Name: - AYUSH PANDEY
BATCH: - AZ-900 + AZ- 104 (12th July 2025)

Assignment 04: - Documentation of Front Door and Traffic Manger

Front Door

- What is Front Door?
  Azure Front Door is like a **main entrance gate** for your web apps, websites, or APIs.

  - It sits in front of your application and makes sure users get connected **fast, securely, and reliably**.
  - It automatically directs users to the **nearest and fastest server** (using Microsoft's global network).
  - It also protects your app from attacks like **DDoS and web threats**.
  - If one server or region goes down, Front Door will **send traffic to another healthy server** so your app stays available.

Steps to Create Azure Front Door

- Sign in
- Go to the Azure Portal.
- Create a Front Door resource
- Search for Front Door in the portal.
- Click Create.
- Select your **subscription** and **resource group**.
- Give your Front Door a name.

Tier:

1. **Standard Tier: -**

   - Think of this as the basic package.
   - Helps your app load faster worldwide.
   - Protects with basic security (like DDoS protection).
   - Supports global load balancing (chooses the nearest/fastest backend).

2. **Premium Tier: -**

   This is the advanced package.

   - Includes **everything in Standard**.
   - Plus, **extra security**:
   - Web Application Firewall (WAF) with advanced rules.

- **Private Link** (connect securely without exposing your app to the internet).
- **Advanced analytics & reporting**.

3. **Pricing: -**

- Front door Standard: - **$35**
- Front door Premium: - **$330**

Secrets: -

In Azure Front Door, Secrets usually mean SSL/TLS certificates that keep your website traffic secure (HTTPS).

- When users open your site (like https://myapp.com), Front Door needs a certificate (secret) to prove the site is safe and encrypted.
- These secrets are stored in Azure Key Vault (a secure storage for keys, passwords, and certificates).
- Front Door connects to Key Vault, fetches the secret, and uses it to enable HTTPS on your custom domain.

Secrets in Front Door = The lock-and-key (certificate) that makes your website safe (HTTPS).

Endpoint: -

An Azure Front Door endpoint is the public address (URL) that people use to reach your app through Front Door.

- When you create a Front Door, Azure gives you a default endpoint like:
  👉 https://myapp.azurefd.net
- You can also add your own custom domain, like:
  👉 https://www.myapp.com

Create the **hostname** (like myapp.azurefd.net) that users will use to access your app.

Add a Route: -

“Add a Route” = Make a rule that tells Front Door where to send the traffic.

Origin Group: -

- “Origin Group” = A team of backend servers where Front Door decides which one to use, based on health and performance.
- “Origin” = your actual backend app/service (like App Service, VM, Storage, or external site).
- “Origin Group” = a pool of these origins working together.

Weight: -

- Weight is a number you assign to each origin inside an origin group.
- It decides how much traffic (percentage of users) goes to each origin.
- Higher weight = more traffic**.**

Example: -

Origins:

- A = 50 weight
- B = 30 weight
- C = 20 weight

Total weight = 50 + 30 + 20 = 100

A → (50 ÷ 100) × 100 = 50% traffic

B → (30 ÷ 100) × 100 = 30% traffic

C → (20 ÷ 100) × 100 = 20% traffic

Session Affinity: -

- Session Affinity = also called **"sticky sessions."**
- It means that once a user connects to your app through Front Door, **all their requests keep going to the same backend server** instead of switching between different servers.

Health Probe: -

A **health probe** is like a **doctor check-up** for your backend servers.

- Front Door keeps sending small requests (pings) to your servers.
- If the server **responds correctly**, it is marked **healthy**.
- If the server **does not respond** or is **too slow**, it is marked **unhealthy**.

Load Balancing: -

- **Distributes Traffic**
  Sends user requests across multiple backend servers so no single server is overloaded.
- **Based on Health Probes**
  Only sends traffic to healthy servers (checked using health probes).
- **Load Balancing Methods**
  Priority → Always send traffic to the main server first, use backup only if main fails.
  Weighted → Divide traffic by weight numbers (example: 70/30 split).
  Latency-based → Send users to the server with the lowest network delay (closest/fastest).
- **Global Reach**
  Uses Microsoft's worldwide network to route users to the nearest backend region for faster response.
- **Session Affinity (Optional)**
  Keeps the same user's requests going to the same backend server (sticky sessions).

- **Failover Support**
  If one region/server goes down, traffic is automatically rerouted to another healthy one.
- **Scalability**
  Can handle sudden traffic spikes by spreading load across multiple servers/regions.

Origin Path: -

Origin Path is an **extra folder/path** that Front Door adds automatically when sending traffic to your backend (origin).

It's like telling Front Door:

"Whenever you send a request to this server, always go inside this folder first."

Example: -

Frontend request: https://www.myshop.com/products
Backend origin: https://mybackend.azurewebsites.net
Origin Path = /store

Then Front Door will send the request to:
https://mybackend.azurewebsites.net/store/products

Traffic Manager

It helps you direct (or route) user requests to the best available server or service across the world. It does not handle the traffic itself — instead, it tells the client which server to go to base on the rules you set.

Key points in simple words:

- DNS-based: It works by giving the client the IP address of the best endpoint (server/service).
- Global load balancing: Routes users to servers in different regions.
- High availability: If one region/service goes down, Traffic Manager can redirect traffic to another healthy one.
- Routing methods: You can choose how to direct traffic (performance, priority, weighted, geographic, etc.).
- Health checks: It keeps checking your endpoints (like VMs, Web Apps, or APIs) to make sure they are up.

Steps to Create a Traffic Manager Profile: -

- Sign in
- Go to the Azure Portal.
- Create a resource group
- Search for **Traffic Manager Profile**.
- Click Create.
- Select your **subscription** and **resource group**.

Basics Tab

- Name → give a unique name (this becomes your DNS name like myapp.trafficmanager.net).
- Routing Method → choose how traffic will be distributed
    - **Priority** → use main server, others as backup.
    - **Weighted** → split traffic by percentage.
    - **Performance** → route to the nearest/fastest server.
    - **Geographic** → send users to servers based on their location.
    - **Multivalue** → return multiple healthy endpoints.
    - **Subnet** → send certain IP ranges to specific servers.

Configuration

- Choose the **DNS Time-to-Live (TTL)** (default is fine, usually 30 seconds)

.

Review + Create

- Check all settings.
- Click **Create** to deploy.

Add Endpoints (after profile is created)

- Go inside your new Traffic Manager profile.
- Select Endpoints → Add.
- Add your Azure service (App Service, VM, etc.) or an External endpoint.
- Repeat to add more servers if needed.

Test Your Traffic Manager DNS Name

- Try accessing myapp.trafficmanager.net.
- It should route traffic as per your chosen method.
- Later, you can map your custom domain (like www.myapp.com) to this Traffic Manager DNS.
