

DEVOPS with MULTI-CLOUD

Practice Tasks

Institute Name : V Cube software solutions
Course : DevOps with Multi-Cloud
Batch : 30
Trainer : Krishna reddy sir

Prepared by : G.Bhavish
(MCD-AZ30-024)

TASK-10 : Application Gateway.

Date : 02/02/26

Objective :-

To securely distribute incoming web traffic across multiple backend servers while providing advanced Layer-7 (HTTP/HTTPS) routing, security, and high availability for web applications.

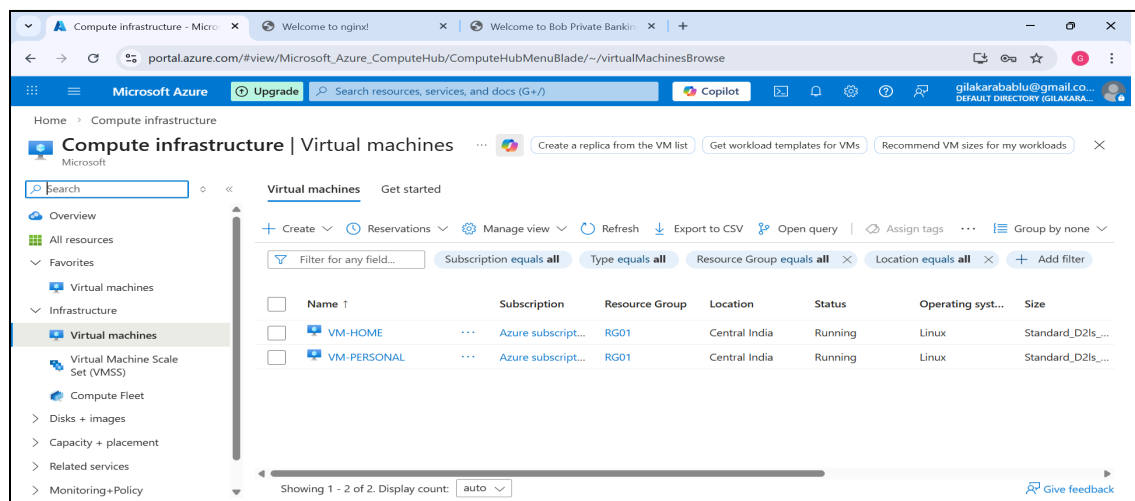
Application Gateway :-

- The application gateway supports regional load balancers and it routes the traffic based on the URL path routing.
- The Application gateway is used when the work loads are in different servers unlike the load balancer which routes the traffic to identical machines.
- It works on OSI layer 7 (application) and supports http and https based applications. It also supports WAF(web application firewall).
- The application gateway has the capability to inspect the packet like what kind of traffic it is and based on that it will route the traffic and it will equally distribute the traffic in the backend servers.
- The prerequisite of the application gateway is that it should have a dedicated SN to itself.

To Implement the Application Gateway :-

→ Our requirement is if we browse the web homepage and in the path if we type /personal/ we should go to the personal page.

→ First create two vm's, vm-home and vm-personal under rg01 with vn01 in sn01, also create a separate dedicated subnet to application gateway sn-ag.



fig(1) successfully created two vm's

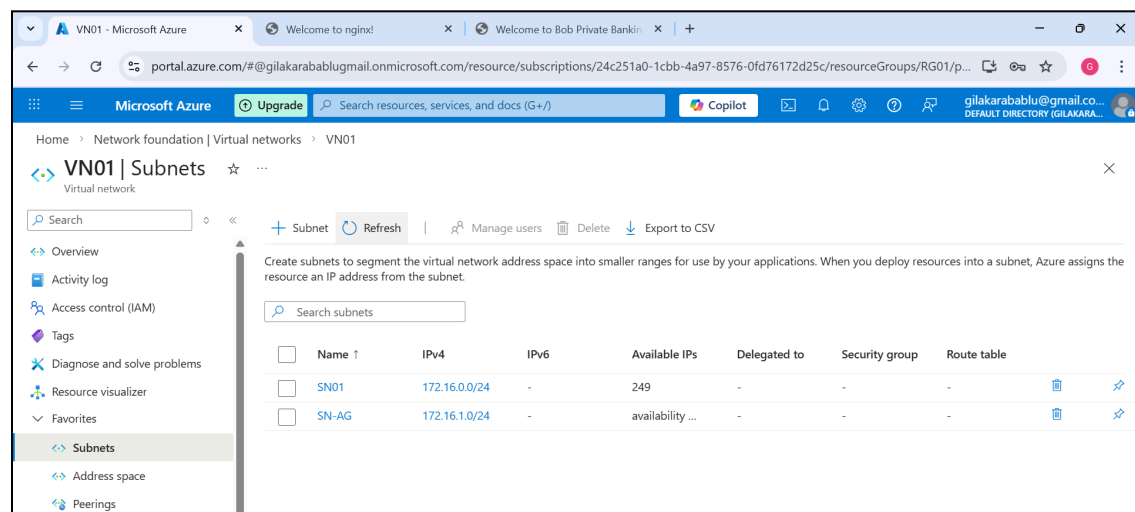


Fig (2) created a dedicated subnet to the application gateway.

→ Install nginx in both the web servers.
→ For our requirement we should create a personal folder beside the path of the home then the html file will move to the personal webpage.

→ now login to the personal vm and create a folder and also change the content inside the personal webpage.

→ login and execute the following commands.

- Sudo su (for root user)
- Cd /var/www/html/ (home path)
- Mkdir personal (creating folder)
- Ls (to list the files)
- Pwd (present directory)
- Mv index.html.file personal/ (for moving files)
- Cd personal/
- Ls (to list the files)

→ Now if we browse the homepage and type /personal/ we get the personal page.

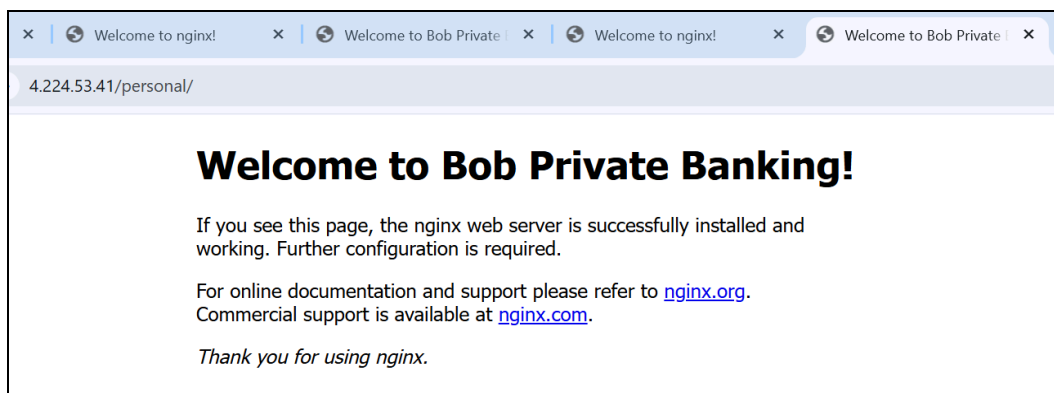
→ Now to change the content :

- Vi index.nginx.file (to edit the file)
- Press i (to go insert mode)
- Make the required changes.....
- Esc : wq (for write & quit)

→ Now refresh the page we will get the edited content.



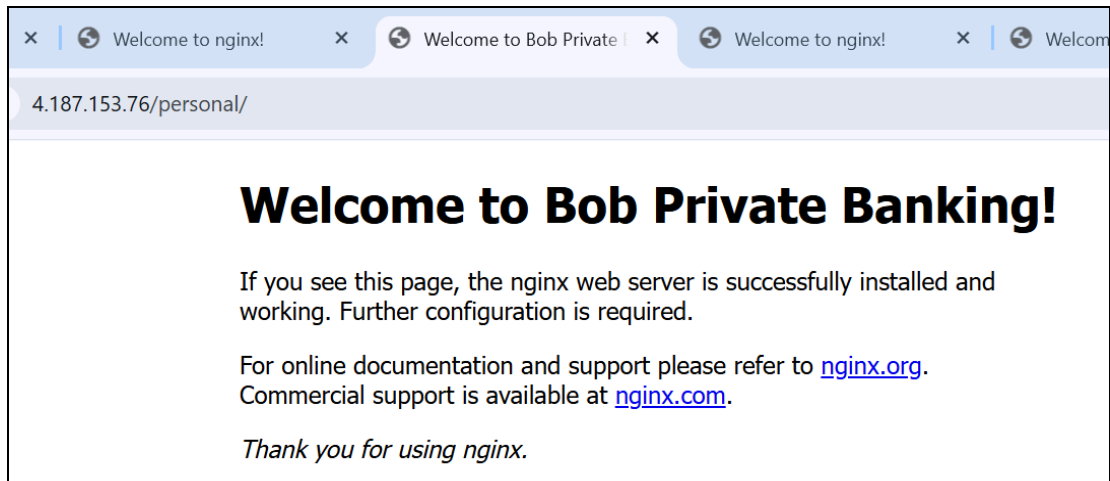
fig(3) Homepage before the creation of a personal path.



fig(4) personal page before giving the personal path.



fig(5) homepage after the personal path.



fig(6) personal page after the path.

→ Now create the Application Gateway :-

- While creating the application gateway first give the dummy details and port numbers and after creating we can edit the details with the original ones.
- Like doing so we can exclude the errors like “health probs” and “give the path of the personal page to the Application Gateway”.

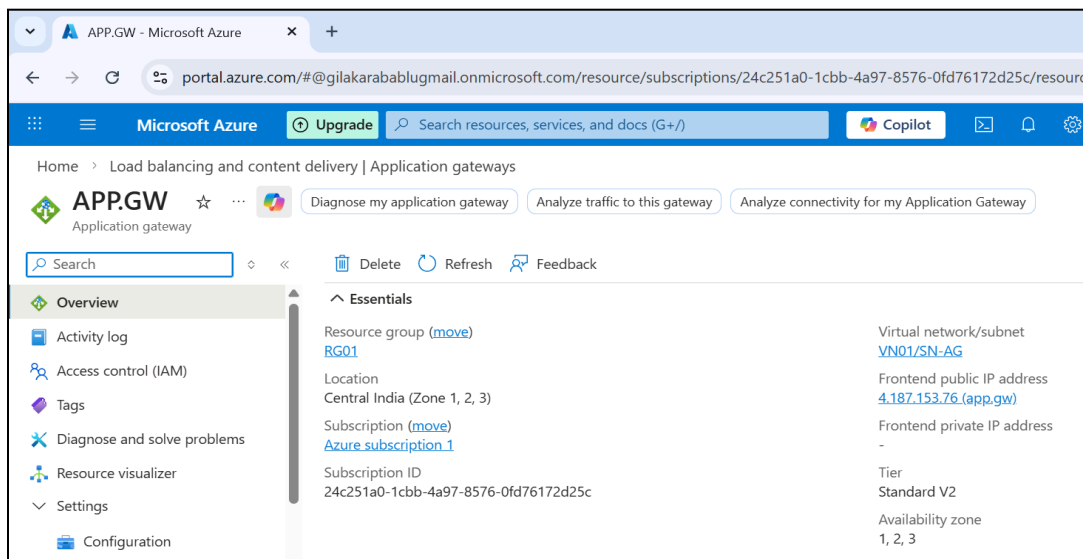


Fig (7) successfully created an application gateway.

→ Now after configuring all the original details, health probs and rules.

→ If we browse with the frontend-ip of the Application Gateway we get the homepage and if we type /personal/ in the homepage path we will go to the personal page.

→ Here, the homepage and the personal page are different servers but still the Application Gateway is able to route the traffic. So like this the Application Gateway can identify the traffic and routes to its required destination.