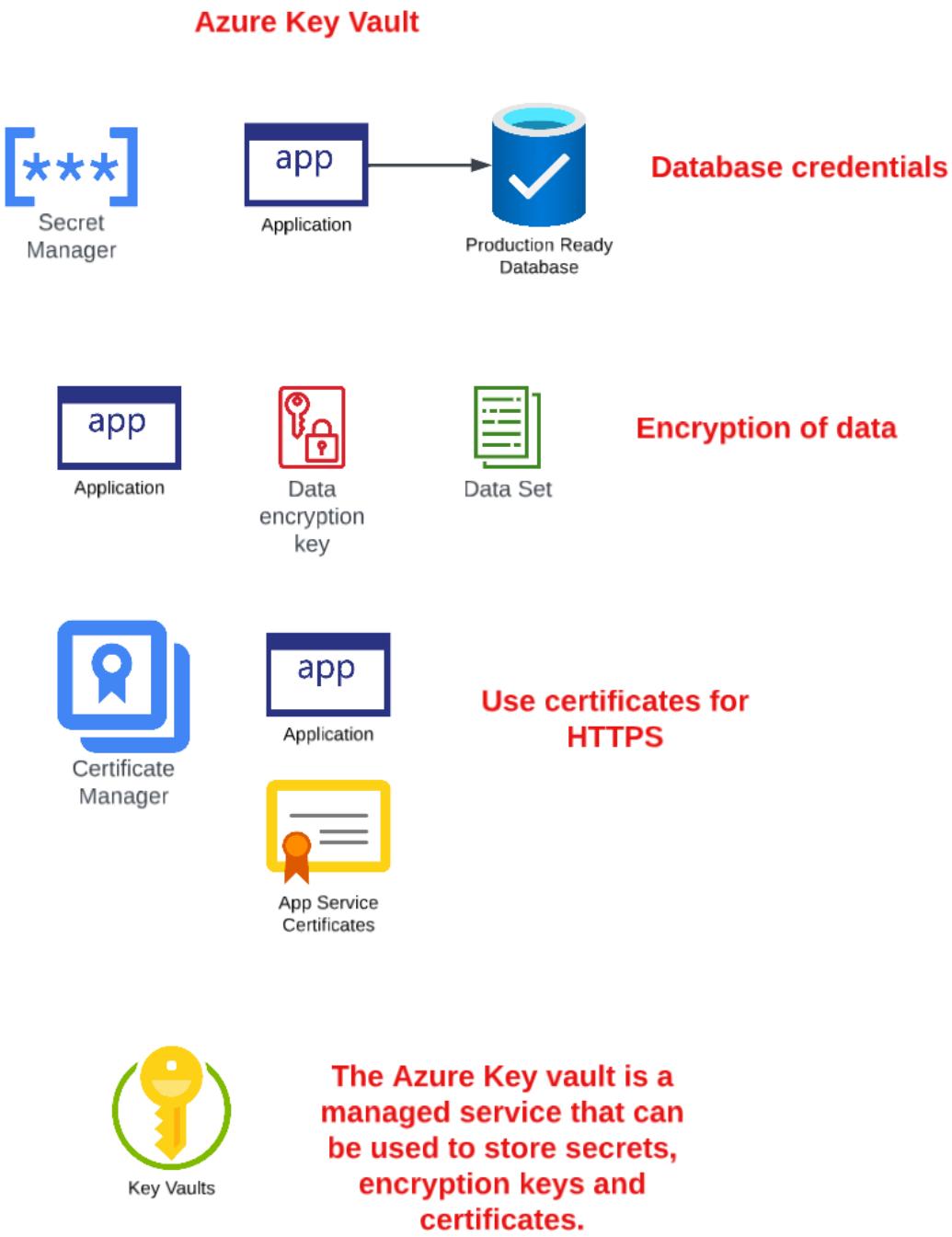
An Azure region has multiple Availability zones.



Data Transfer	Price
Data Transfer In	Free
Data transfer between Availability Zones(Egress and Ingress)*	\$0.01 per GB
Data transfer within same Availability Zone	Free

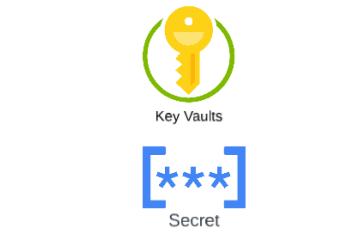
There is no cost for using an Availability zone, but there is a bandwidth cost.

Lab - Azure Key Vault

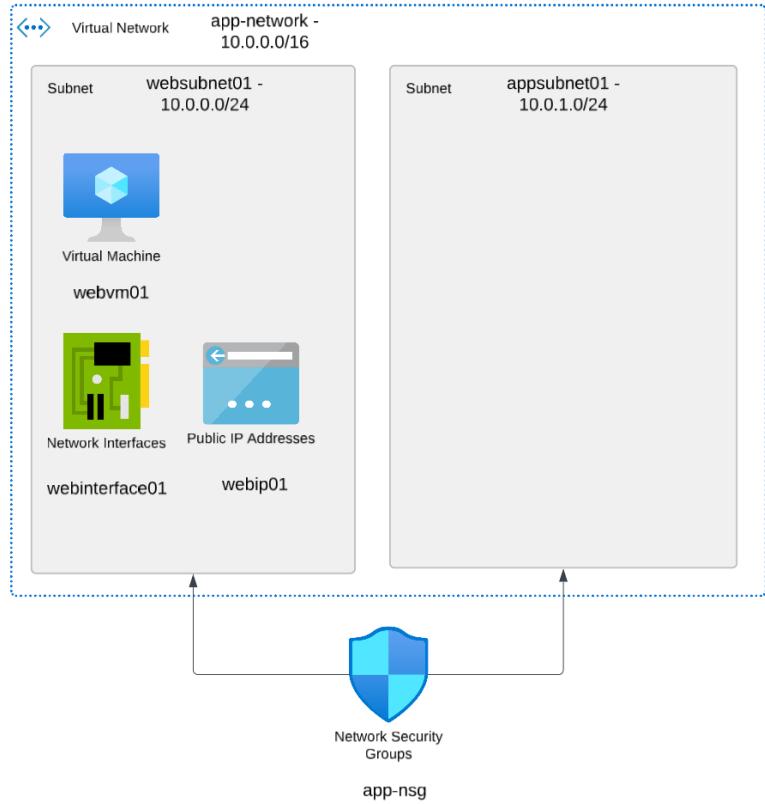


Lab - Using Data sources

Let's just deploy one Azure virtual machine



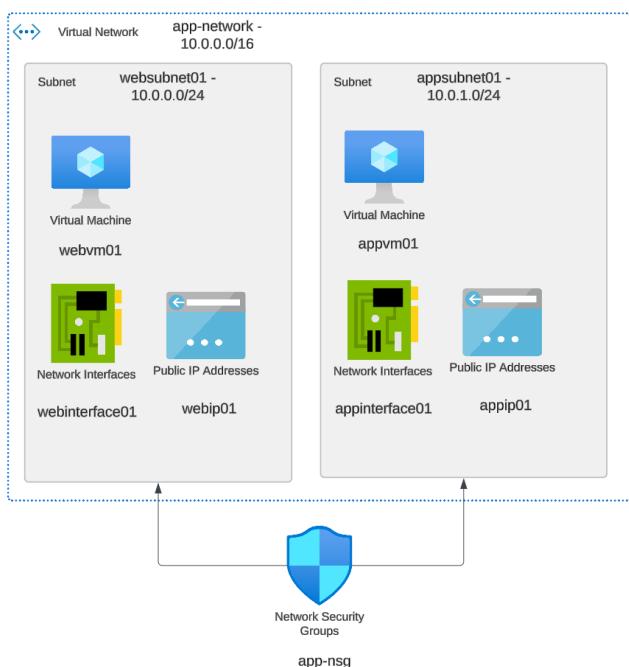
Here the admin password would be stored in the key vault.



Lab - Building a Linux machine

Let's deploy a Linux based machine

Also for the machine via the use of cloud init, let's install the nginx package so that the machine behaves as a web server.

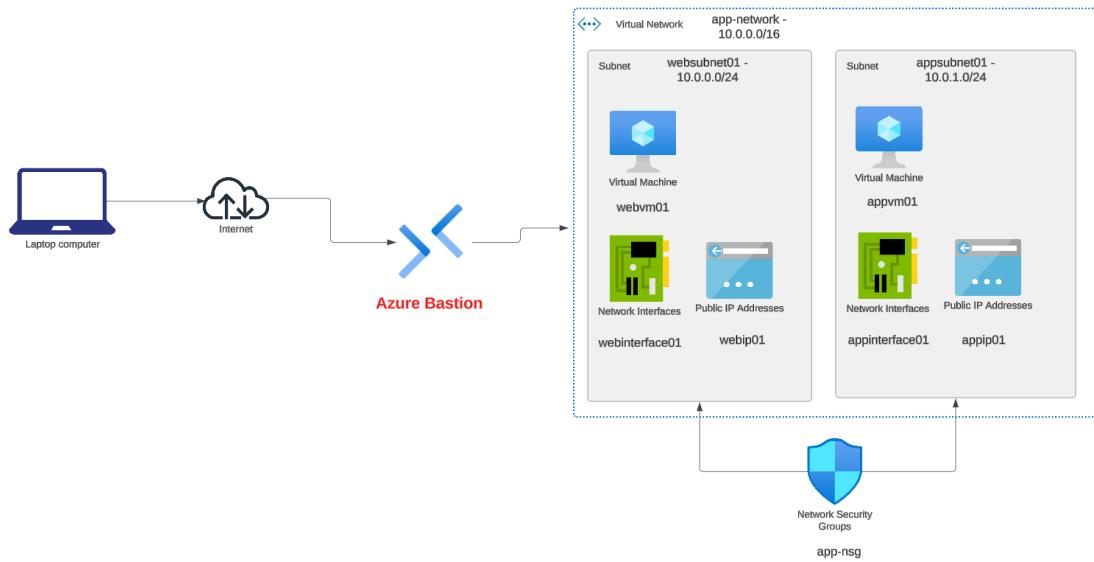


Lab - Azure Bastion

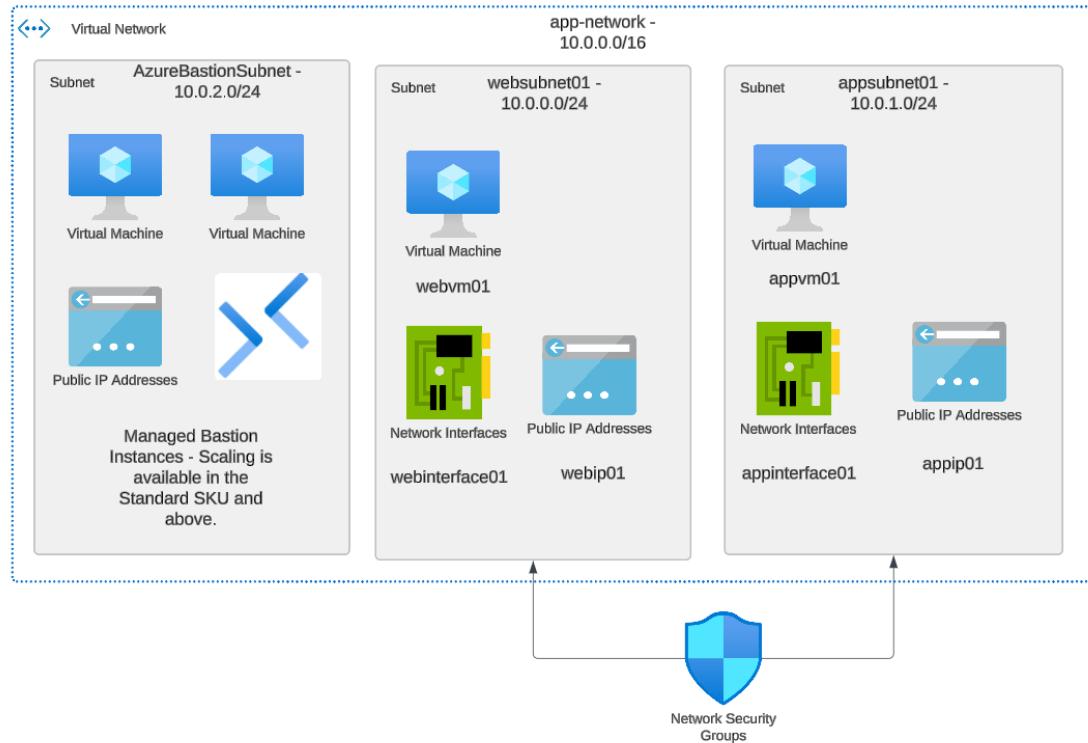
Azure Bastion

This is a fully managed service that provides secure connections to virtual machines without the need of machines needing to have a Public IP address.

You can establish RDP and SSH connections to virtual machines from the Azure Portal.



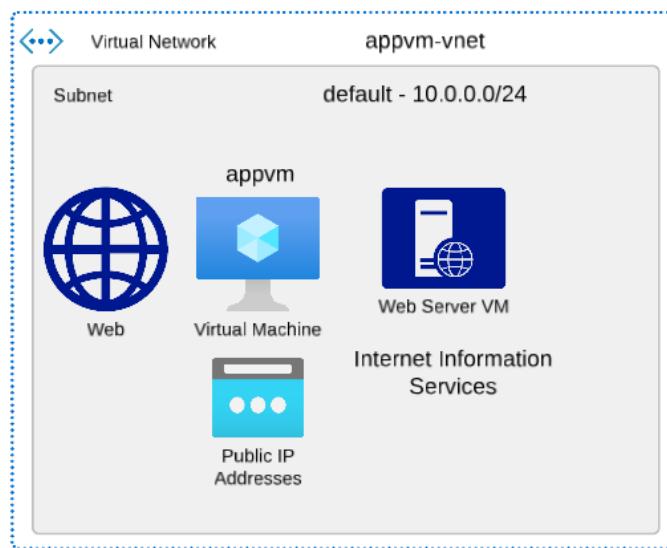
Let's deploy an Azure Bastion resource



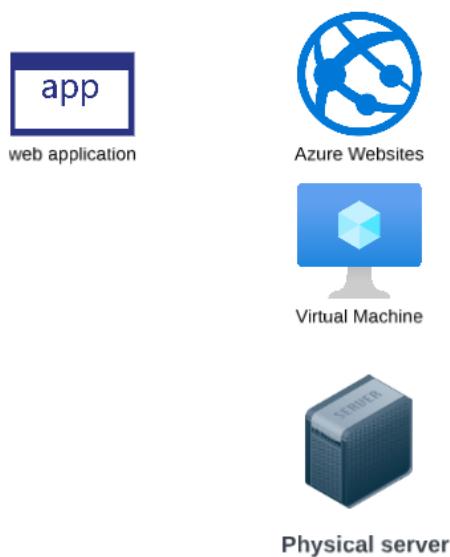
Azure Web App and SQL Database

Azure Web App service

We understand the concept wherein we can host web applications on Azure virtual machines



Azure Web App Service



This is a managed service. Here the virtual machine and physical infrastructure is managed for you.

There is support for web applications based on .NET, Java, Node.js, PHP, Python.

Here the patching of the framework and the operating system is managed by the service.

You also get other features such as High Availability.

If you have a web application that fits the framework and you don't want to manage the virtual machines, then you can opt for the Azure Web App service.

But if you need to host a custom application that needs to be installed, then you would probably need to use the Azure virtual machine service.

Understanding on deployment slots



Application

We have deployed our application



Azure Websites



Application v1

Now before we actually deploy the newer version of the application, we would ideally first want to test the application. At one phase, with a set of business users.

We now have a newer version of the application.



Application v1



Azure Websites - Test

1. Create a new Azure Web App
2. Deploy the newer application
3. Test the application
4. Publish the application after successful testing to the primary Azure Web App

In Azure Web Apps , we can make use of deployment slots.



Azure Websites

This feature is available with the Standard, Premium and Isolated App Service Plan.



Production - Web Slots



Staging - Web Slots

Create a new slot and publish the newer version of the application to the slot.



Application



Application v1

Each slot is a live web application with its own host name.



Azure Websites



Production - Web Slots



Staging - Web Slots



Application v1



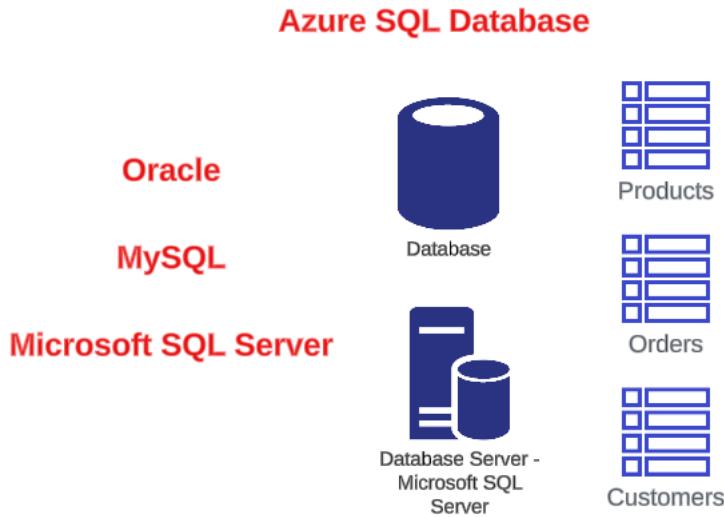
Application

Then at any point in time, you can swap the slots. So that the newer version of the application runs as part of the production slot.

This helps in first testing of the application in the staging slot and then swapping the slots at any point in time.

It also helps in recovery from failure. If the swap succeeds , but the newer version of the application is not working, you can easily swap back at any point in time.

The Azure SQL Database service



We need to install the database software on some sort of compute hardware.



We can install the database engine on a virtual machine.

Database administrator responsibilities

1. Uptime of the database server
2. Database backups and restore
3. Patch installation at the operating system and database engine level.



If the company does not want the burden of managing the underlying infrastructure, they can opt to use the Azure SQL Database service.

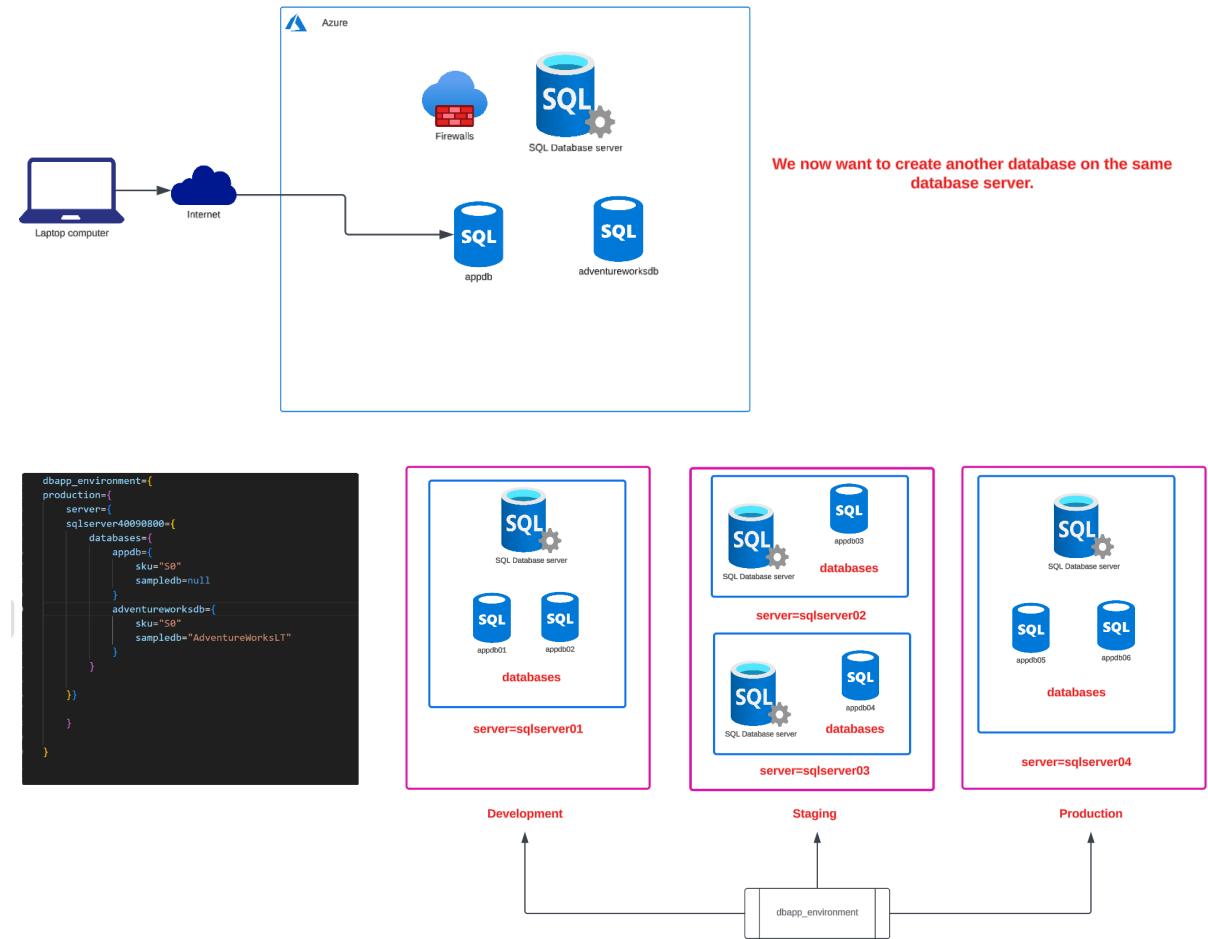


Here the underlying server is managed for you. The database software will be in place. It also has features such as backup/restore and several other features.

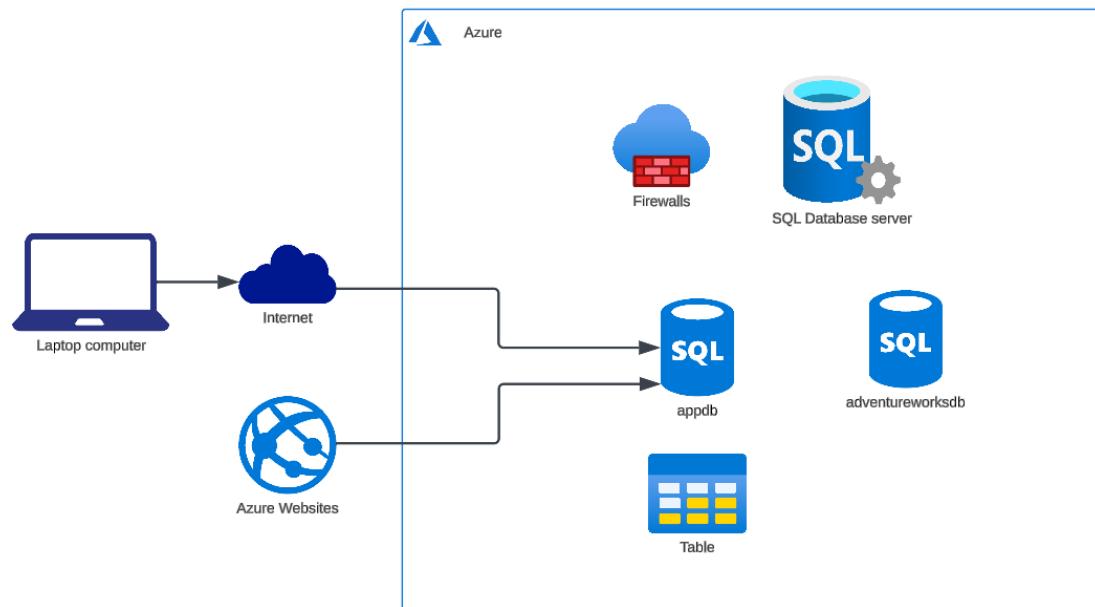


You can simply start hosting your databases. The Azure SQL database is the cloud version of Microsoft SQL Server.

About deploying another Azure SQL database



Connecting a web app to SQL database – Overview



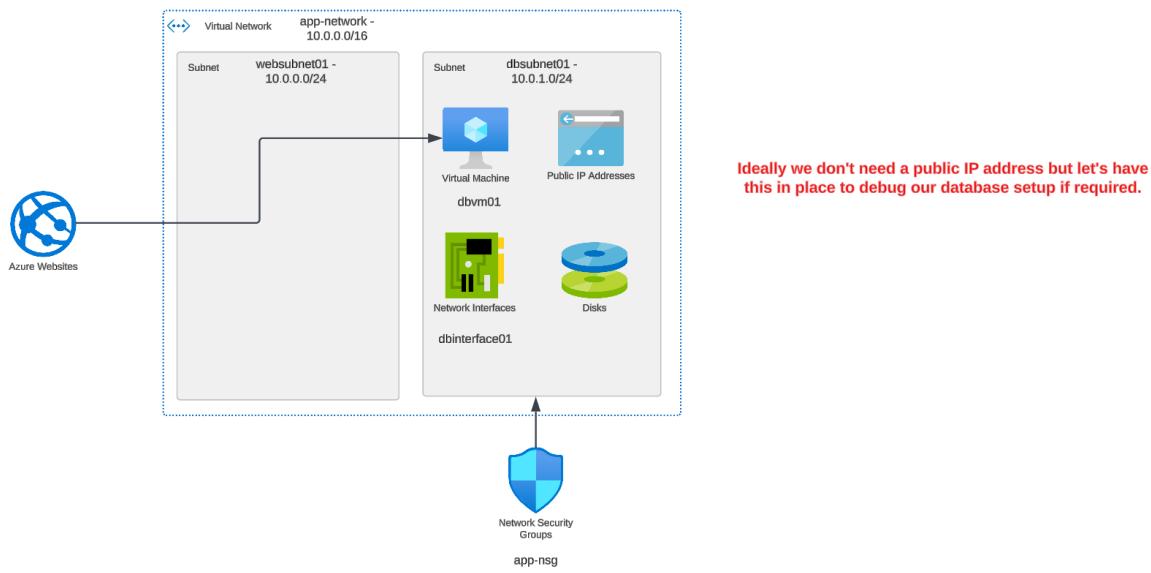
We will deploy an Azure Web App

We will publish code onto a GitHub repository - This will be based on .NET.

Using Terraform we will publish the .NET app from GitHub onto the Azure Web App.

The web application will connect to the Azure SQL database.

Mini-Project – Overview



Ideally we don't need a public IP address but let's have this in place to debug our database setup if required.

What we want to accomplish - A public facing web app needs to be able to privately connect to an Azure virtual machine that hosts a MySQL database.

Step 1: Create an Azure virtual machine that has MySQL installed.

Step 2: Configure the user credentials for the database, have a database, a table and data in place.

Step 3 : Deploy a web application from GitHub onto an Azure Web App.

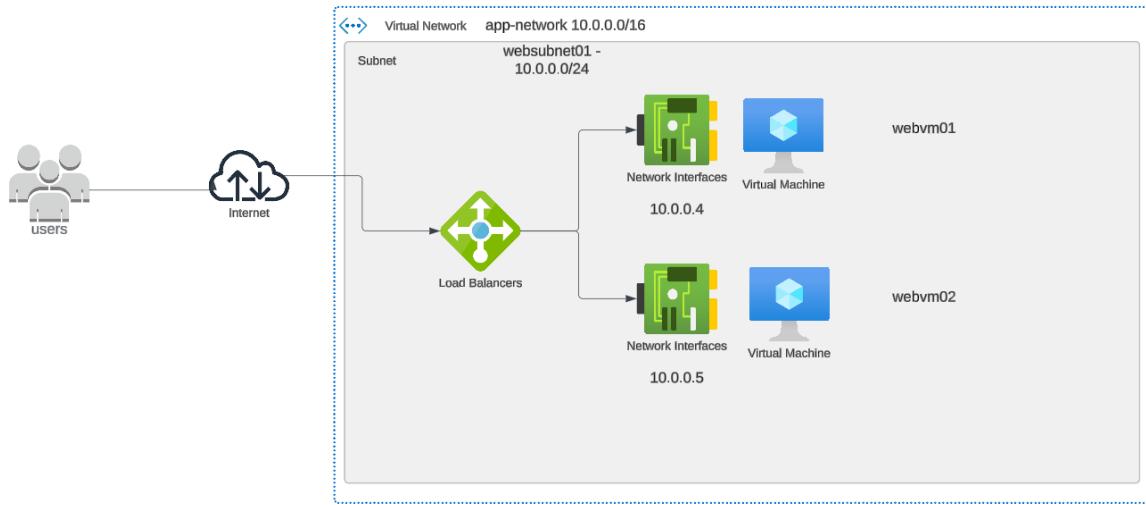
Step 4: Implement virtual network integration for the Wbe App which allows the web app to private communicate with the resources in an Azure virtual network.

Azure Networking

About the Azure Load Balancer

Azure Load Balancer

A load balancer is used to distribute the incoming network traffic across a set of backend servers.

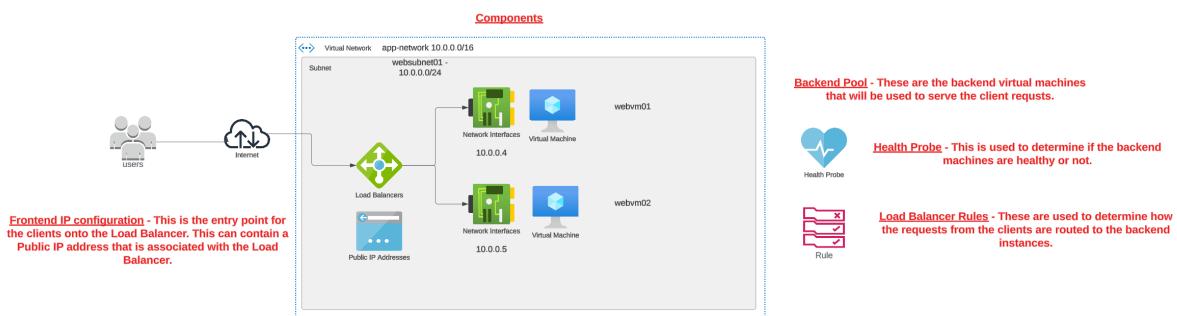


You can have an application hosted on a set of machines.

You want user traffic to be distributed equally across the machines.

For this we can make use of the Azure Load Balancer service.

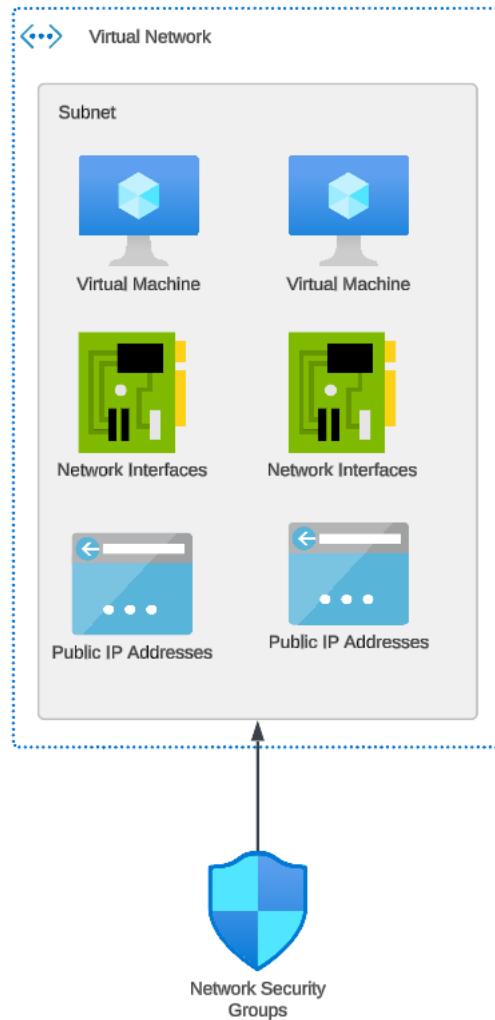
Here the Azure Load Balancer can distribute traffic across the private IP addresses of the backend machines.



We are going to make use of modules

Terraform modules - These are used as a container for multiple resources that are used together. This is a collection of .tf or .tf.json files that are kept together in a directory.

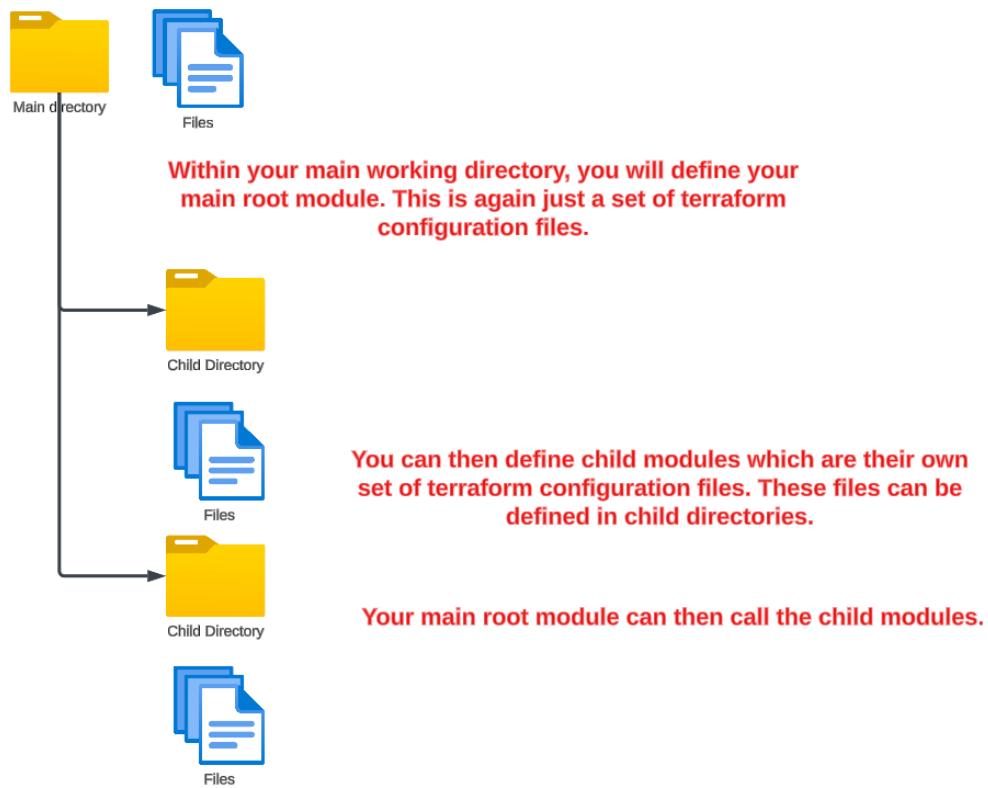
What's the point of using modules



Let's say that the deployment of resources such as the virtual network and virtual machines is common across a lot of environments.

We can have one module that contains the deployment of the network and machine resources.

And then re-use the modules in the deployment of the resources in other environments.



The parent module can call the child module and pass data to the child module via the use of Input variables.

The input variables are defined in the child module.

The child module has no direct access to the variables or data available in the parent module. That is why we need to pass data to the input variables of the child modules.

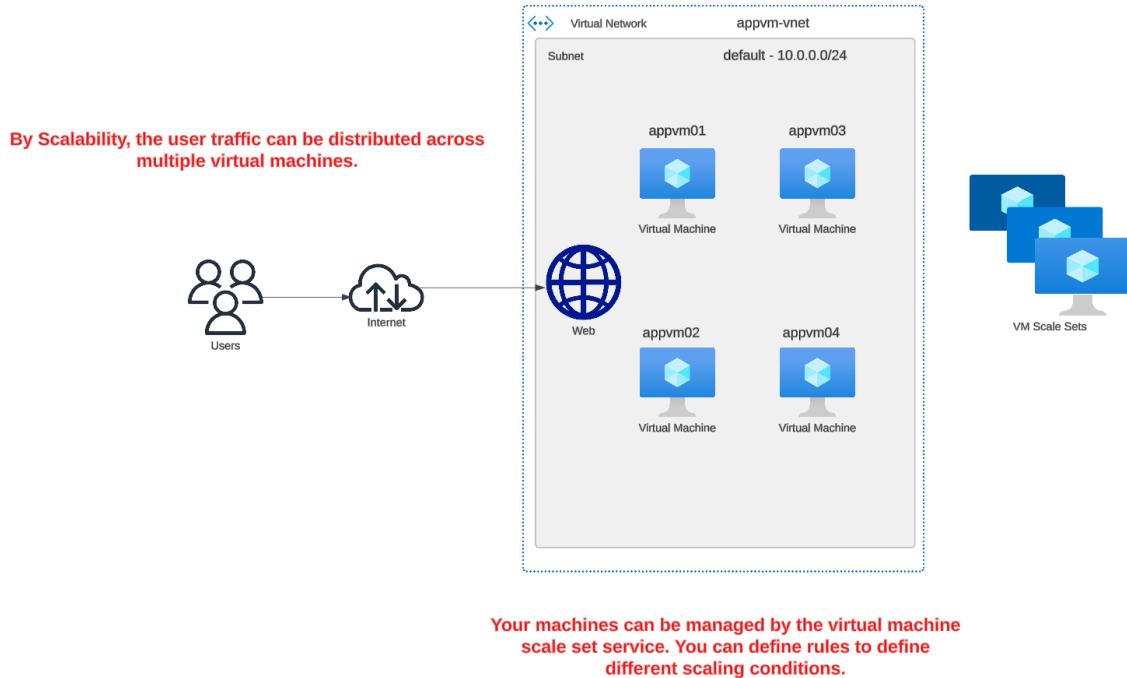
The child module can then pass in data to the parent module via the use of output variables.

Lab - Virtual Machine Scale Set Deployment - Manual Process

[Azure virtual machine scale sets](#)

This service helps you to create and manage a group of load balanced virtual machines.

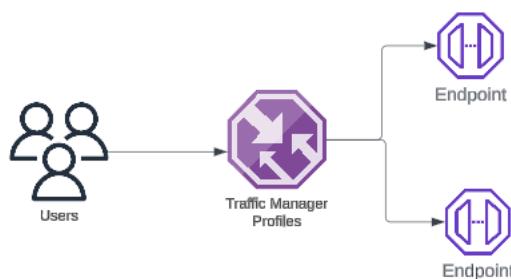
The number of virtual machines can then grow based on demand or on a schedule.



About Azure Traffic Manager

[Azure Traffic Manager](#)

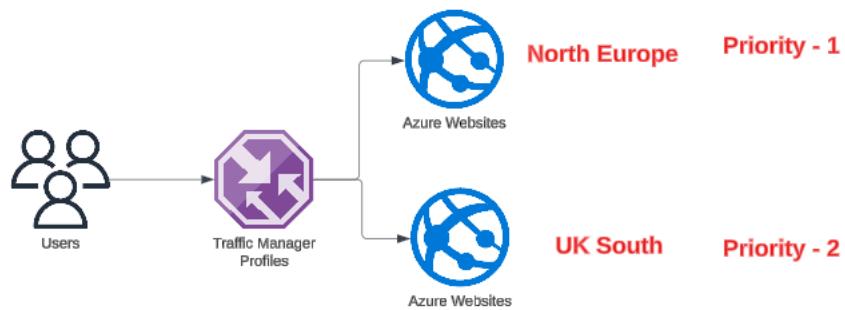
This is a DNS-based load balancer. Here you can distribute traffic across your resources at a global scale.



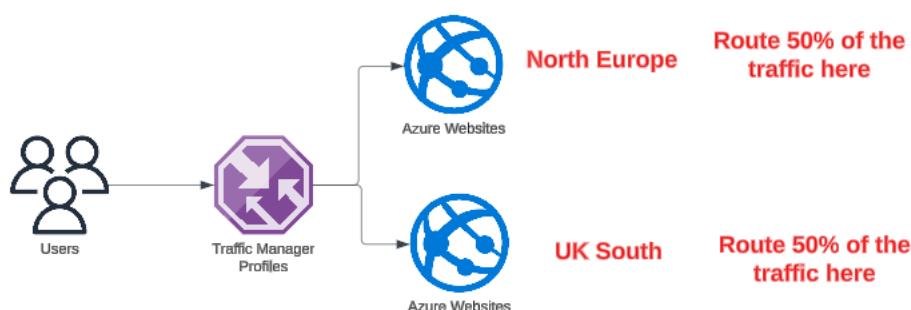
Here you can direct traffic to different endpoints. The endpoints could be inside or outside Azure.

You can also specify different routing methods when using Azure Traffic Manager.

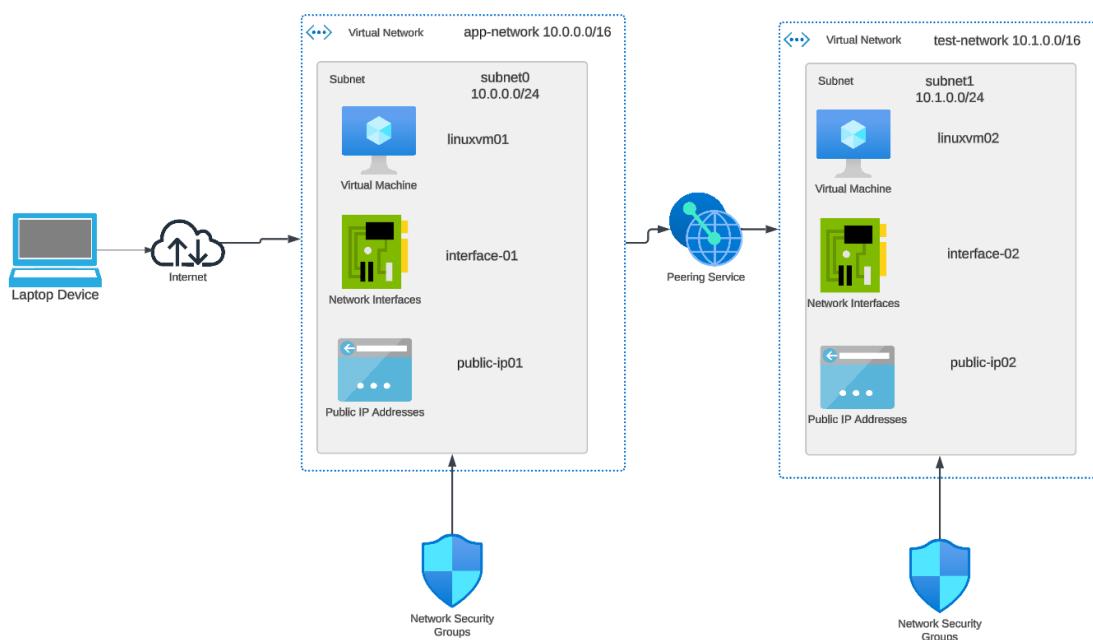
Priority Routing Method



Weighted Routing Method



Lab - Virtual Network Peering - via Terraform - Virtual Network Setup



We can establish connectivity between virtual networks via the use of Virtual Network Peering.

Lab - Azure Application Gateway - Implementation - Manual Process

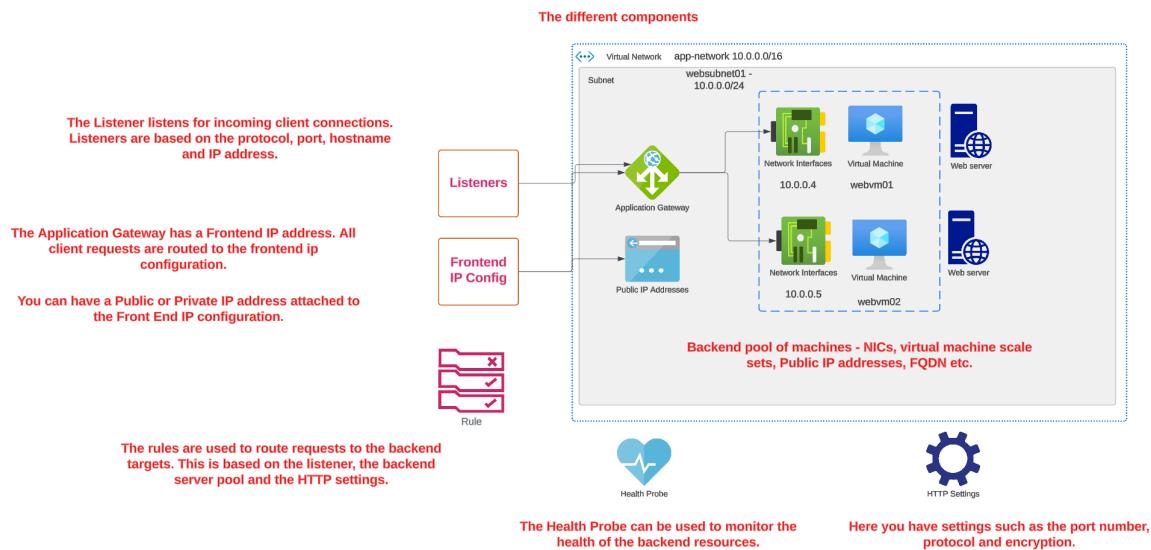
Azure Application Gateway

The Azure Application Gateway is a Layer 7 load balancer. Here the routing decisions can also be made based on the details of the HTTP request.

When you make a request for a URL, there are a lot of attributes associated with the HTTP request.

X	Headers	Preview	Response	Initiator	Timing	Cookies
▼ General						
Request URL:	https://learn.microsoft.com/en-us/azure/application-gateway/overview					
Request Method:	GET					
Status Code:	200 OK					
Remote Address:	23.51.49.217:443					
Referrer Policy:	strict-origin-when-cross-origin					

The Azure Application Gateway can parse the HTTP request and route the request accordingly.

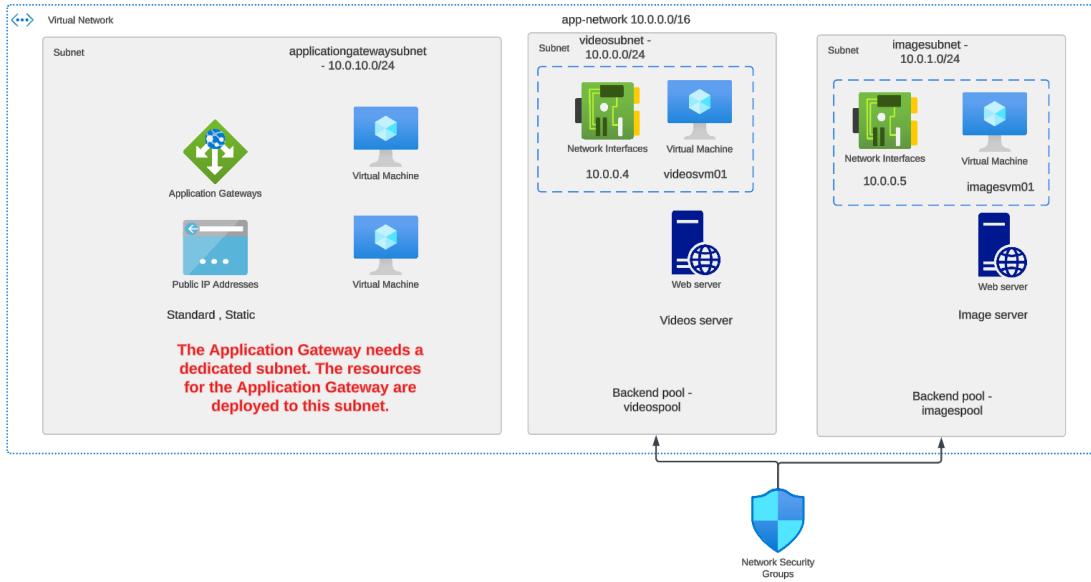


Our implementation

We want to base the routing of requests based on the URL.

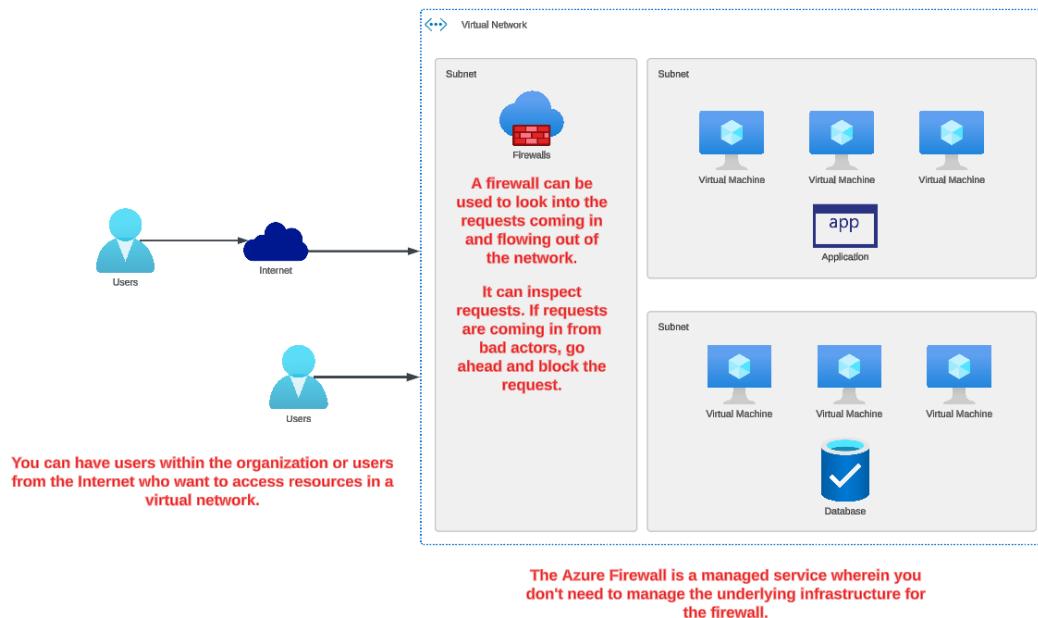
If the URL contains /videos then the request needs to be directed to the backend pool that contains the videos server.

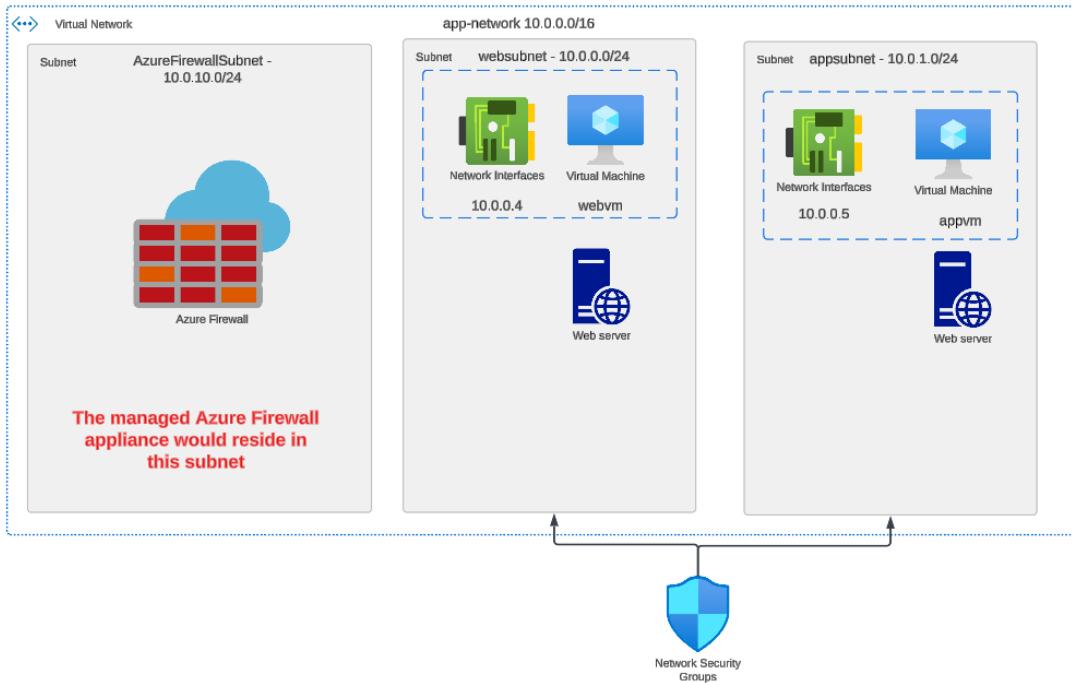
If the URL contains /images then the request needs to be directed to the backend pool that contains the images server.



Lab - Azure Firewall - Virtual Machine setup

Organizations normally have a full fledged firewall appliance for protecting their resources in networks.





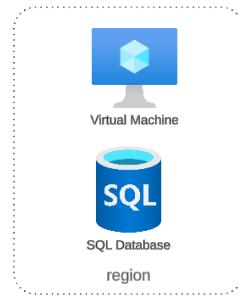
Identity Management, Monitoring and governance

Quick look into Azure Monitor

This service allows you to collect data for your resources in Azure and your on-premises resources as well.



You can analyze and work on the analyzed data.



You can look at the metrics collected for various resources



Alerts can be generated if metrics for resources go beyond a particular threshold.



You can also collect logs for various resources.



You can get insights when it comes to resources such as Virtual Machines



You can get reports and even Visualize the data.

Using a Log Analytics workspace



Monitor

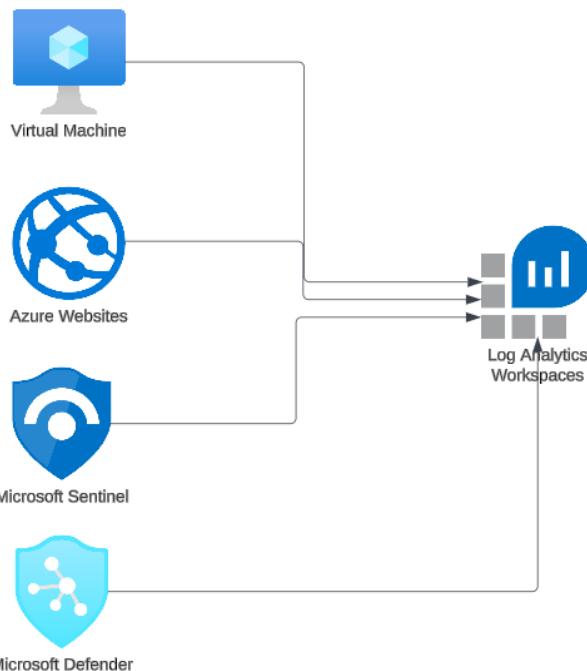


Log Analytics Workspaces

This is an environment that can be used to collect log data.

A single workspace can be used for the collection of data.

Events occurring on the machine.

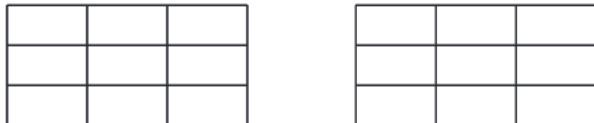


Enable Diagnostics settings to send metrics and log data.

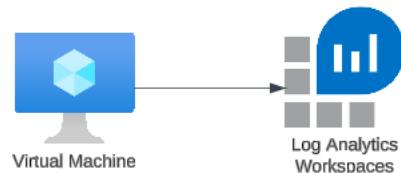
Data collected sent to a Log Analytics workspace.



Within the Log Analytics workspace, the data is collected into tables that have rows of data.



You are charged for the data that is ingested into the workspace and for how long you plan to retain the data.

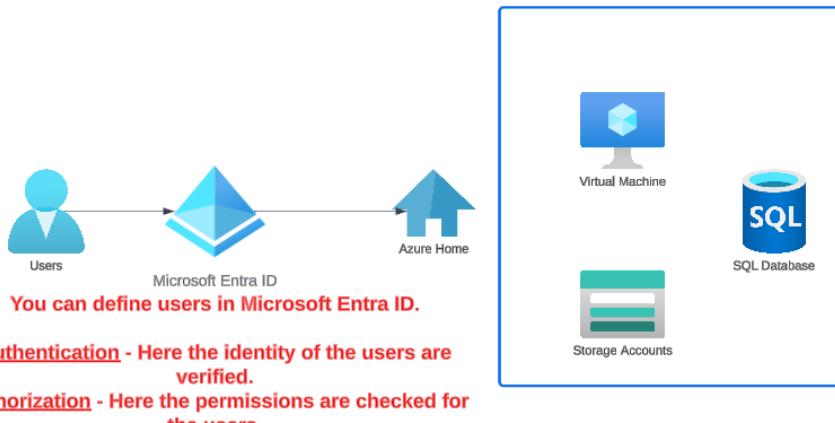


You can have multiple virtual machines that send data to a Log Analytics workspace.

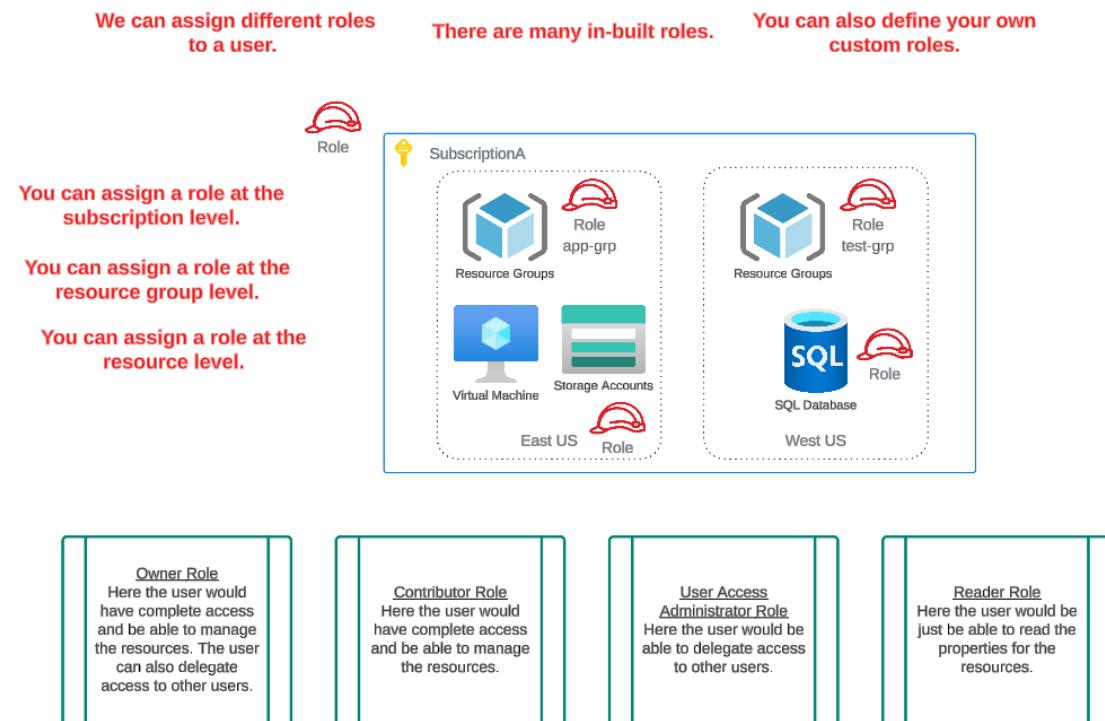
For collecting data from Azure virtual machines , we need to define a data collection rule.

In the rule, we can define the source when it comes to what is the data we need to collect. And then define where to deliver the content to.

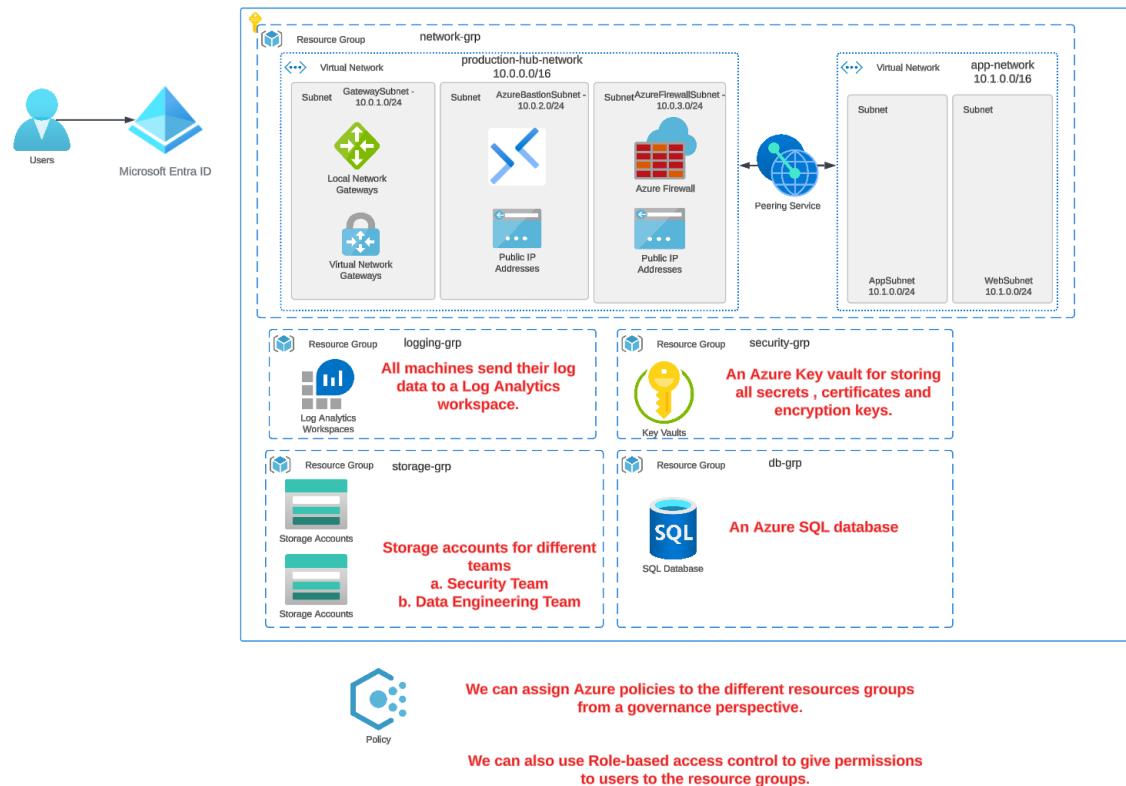
About Azure Role-based access control



Role-based access control



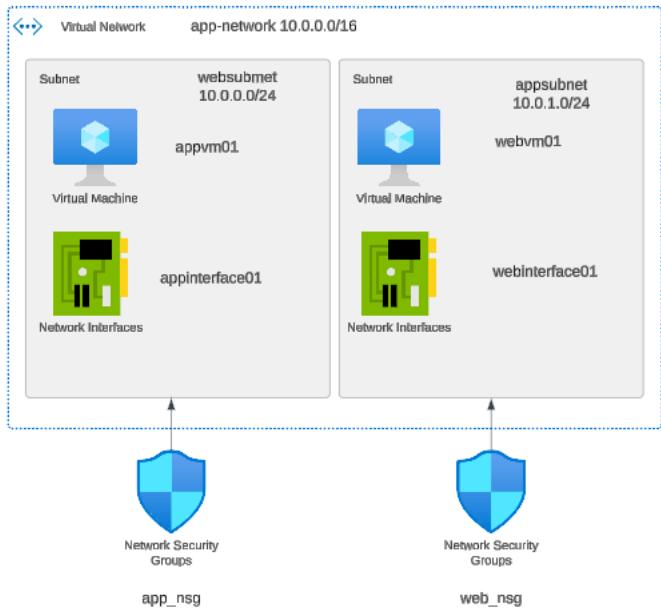
Mini Project - Application Landing Zone



Other aspects when it comes to Terraform

Building infrastructure across multiple environments

We have seen how to use Terraform code to build infrastructure on Azure. We have seen how to use modules to manage reusable code.



Now let's consider the most common scenario where we need to deploy infrastructure across multiple environments.



Lab - Starting with Terraform workspaces



Using git for version control

Git - Version control system



In large organizations, you will have multiple developers working on the source code for an application.

Using Git, you can have multiple versions of your code base. This allows you to easily revert to an older version if there are issues with the newer code-base.



To get started with Git, you first need to install the Git software on your local machine.



Then in the folder where you have your code base you would initialize a git repository.



You can then start making changes to your source code and committing the code changes to your git repository.

Git would make a note on all code changes made to the repository.

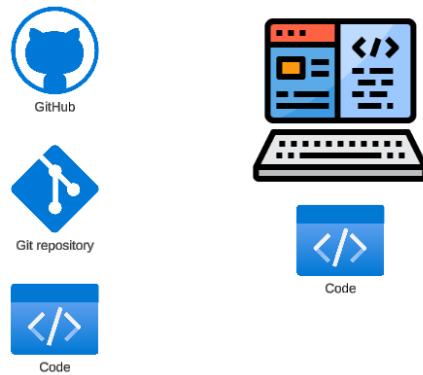


If you want to make your code available to other developers, you can publish your changes to a git-enabled online repository.

GitHub is popular when it comes to an online platform for managing your git-based repositories.

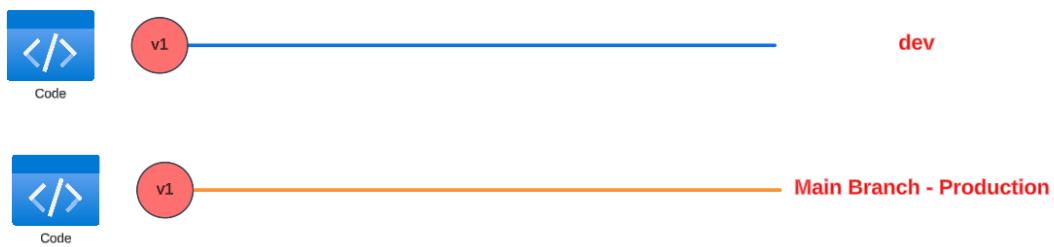
Maintaining different branches

Now our Terraform code is on our local machine and also part of a Git repository on GitHub.



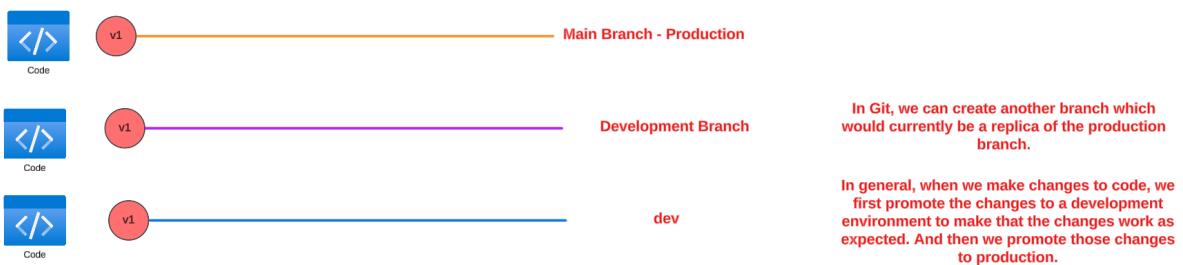
How is code managed across multiple developers and multiple environments.

We make use of Git branches.

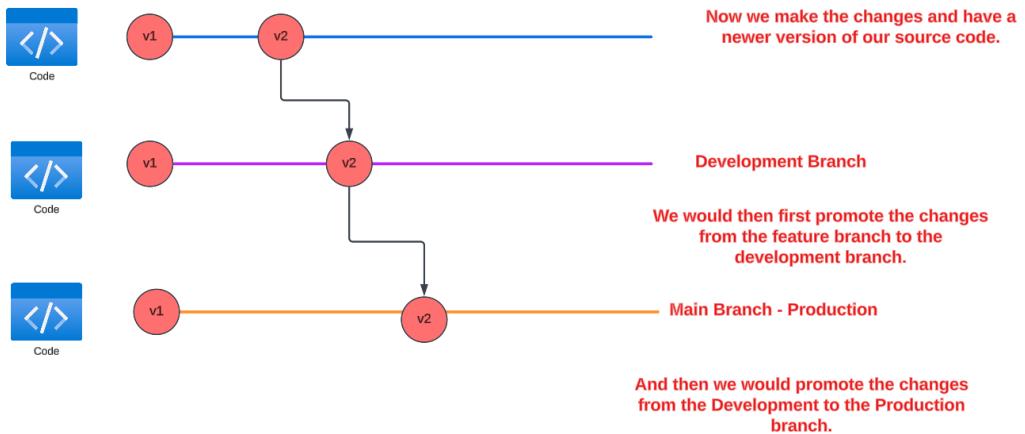


The same code base is also on our dev branch in GitHub.

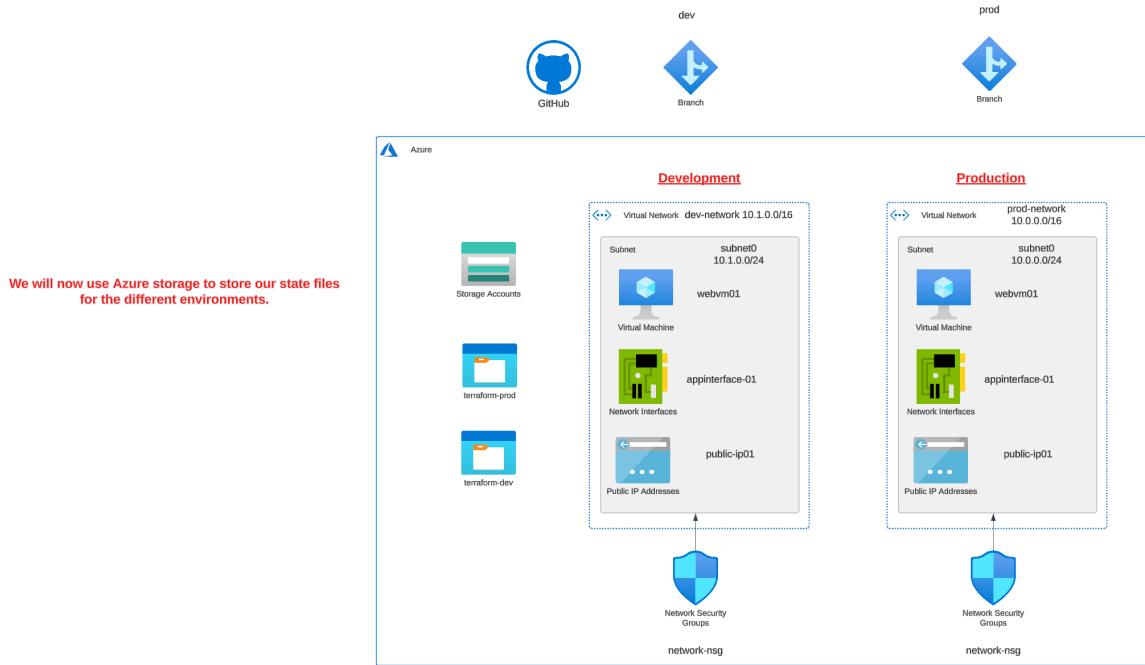
Now let's make this our current production branch , this represents what present in our production-based environment.



Making changes

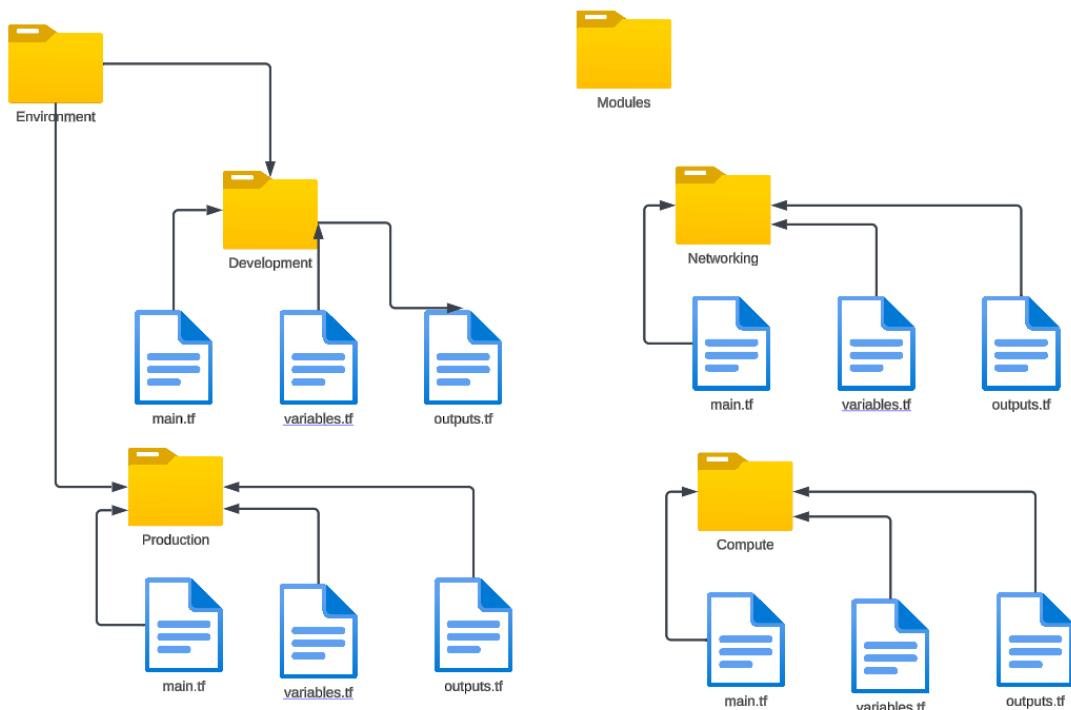


Using Azure storage for maintaining the state file



Note on having multiple folders for multiple environments

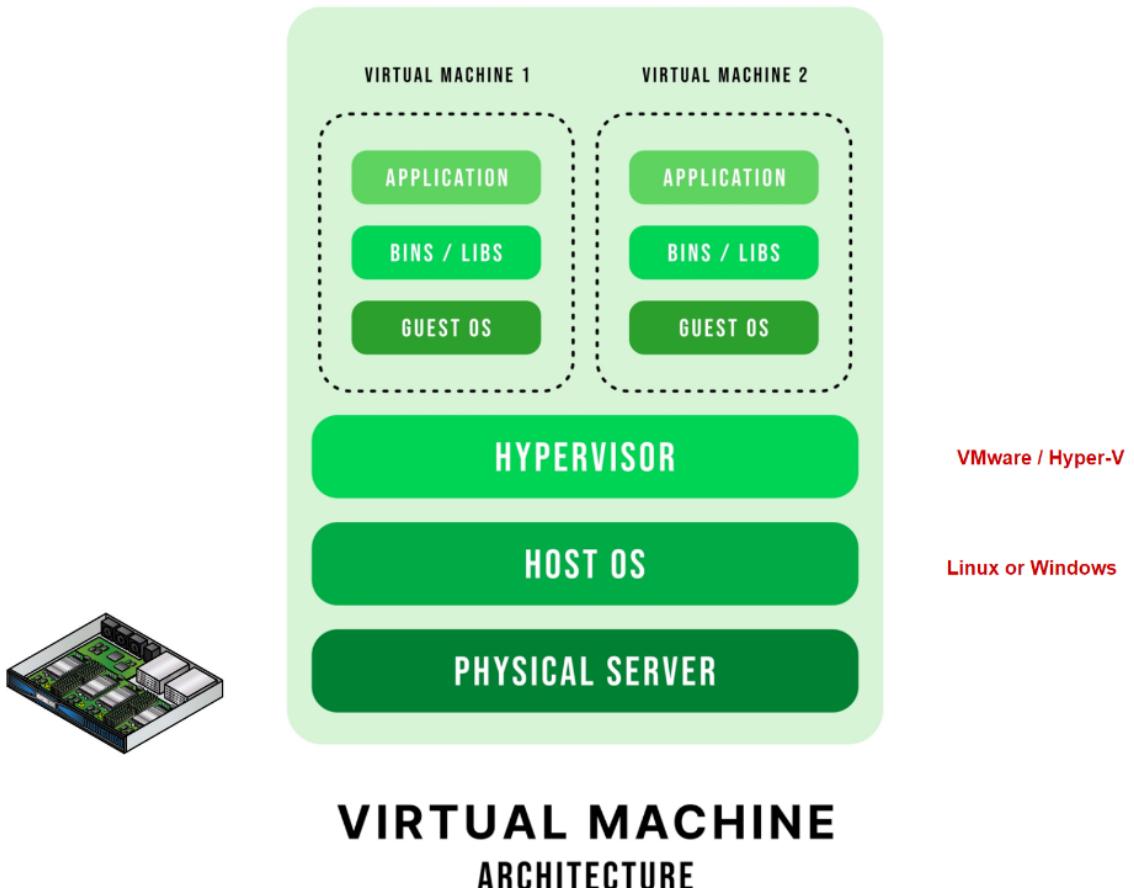
We can also manage environments by maintaining different folders in source control.



Looking into hosting applications within Docker containers

Virtual Machines

These are virtual computers that run on physical machines



Each virtual machine can make use of resources such as CPU/Memory from the underlying physical server.

Each virtual machine is isolated from each other.



Containers

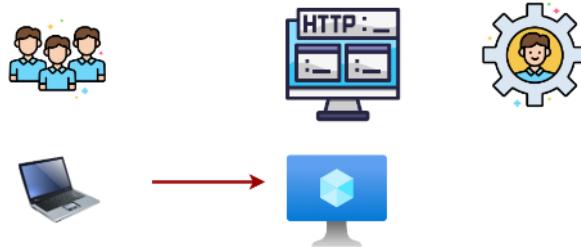


Having virtual machines was a big breakthrough

Companies could host multiple virtual machines on a physical server and make use of the server.

But then there were issues when it came to deploying applications.

Developers develop the application on their machines



When the application is deployed to a virtual machine it does not work as intended.

This could be because of differences in machine software configuration, libraries not present etc.



You have 2 applications on the same machine.

One application update requires a library/component to be installed.

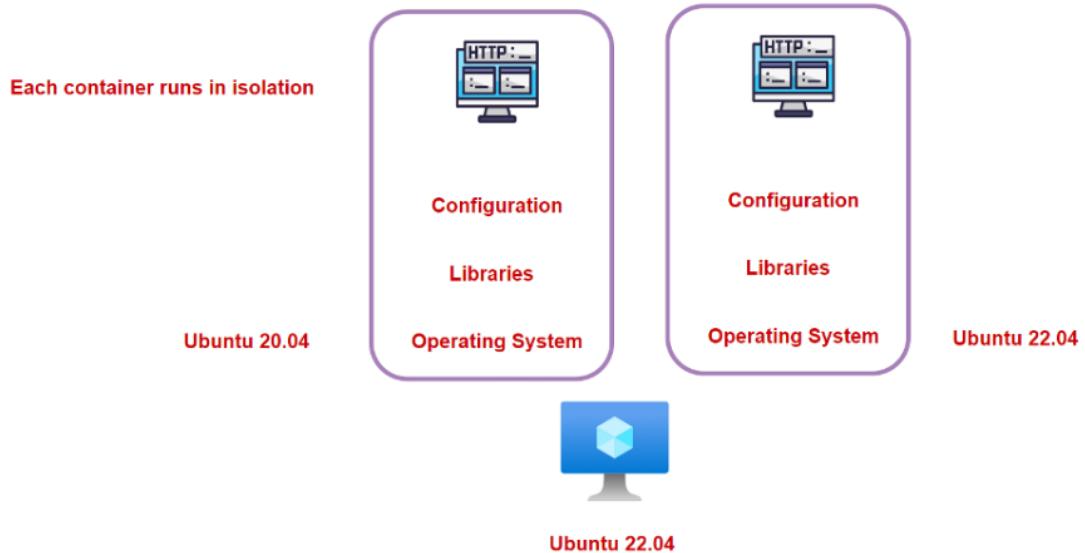
This causes the other application to stop working.

Welcome to containers

This is a unit of software that packages up all the code and dependencies that are required for the application to run.

CONTAINER ARCHITECTURE

The underlying container will have a light-weight operating system, the application, libraries etc.

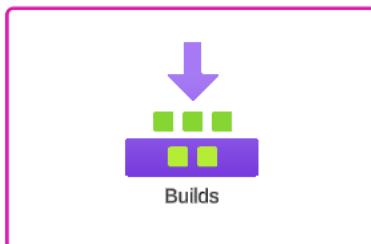


Containerize an application

There are readymade images available on Docker Hub.

You can use these images to run containers.

But let's say we want to run our own application within a container.



We can take our code, create a build and use Docker to first create an image that would contain our application.



We can then run a container out of the image and hence run our application.



We need a place to store our newly created application image. We can store it in Docker Hub. Or use an Azure service known as Azure Container Registry. This is a private registry that can be used for storing your Docker-based images.

Continuous Integration

Continuous Integration

Automatically building and testing code every time a team member commits code changes to version control.

Our local machine

In Visual Studio , when we run an application, a build of the application is conducted first.

The build tool builds the project and the dependencies into a set of binaries.

The binaries can then run on the target machine with the help of the .NET runtime.



Coded an application using the .NET 8.0 SDK



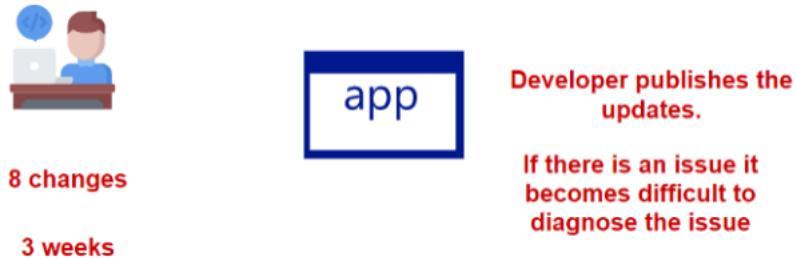
The machine needs to have .NET runtime installed to run the application.

Deploy the application on a virtual machine



You normally have multiple developers working
on an application

Each developer might be working on multiple
changes that can take time.



Automation is key



And this is the concept behind
continuous integration