**Load Balancers**

**A load balancer** is a device or software that distributes incoming network traffic across multiple servers. This distribution ensures that no single server is overwhelmed (overloaded) with requests, leading to improved performance, reliability, and availability of your applications.

**Benefits of using a load balancer:**

* **Improved Performance:** By distributing traffic across multiple servers, a load balancer can handle more requests concurrently, reducing latency and improving response times.
* **Increased Availability:** If one server fails, the load balancer can automatically redirect traffic to other available servers, ensuring that your application remains accessible.
* **Scalability:** Load balancers can easily accommodate increasing traffic demands by adding more servers to the pool.
* **Resource Optimization:** Load balancing ensures that all servers are utilized efficiently, preventing any single server from becoming a bottleneck.

In Azure there are four types of load balancers:

1. Azure Load Balancer
2. Application Gate Way
3. Traffic Manager
4. Front door
5. **Azure Load Balancer**

* It is also known as Network Load Balancer.
* It can supports both Regional and Globally. That mean if over VM’s present within region or outside the region it can be distributes Network traffic easily.
* It works as proxy based routing.

**Proxy-based Routing** is a network architecture where a **proxy server** acts as an intermediary between clients and the actual servers they want to access. Instead of clients directly connecting to the target servers, they send their requests to the proxy server.

* It operates at the Transport Layer (Layer 4) of the OSI model.
* It supports TCP & UDP protocol.
* It sends the request to the Backend pool (where servers are present) using Round Robin method.

**How Round Robin Works in Azure Load Balancer:**

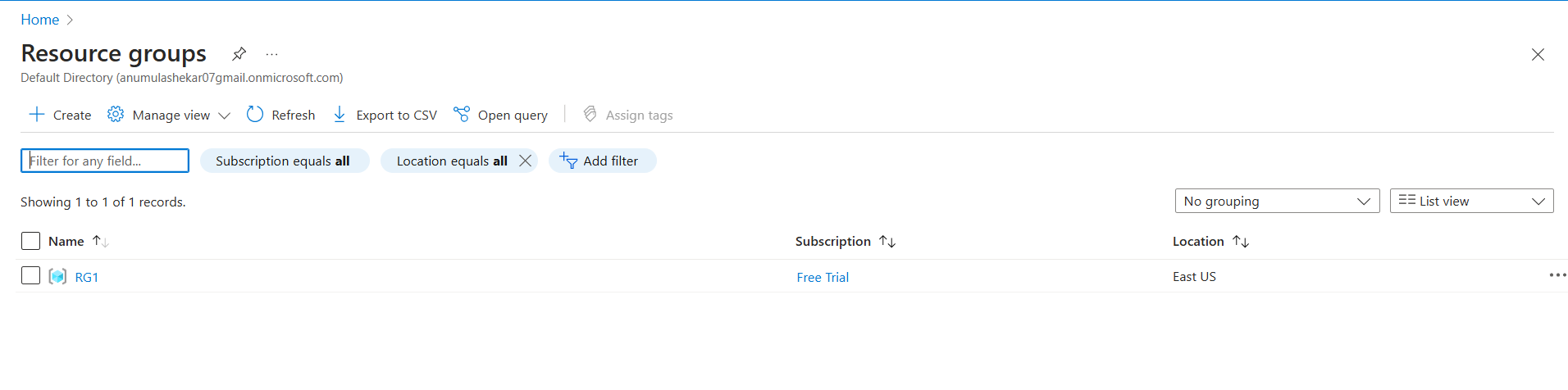
1. **Server List:** The load balancer maintains a list of available backend servers (virtual machines).
2. **Sequential Distribution:** When a request arrives, the load balancer sends it to the next server in the list.
3. **Cycling:** Once it reaches the end of the list, it starts again from the beginning, ensuring all servers receive traffic in a cyclical manner.

**Note1:** While creating Load Balancer we will get a Public IP which is assigned to Load Balancer.

**Note2:** VMs or servers connect to a load balancer using their private IP addresses.

Let’s experience the perform of Azure Load Balancer

**Step1:** Create the resource group RG1



**Step2**: Create TWO VM’s in same region of single Vnet. (VM-1-Vnet )

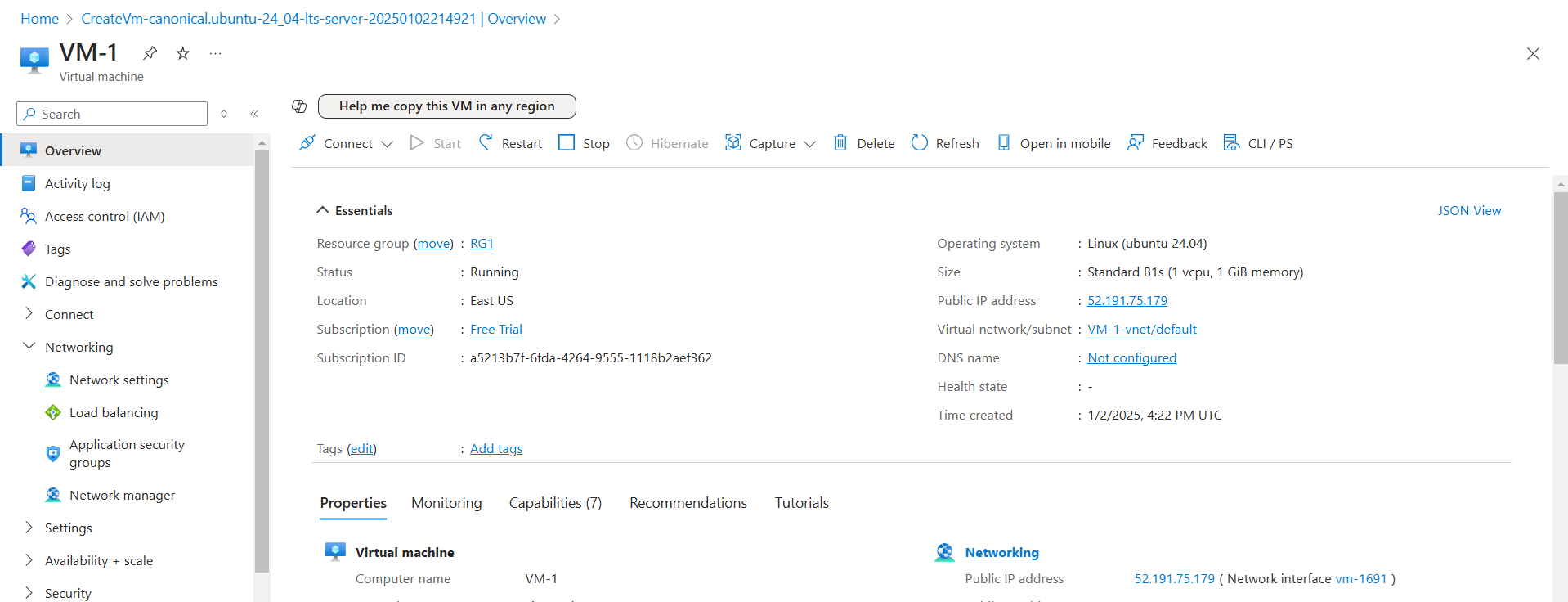


  Fig: VM-1 in the region East US.

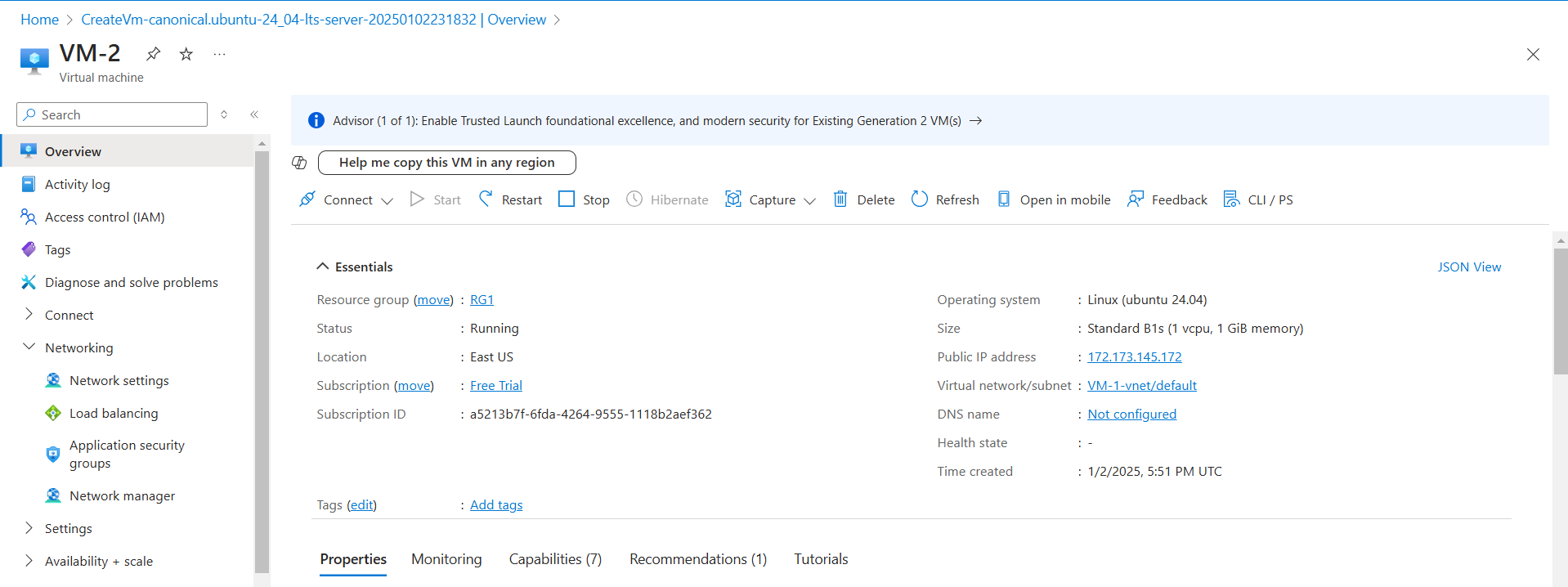
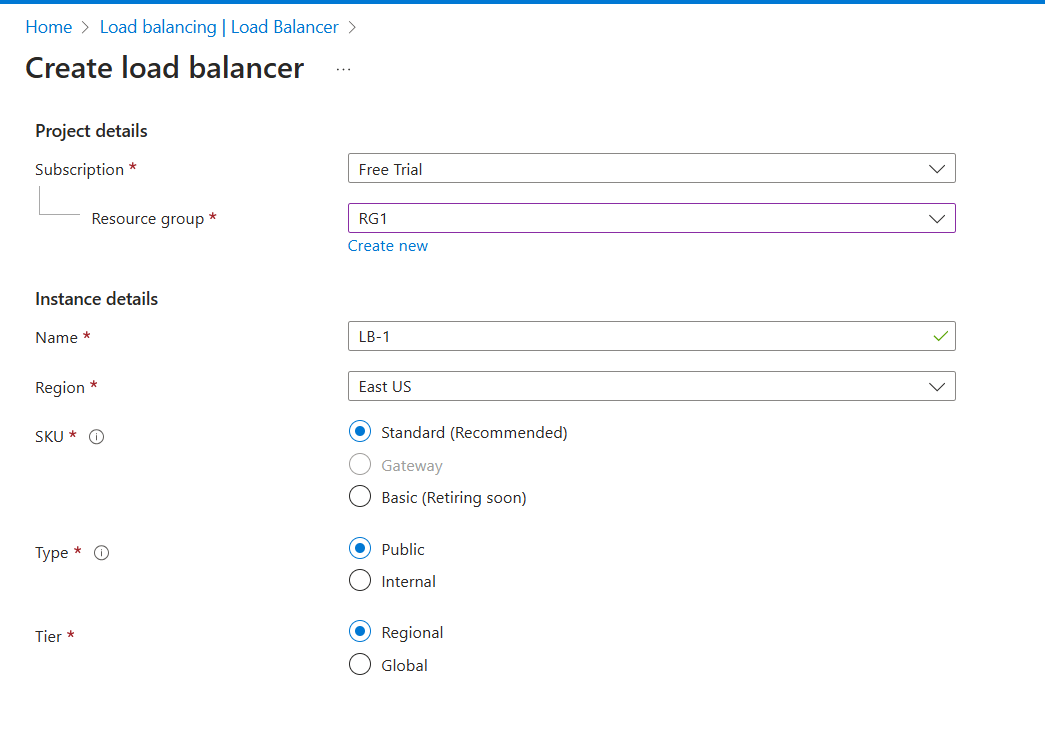


Fig: VM-2 in the region East US.

**Step3:** Create the Azure Load Balancer.

Note: Azure Load Balancer can support regional and globally



While creating Load Balancer Configuring of Frontend IP is Essential.

**Frontend IP Configuration** in Azure Load Balancer refers to the IP address (es) that clients use to access your application. It's essentially the "public face" of your load balancer.

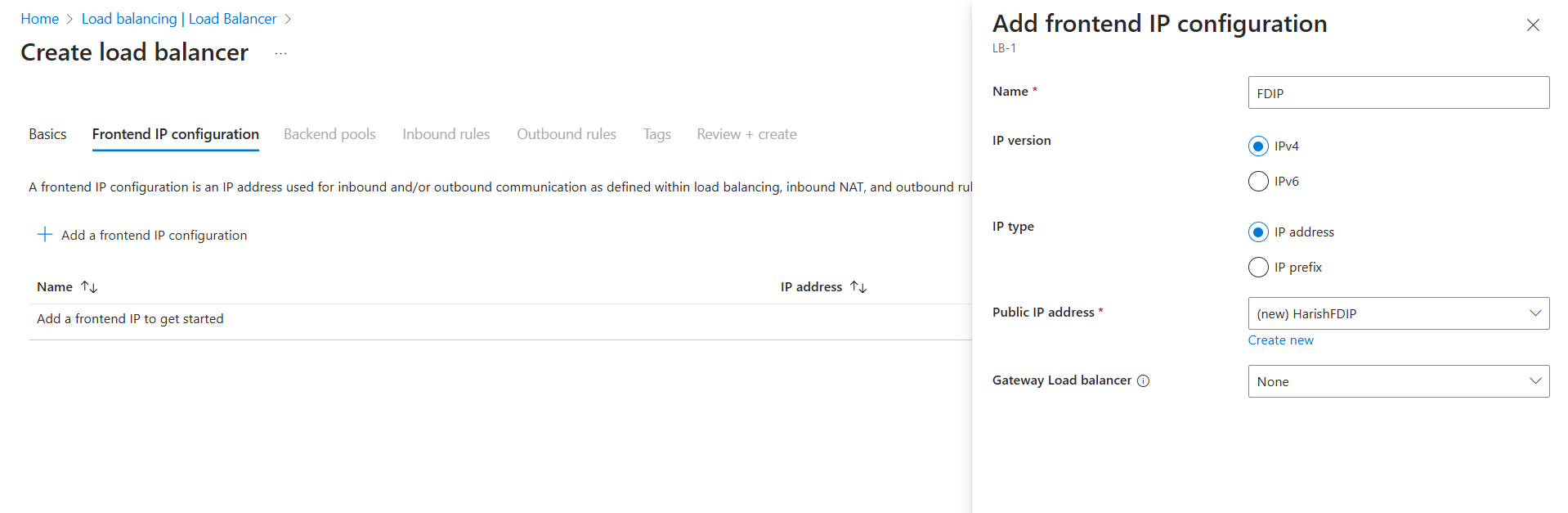


Fig: Frontend IP configuration of Load Balancer.

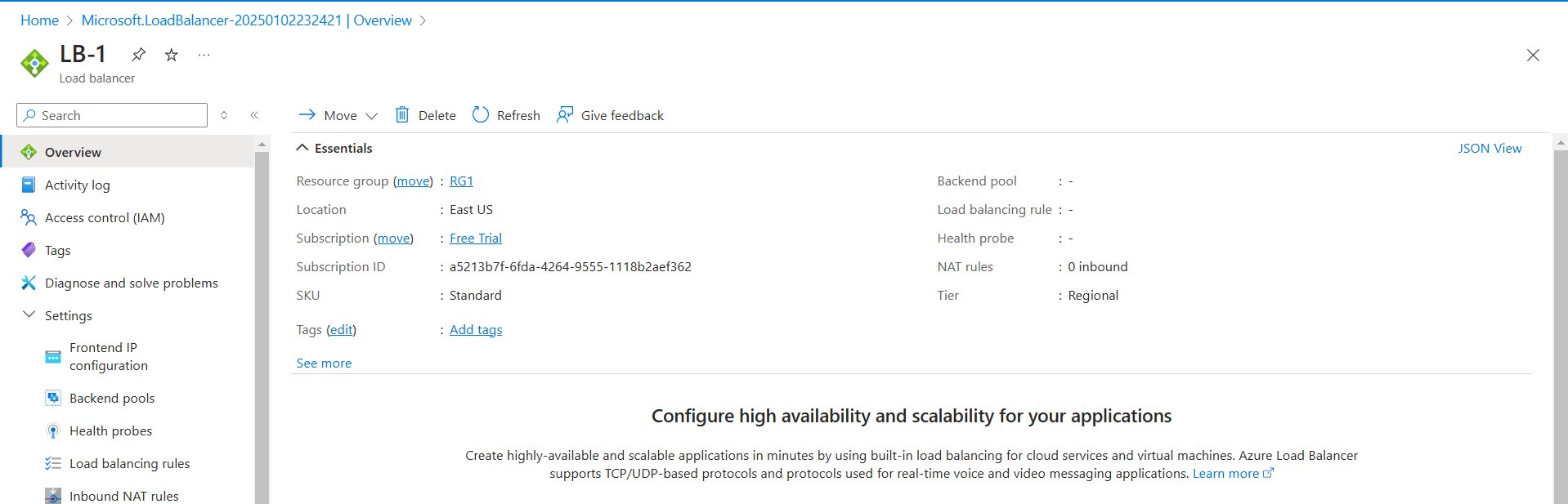
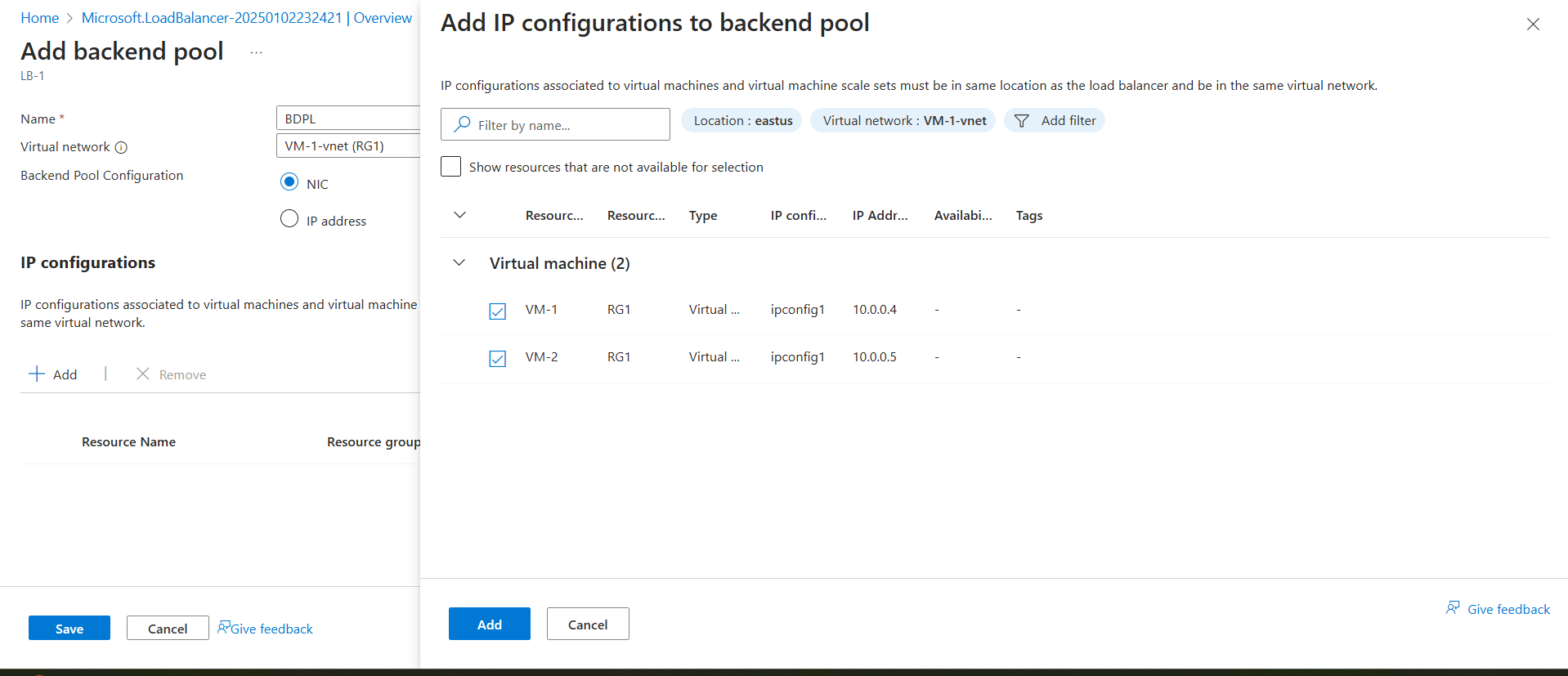


Fig: Load Balancer (LB-1)

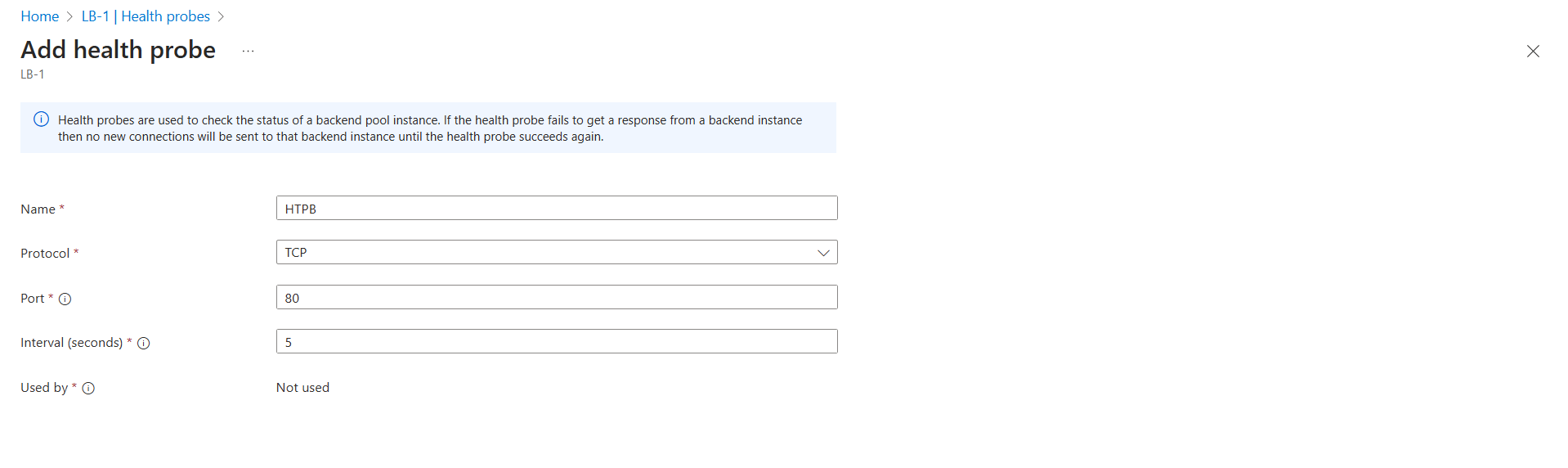
Step4: Add the VM’s using the Backend pool.



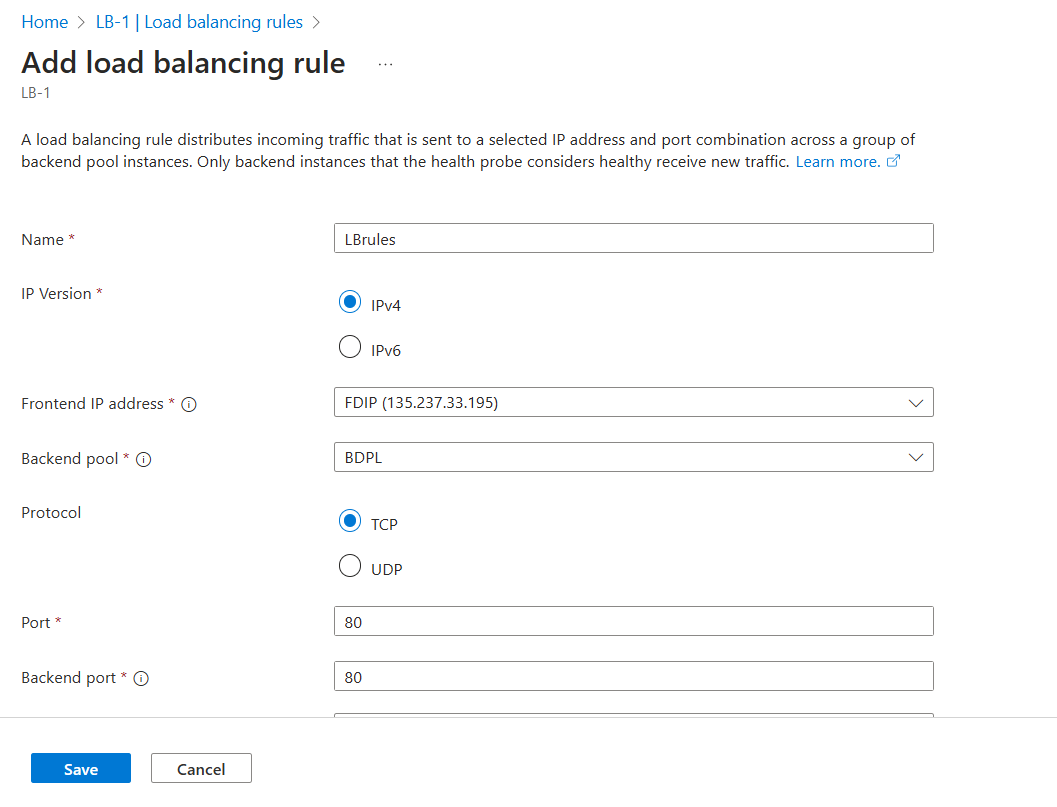
Step5: create an Health probe

These probes act as vigilant guardians, continuously monitoring the health of your backend servers (virtual machines) to ensure they are functioning correctly.

That means it regularly checks the heat beat of the backed servers by sending dummy massage packets to the each VM (server).



Step6: Add the Load Balancer Rules.



Step7: now copy the IP of Load Balancer and brows it.

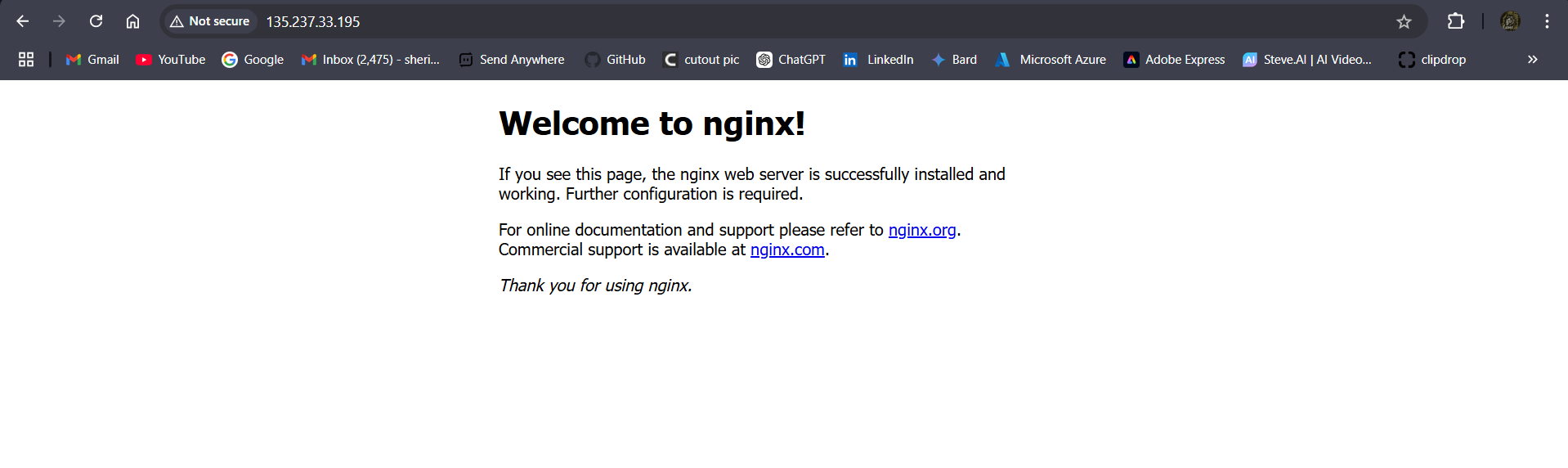
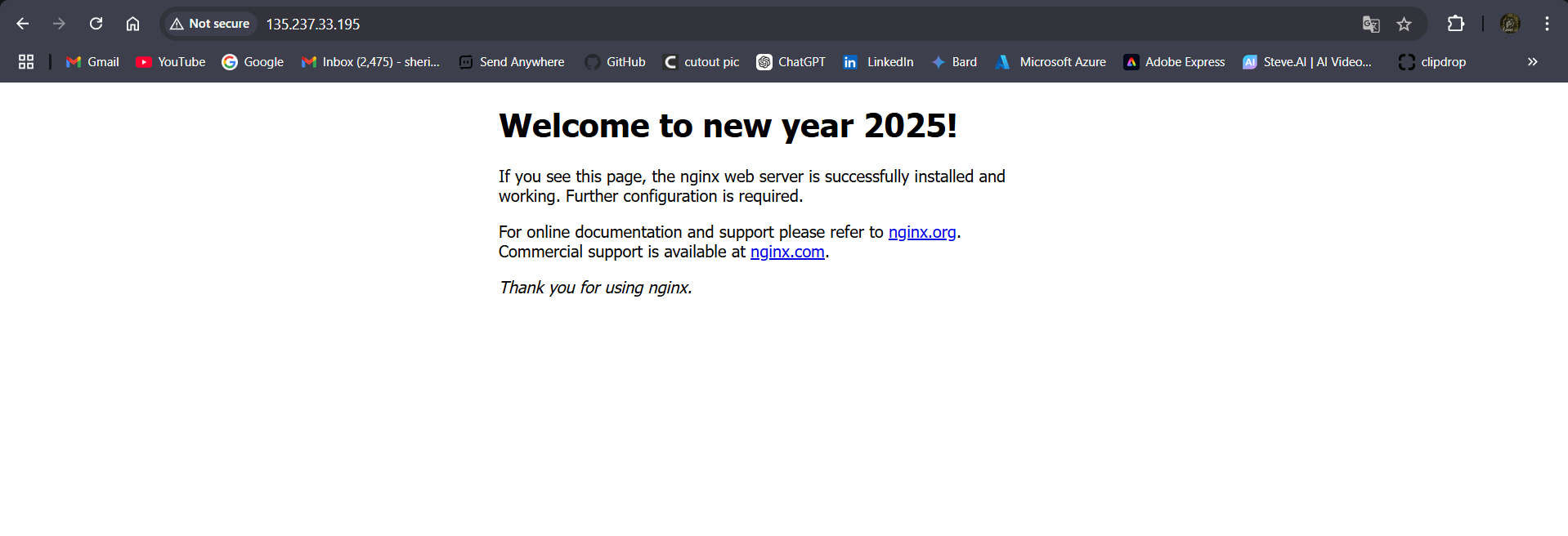


Fig: Server (vm-1) application.

Similarly let’s check if VM-1 is down and then traffic is routed to the VM-2 by the Load Balancer as show below figure.



Block Diagram:

Public IP

VM-1-Vnet

VM-2

VM-1

LB-1

Load Balancer