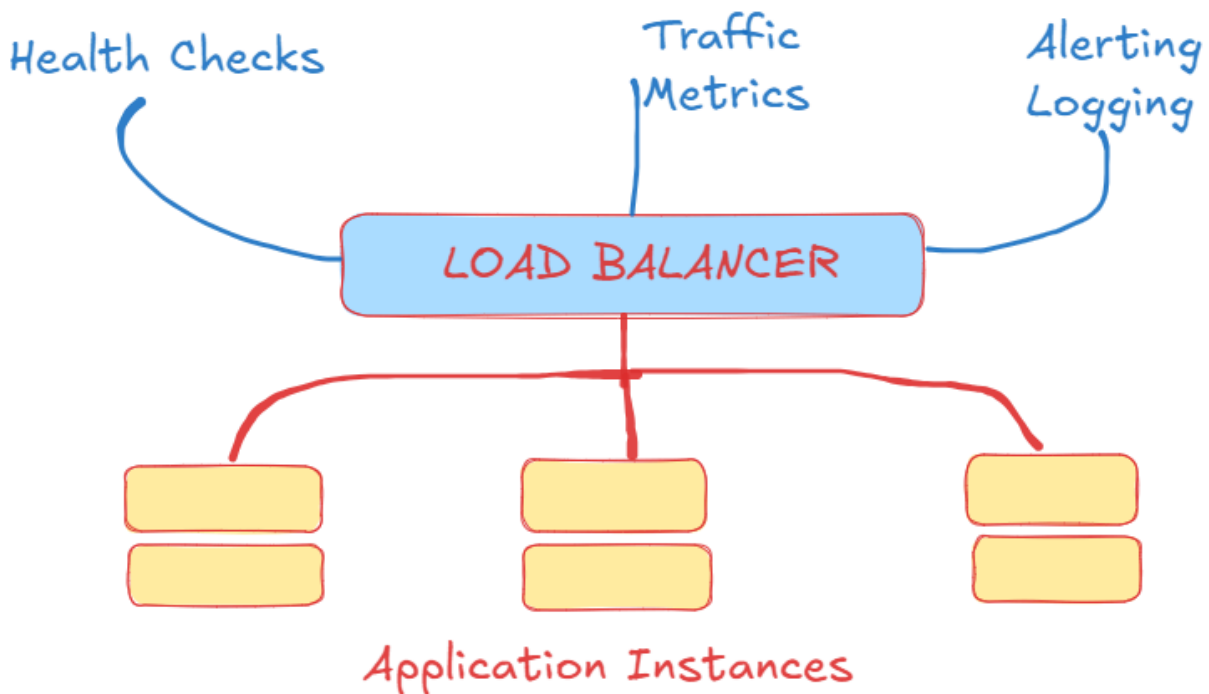


AWS ELB

Application Monitoring Using Load Balancer



Load balancers playing a very important role while providing high availability, performance and fault tolerance in DevOps.

Health Check:

- Monitor the health of backend servers

- what to check? instances are enough healthy to handle the traffic

- How to check?

 - HTTP Status Code (200 OK)

 - Response Time

 - Timeout or connection failure

Benefit: It will remove the traffic automatically if the server is unhealthy.

Traffic Metric:

- real time traffic distribution

- check request per seconds

- response time

- error rates (4xx, 5xx)

- active connections per instance

Tools:

AWS Cloud Watch

Grafana

Logging:

generate logs and access to monitor.

from log you can check:

Source IP

Requested Path

response status

ELK (Elastic Logstash and Kibana to do logging)

Alert:

is there any high error rates then generate Alert messages

Cloud Watch Alert

Let's Create 2 servers (2 instances) manage the request between them using load balancer.

Step 1:

create aws instance with AMI ubuntu

select default vpn but in subnet choose 1 zone, for another instance choose another zone.

security group allow HTTP port

create instance.

Connect to instance: `sudo apt update`

`sudo apt install nginx`

`sudo systemctl enable nginx`

`sudo systemctl start nginx`

check the serving index.html page location: `cat /etc/nginx/sites-available/default | grep root`

showing `/var/www/html` (location)

edit the file in both instances: `sudo nano`

`/var/www/html/index.nginx-debian.html`

change message like welcome from server 1

another instance Welcome from server2 and then restart server

`sudo systemctl restart nginx`

Step 2: Create Target Group to group Instances:

Create target group --> instances --> name target group (tg-sample) --> protocol http and port 80

ipv4 --> select default vpc --> protocol version http1 --> health check http and path

click on next --> select the instance which you want to add in group
port will be 80 and click on include as pending below --> click on create target group

Step 3: Create Load Balancer:

click on create Load balancer
select application load balancer --> click on create --> name (my-alb)--> internet-facing
select Ipv4 select default vpc in network --> select availability zones --> select security
group
--> listeners and routing (select target group)

Once the load balancer is created, copy the DNS name and then check responses coming
from different
instances.

After completing this clean resources.
delete them