WEB APPLICATION FIREWALL ON LOAD-BALANCER(OCI)

1. What is OCI?

- OCI, or Oracle Cloud Infrastructure, is Oracle's cloud computing platform. It offers a range of services for deploying and scaling applications with high performance and security. OCI is known for its global presence, robust infrastructure, and support for diverse workloads, making it a preferred choice for enterprises.

2. What is OCI Security?

- OCI security involves protective measures in Oracle Cloud Infrastructure to safeguard data and resources from unauthorized access and cyber threats, using features like encryption and identity management.

3. What is WEB APPLICATION FIREWALL?

- A Web Application Firewall (WAF) in Oracle Cloud Infrastructure (OCI) is a security solution that protects web applications from various online threats and attacks by monitoring, filtering, and blocking malicious traffic before it reaches the application.

4. Importance of the WEB APPLICATION FIREWALL?

- **Protection**

- **Security Compliance**

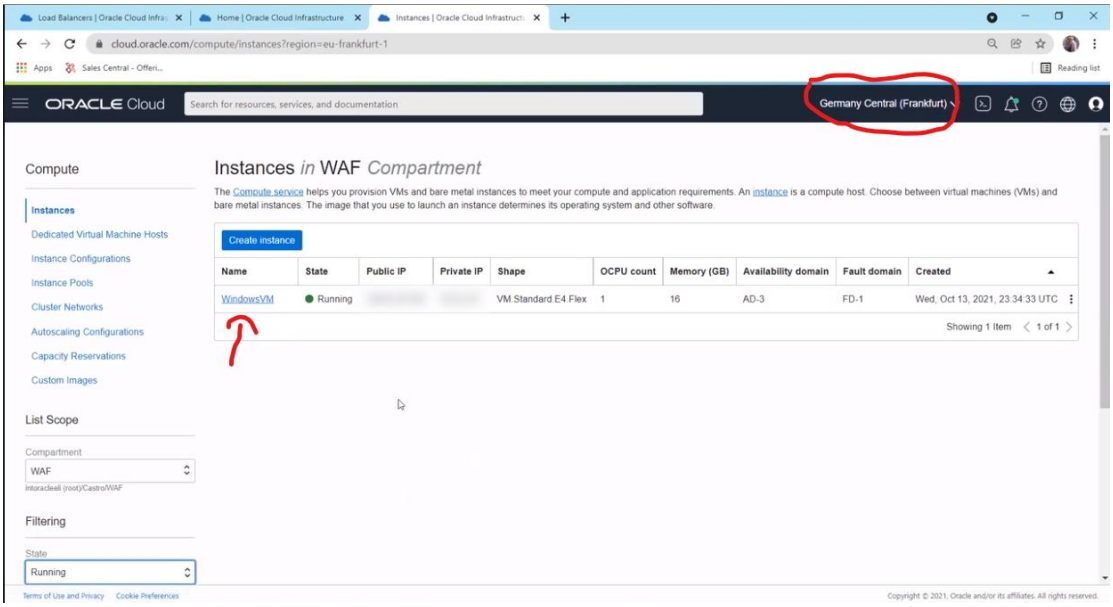
- **Threat Detection**

- **Performance Optimization**

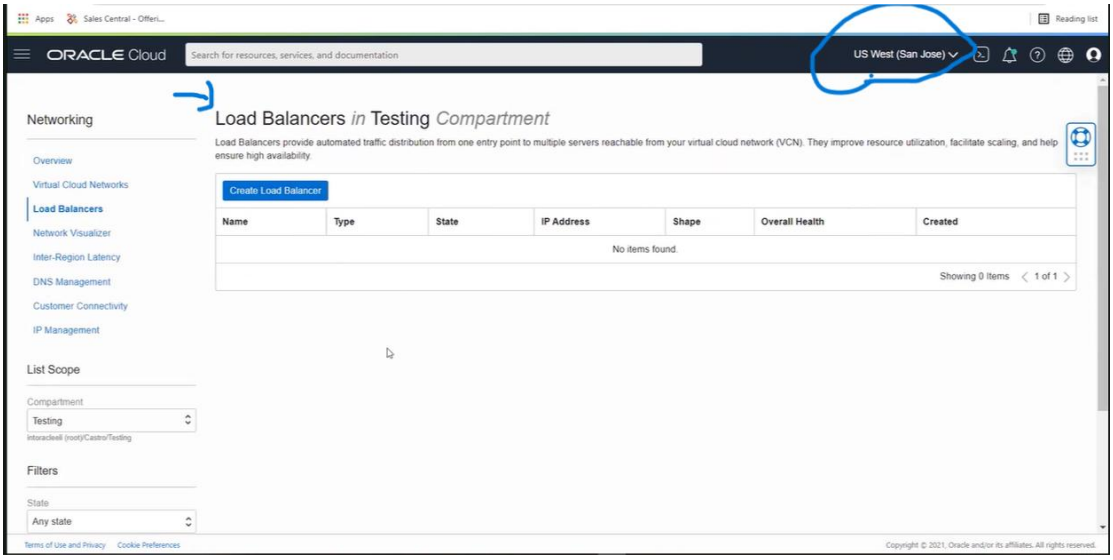
- **Mitigation of DDoS Attacks**

- **Customization Security Policies**

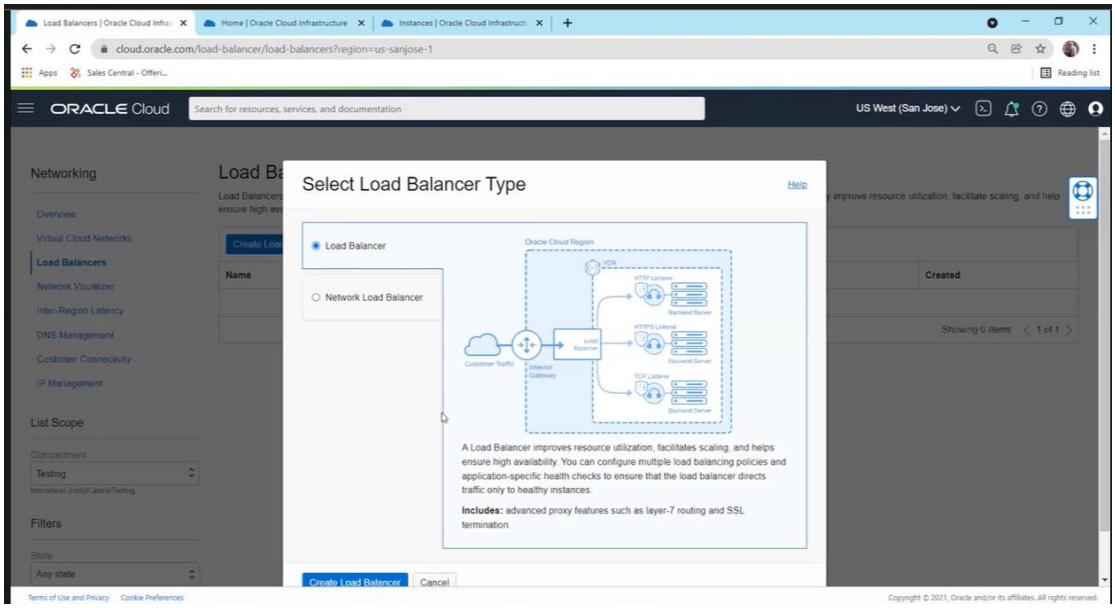
STEP 1 :- We will work with 2 tenancy in this, at our first tenancy we have a VM in german



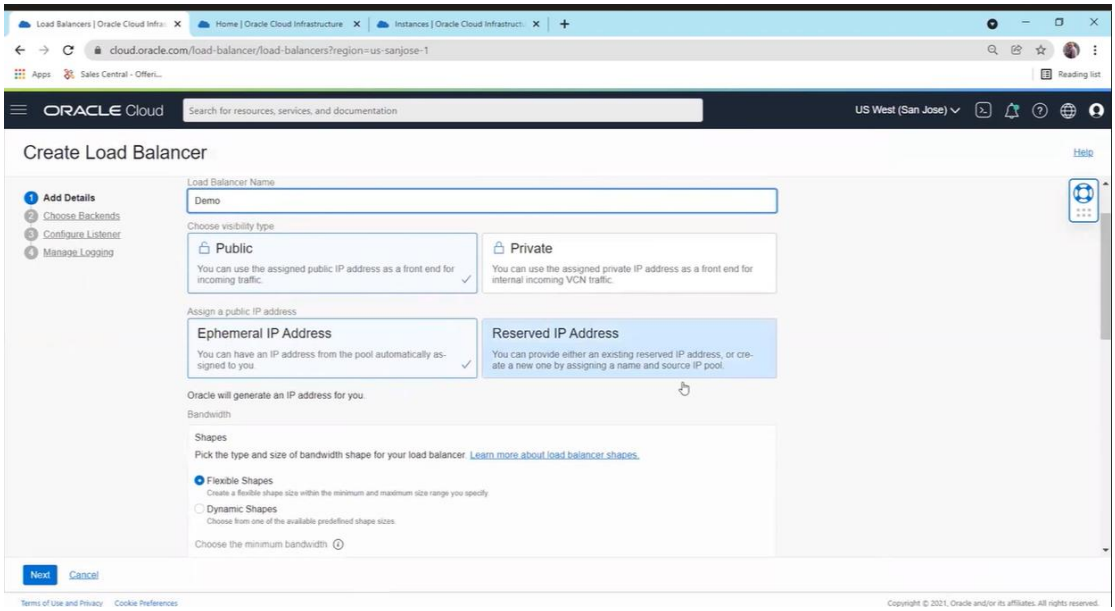
STEP 2 :- In our Second tenancy which is in US we will create a Load Balancer



STEP 3 :- Click on create Loan balancer and choose the 1st option
Load balancer = Application layer (layer 7)
Network Load balancer = Network Layer (layer 3)



STEP 4 :- Give the name as required, choose between public and private, and assign IP from Ephemeral IP or Reserved IP



Public vs. Private Load Balancer:

Public Load Balancer: This type of load balancer is accessible from the internet, allowing external users to send requests to your applications or services. It's commonly used for public-facing applications or services that need to be accessed by users over the internet.

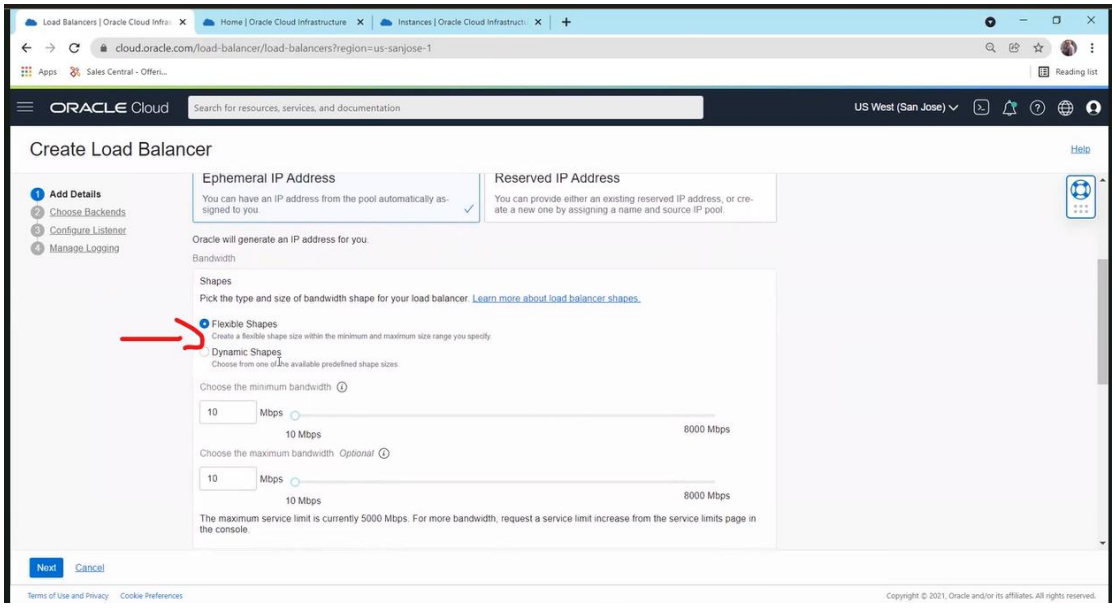
Private Load Balancer: Unlike a public load balancer, a private load balancer is not directly accessible from the internet. It's deployed within a private subnet and is primarily used for internal communication between resources within your virtual cloud network (VCN). Private load balancers are suitable for applications or services that do not need to be exposed to the public internet, enhancing security by limiting external access.

Ephemeral IP vs. Reserved IP:

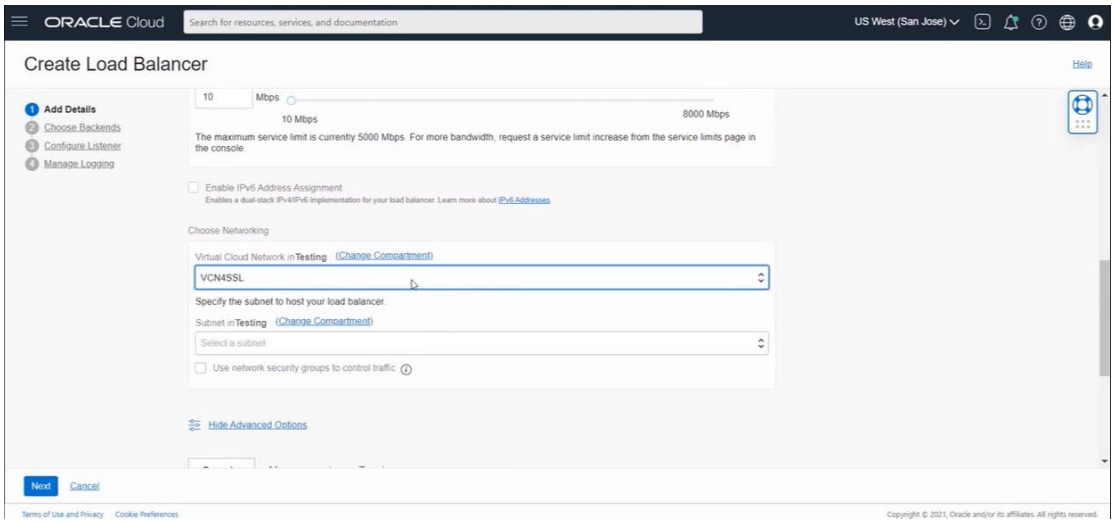
Ephemeral IP: An ephemeral IP address is dynamically assigned to the load balancer and is temporary in nature. It's typically used for short-term deployments or testing purposes. Ephemeral IPs are released when the associated load balancer is terminated, and they cannot be reserved for long-term use.

Reserved IP: A reserved IP address is a static, persistent IP address that you can reserve and assign to your load balancer. Unlike ephemeral IPs, reserved IPs remain associated with the load balancer even if it's terminated and can be reused across different deployments. Reserved IPs are suitable for production environments or scenarios where consistent IP addressing is required for routing traffic.

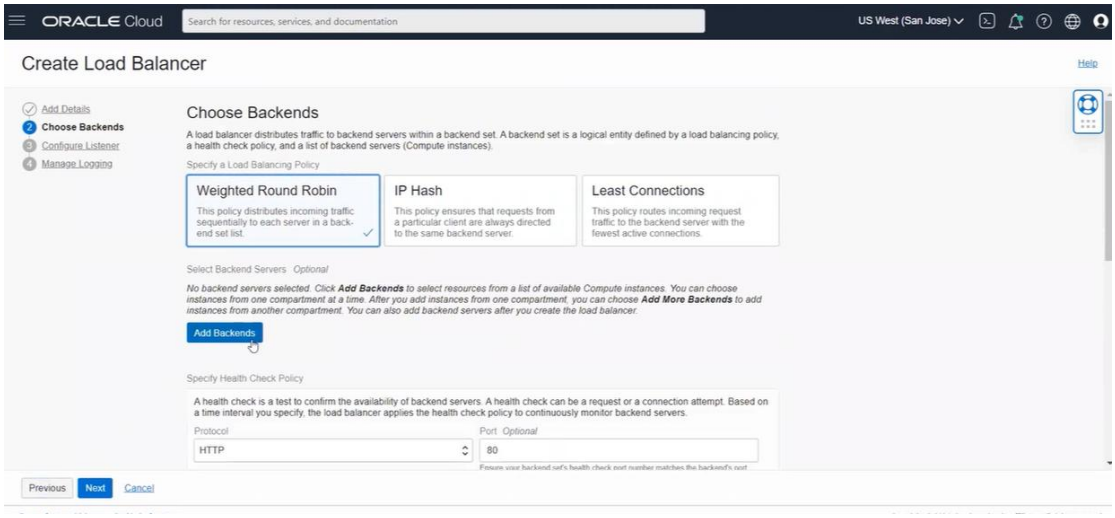
STEP 5 :- Keep in mind While creating chooser **Flexible Shape** cause web appliaction firewall policy will only work with Flexible shape.



STEP 6 :- Choose the VCN and Subnet in which we want to create the Load balancer



STEP 7 :- Click on next and choose from Weighted Round Robin, IP Hash, Least Connections



Weighted Round Robin:

This algorithm assigns a weight to each server based on its capacity or performance.

Traffic is distributed to servers in a cyclic manner according to their weights.

Servers with higher weights receive more traffic compared to those with lower weights.

Useful when you want to prioritize certain servers over others based on their capabilities.

IP Hash:

In this algorithm, the source IP address of the client is used to determine which server will handle the request.

The hash function generates a unique identifier from the client's IP address, and this identifier is used to select the server.

Ensures that requests from the same client are always routed to the same server.

Useful for session persistence or when maintaining state information between the client and server.

Least Connection:

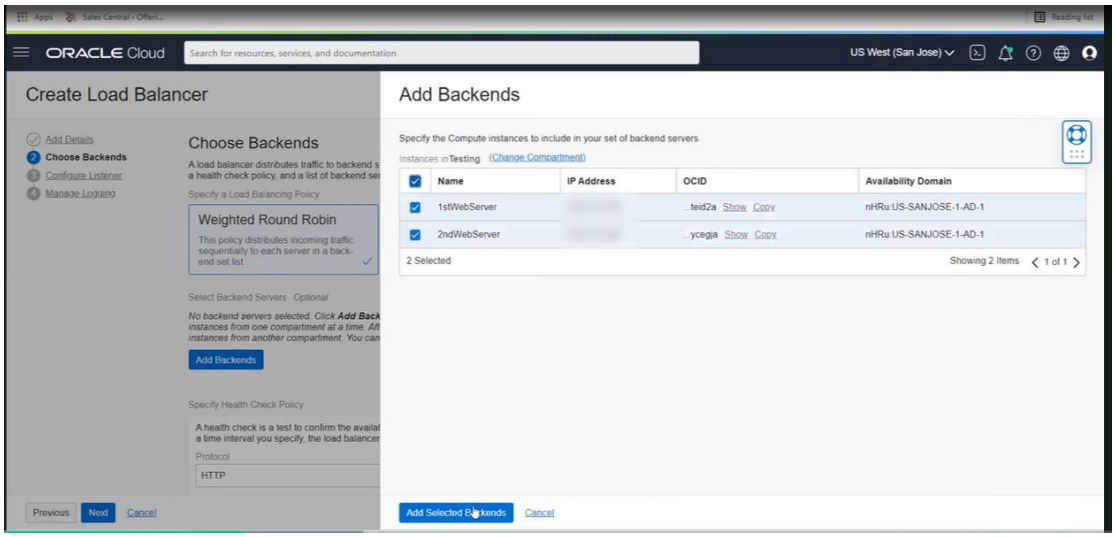
This algorithm directs traffic to the server with the fewest active connections at the time of the request.

It dynamically adjusts the load distribution based on the current workload of each server.

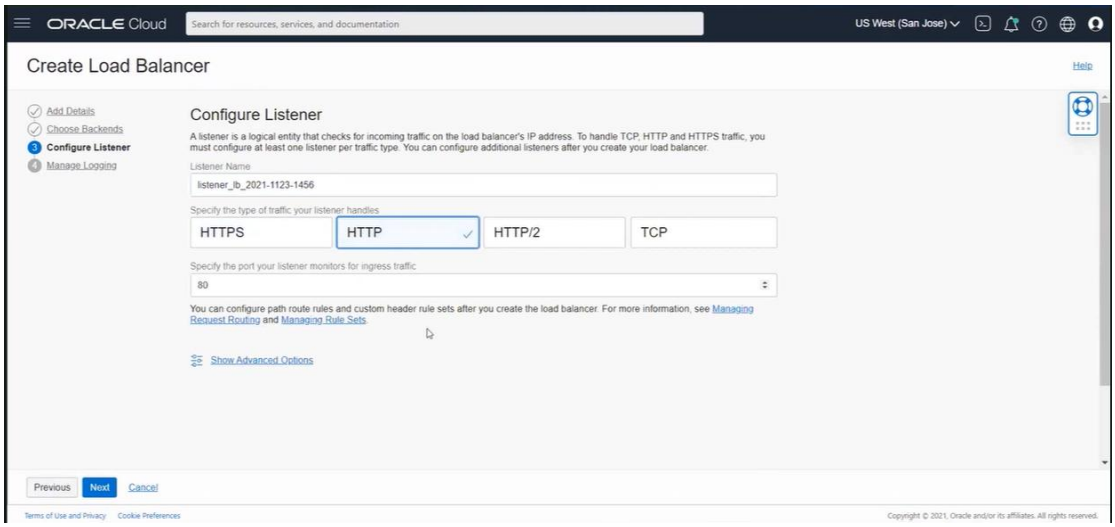
New connections are sent to the server with the least number of active connections.

Helpful in evenly distributing the load among backend servers and preventing overloading of any single server.

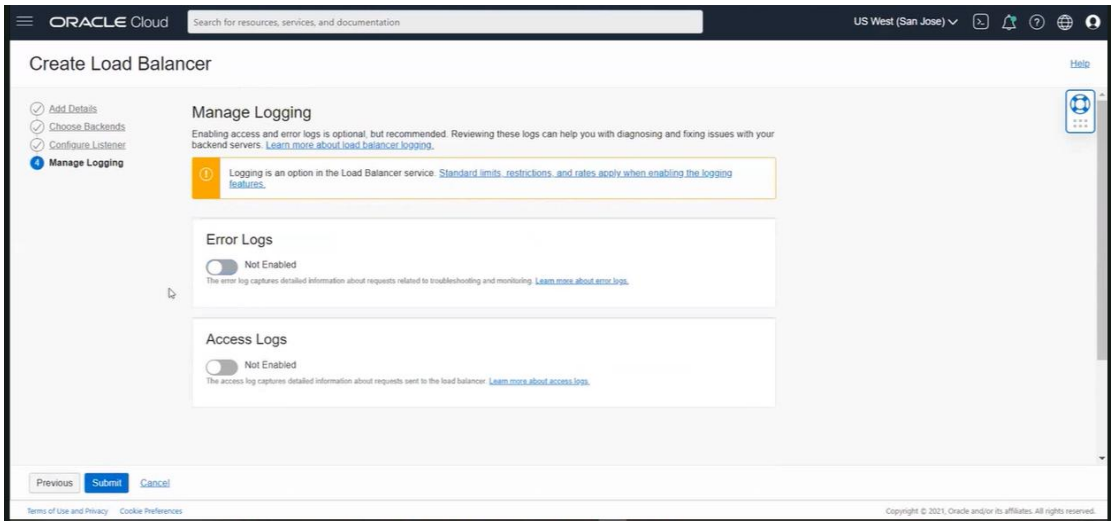
STEP 8 :- Add Backend Servers



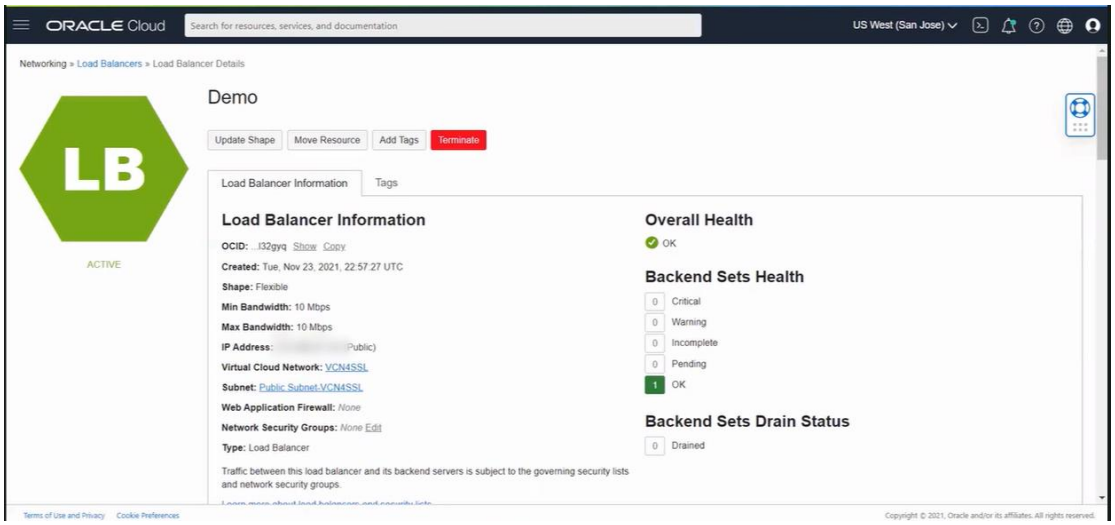
STEP 9 :- Click on next and choose on which protocol you want to configure Listener



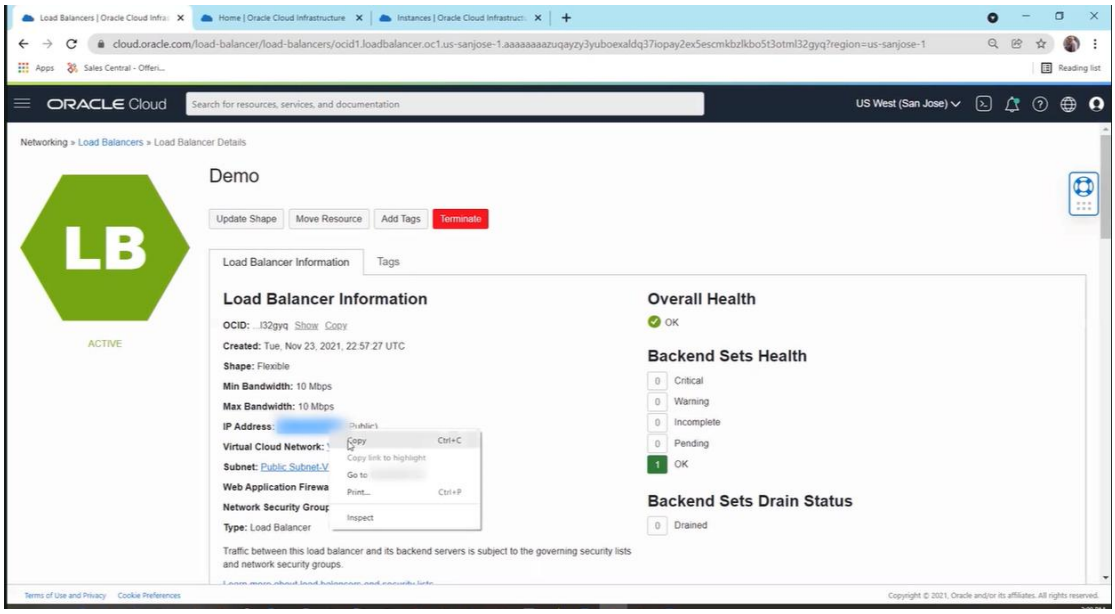
STEP 10 :- Click next ad Choose if you want enable or disable log and then click on submit



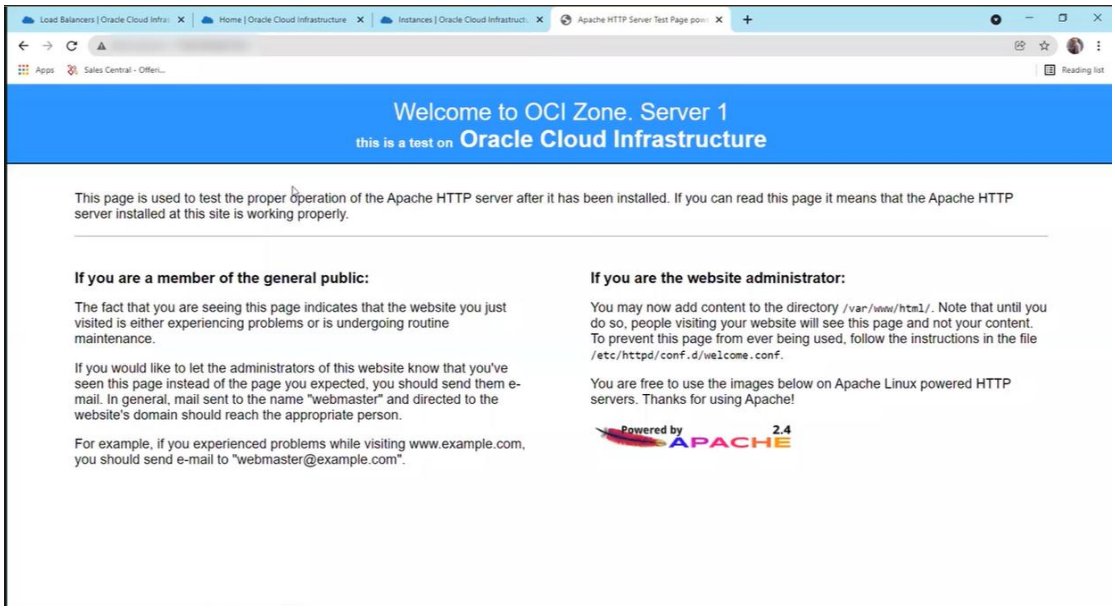
STEP 11 :- Successfully Created Load Balancer



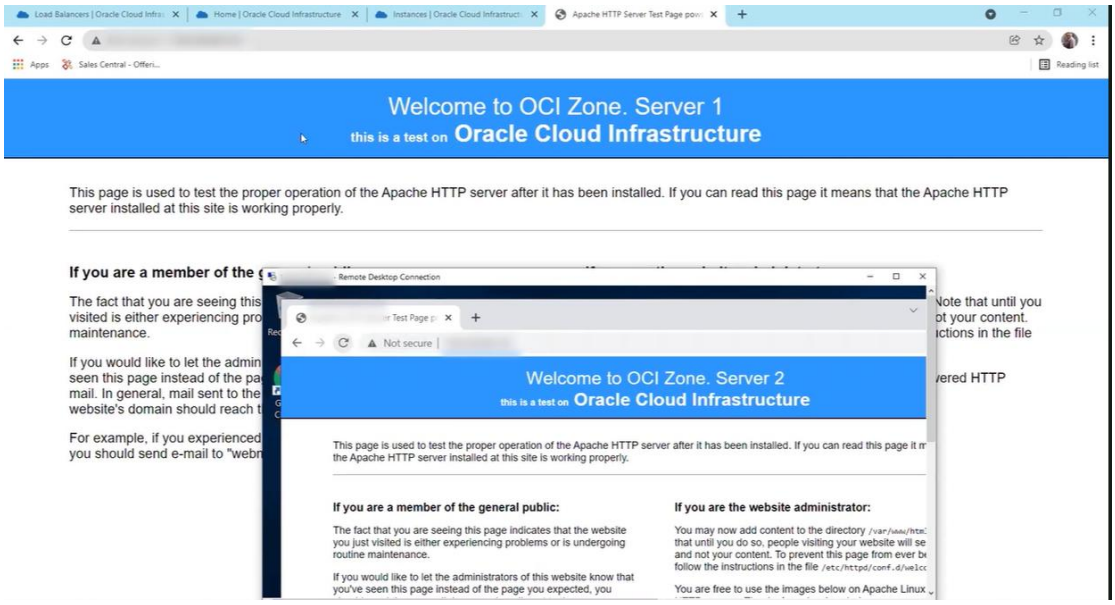
STEP 12 :- To check the working copy the IP address of the Load balancer



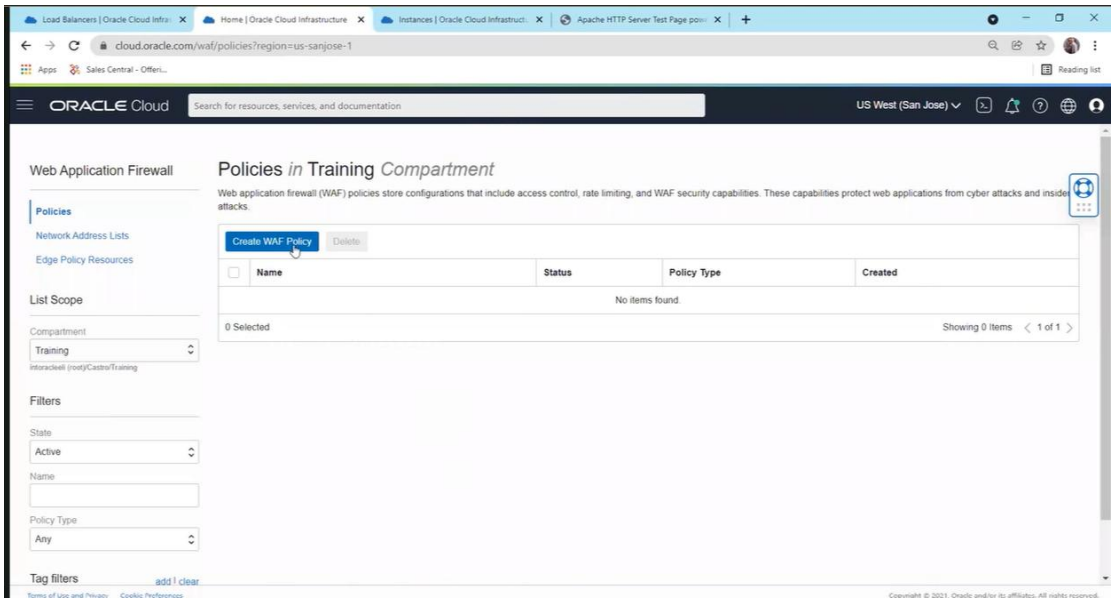
STEP 13 :- Paste it in the browser from US



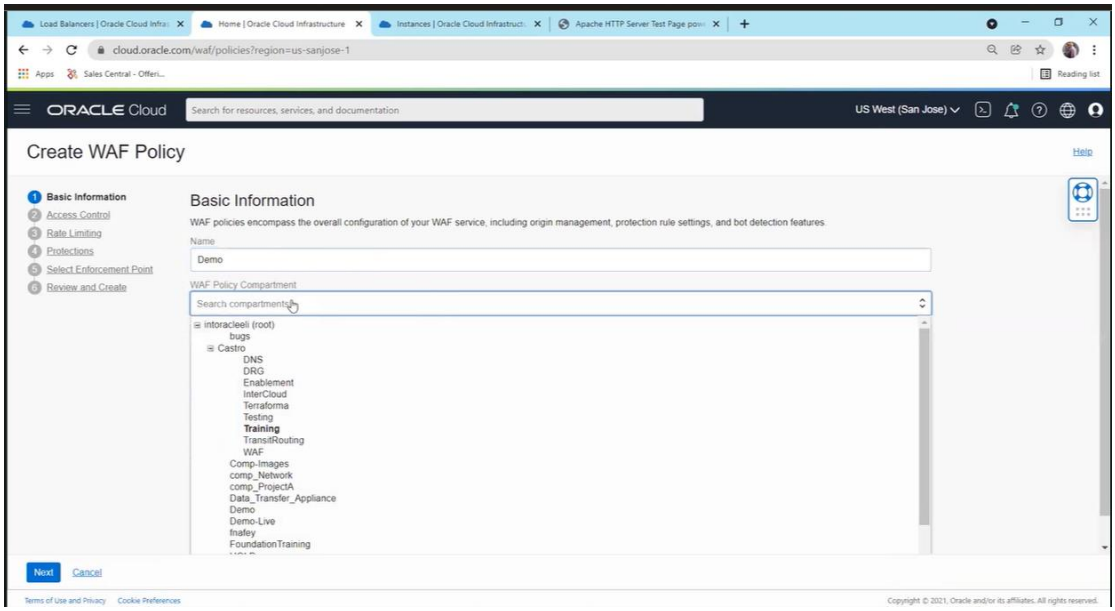
STEP 14 :- Paste it in the browser from German



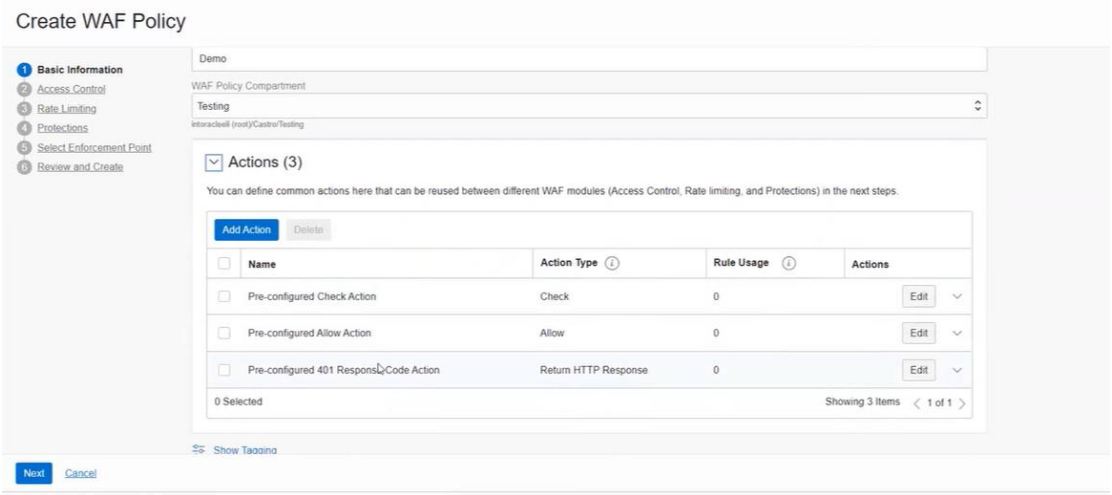
STEP 15 :- Now Creating a Web-Application Firewall policy



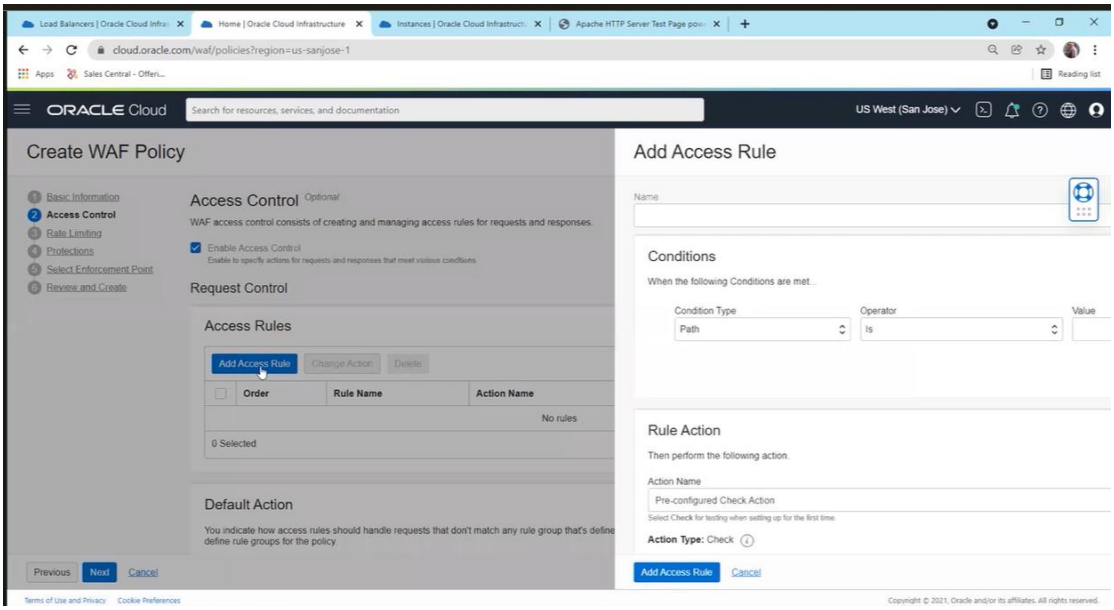
STEP 16 :- First the Basic Information Section Fill the name according to the requirement and same goes with the compartment



STEP 17 :- This are the Action we have Check, Allow and Detect, write now we don't have any rule
So we will just click on the Next



STEP 18 :- Will Click on the Access Control



Access Control:

Access control refers to setting rules and policies to control who can access your web application. This involves defining whitelist and blacklist rules to allow or block specific IP addresses or ranges from accessing your application. Access control helps in preventing unauthorized access and protecting your web application from malicious users or bots.

Rate Limiting:

Rate limiting involves setting limits on the number of requests that can be sent to your web application within a certain period of time. It helps in mitigating potential DDoS (Distributed Denial of Service) attacks by limiting the rate at which requests are processed. Rate limiting rules can be configured based on various parameters such as IP address, URL path, or HTTP method, allowing you to control the rate of incoming traffic and ensure the availability of your application.

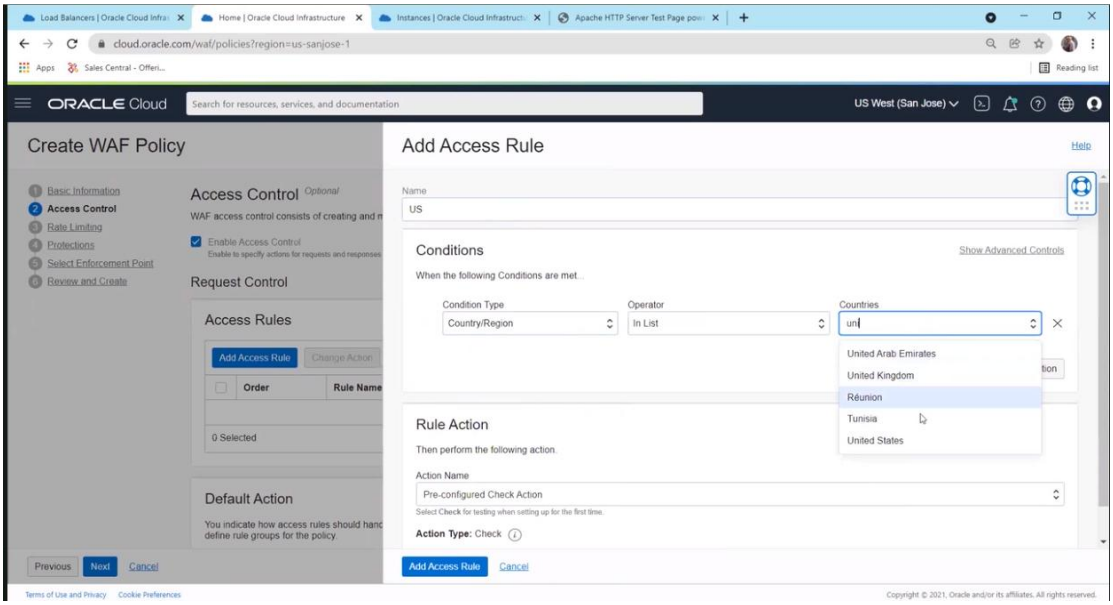
Protection:

Protection mechanisms in WAF involve implementing rules and policies to protect your web application from common security threats such as SQL injection, cross-site scripting (XSS), and other OWASP (Open Web Application Security Project) top 10 vulnerabilities. These protection rules are designed to inspect incoming requests and block or sanitize malicious payloads before they reach your application servers. By enforcing protection rules, WAF helps in safeguarding your web application and preventing potential security breaches.

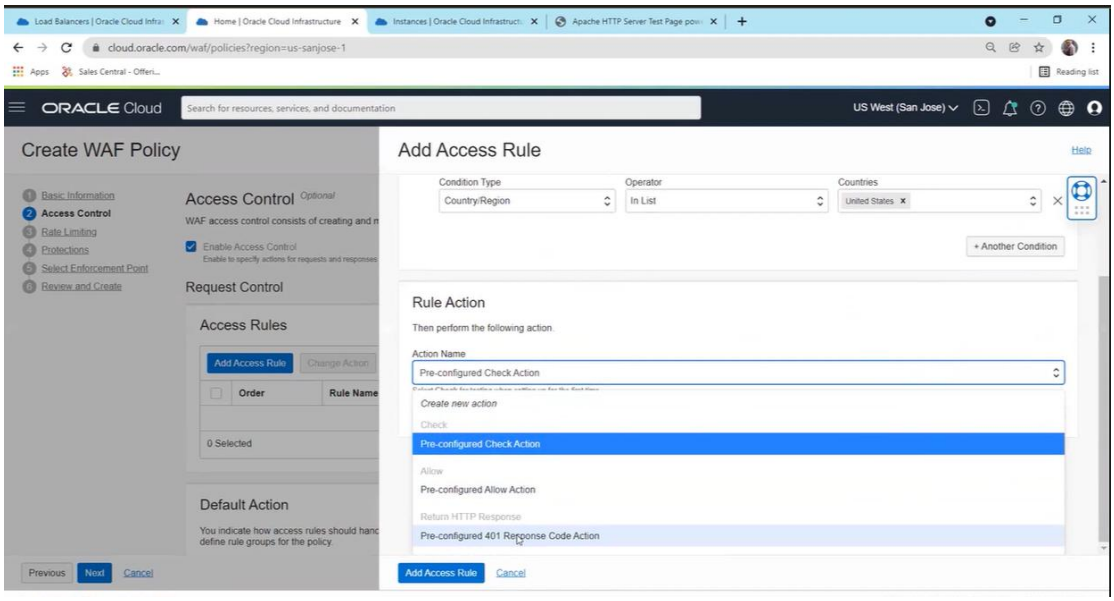
Enforcement Point:

The enforcement point is the location within your network where the WAF is deployed to intercept and inspect incoming traffic destined for your web application. In OCI, the enforcement point typically resides within the network path between the client and your application servers. It acts as a gateway through which all incoming requests must pass before reaching your application, allowing the WAF to inspect, filter, and enforce security policies in real-time.

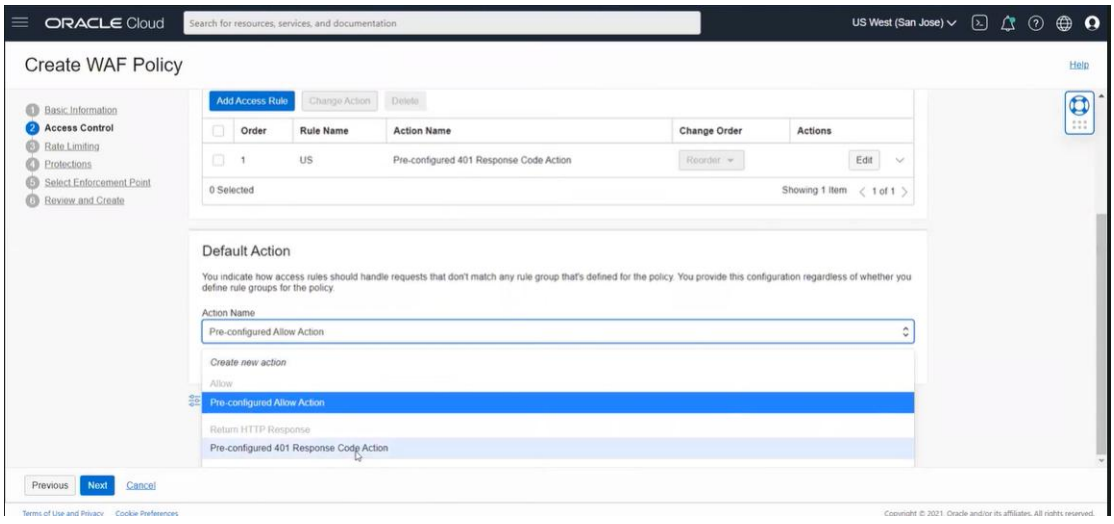
STEP 19 :- give a Name and choose a Condition



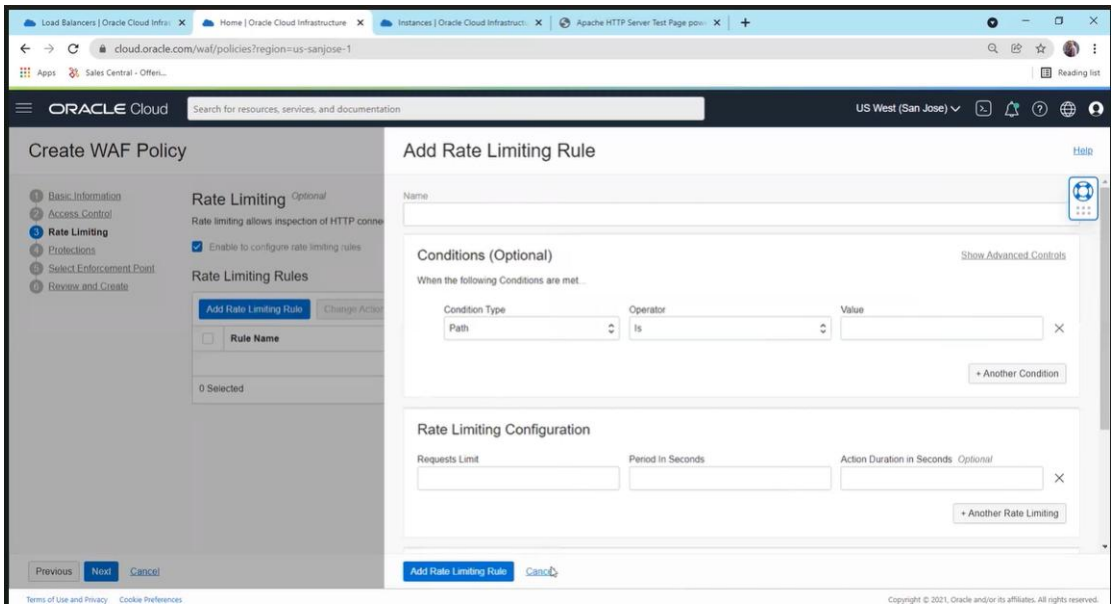
STEP 20 :- Choose the Rule Action



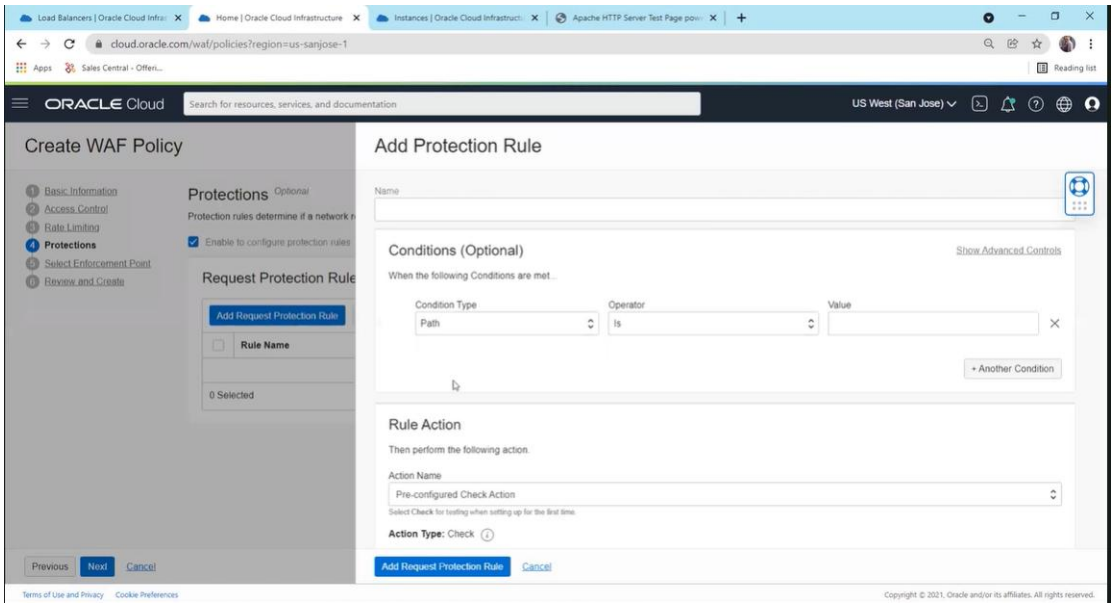
STEP 21 :- After adding the rule we have to choose the Default rule as well



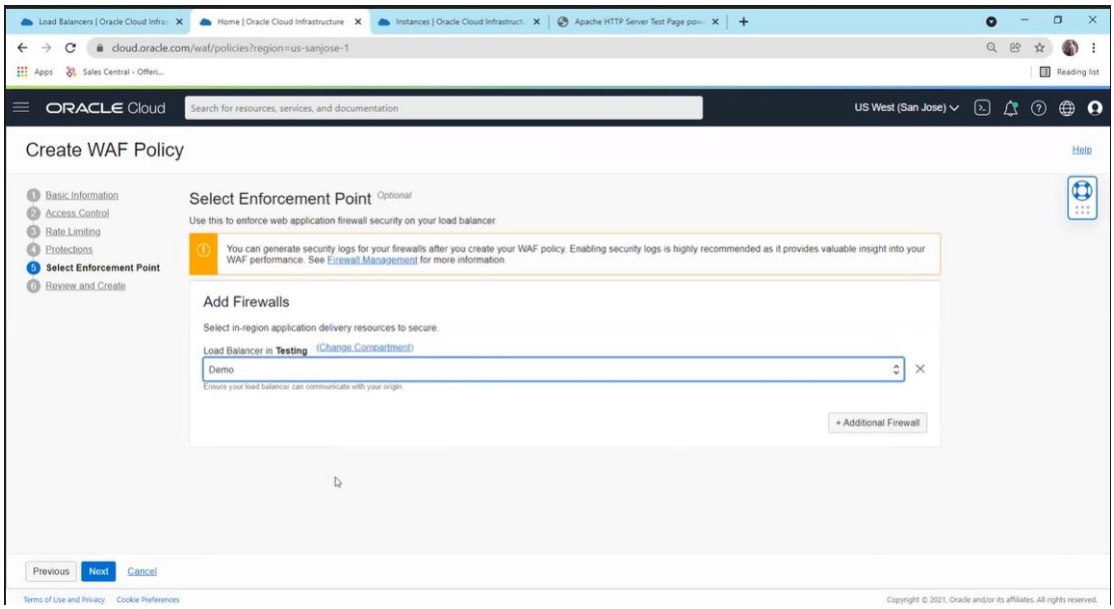
STEP 22 :- We can add the Rate Limiting Rule



STEP 23 :- We can add the protection Rule as Well



STEP 24 :- On the Enforcement Point we can select our Load Balancer



STEP 25 :- Review and create the Web Application Firewall