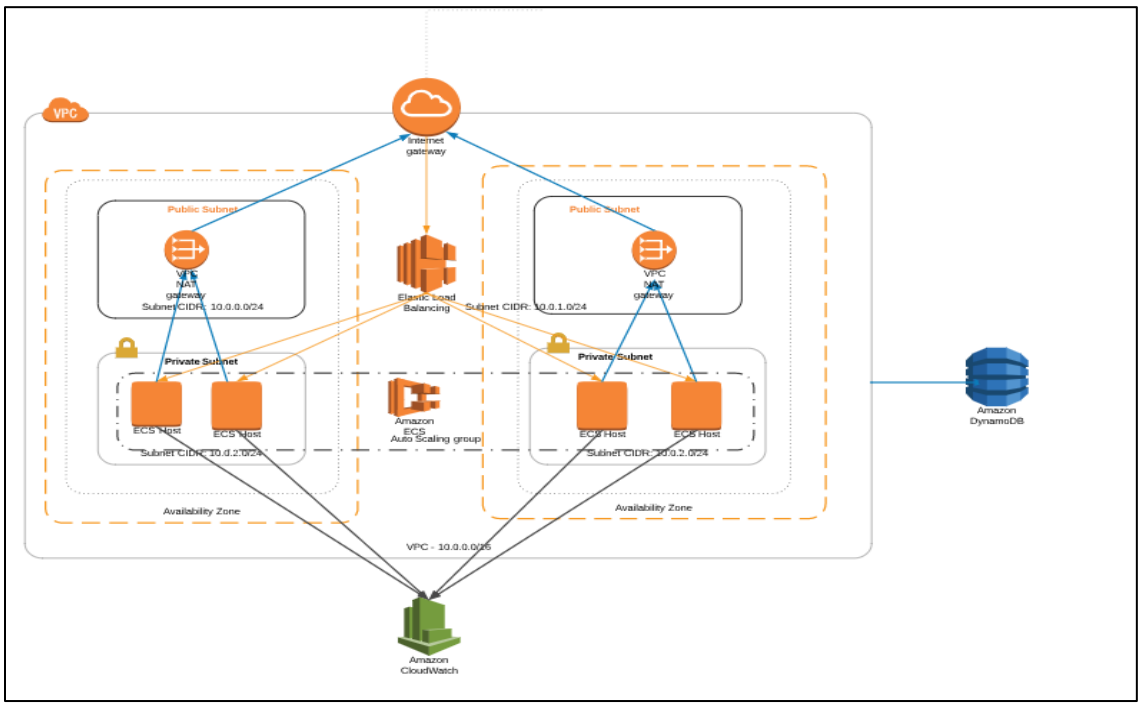
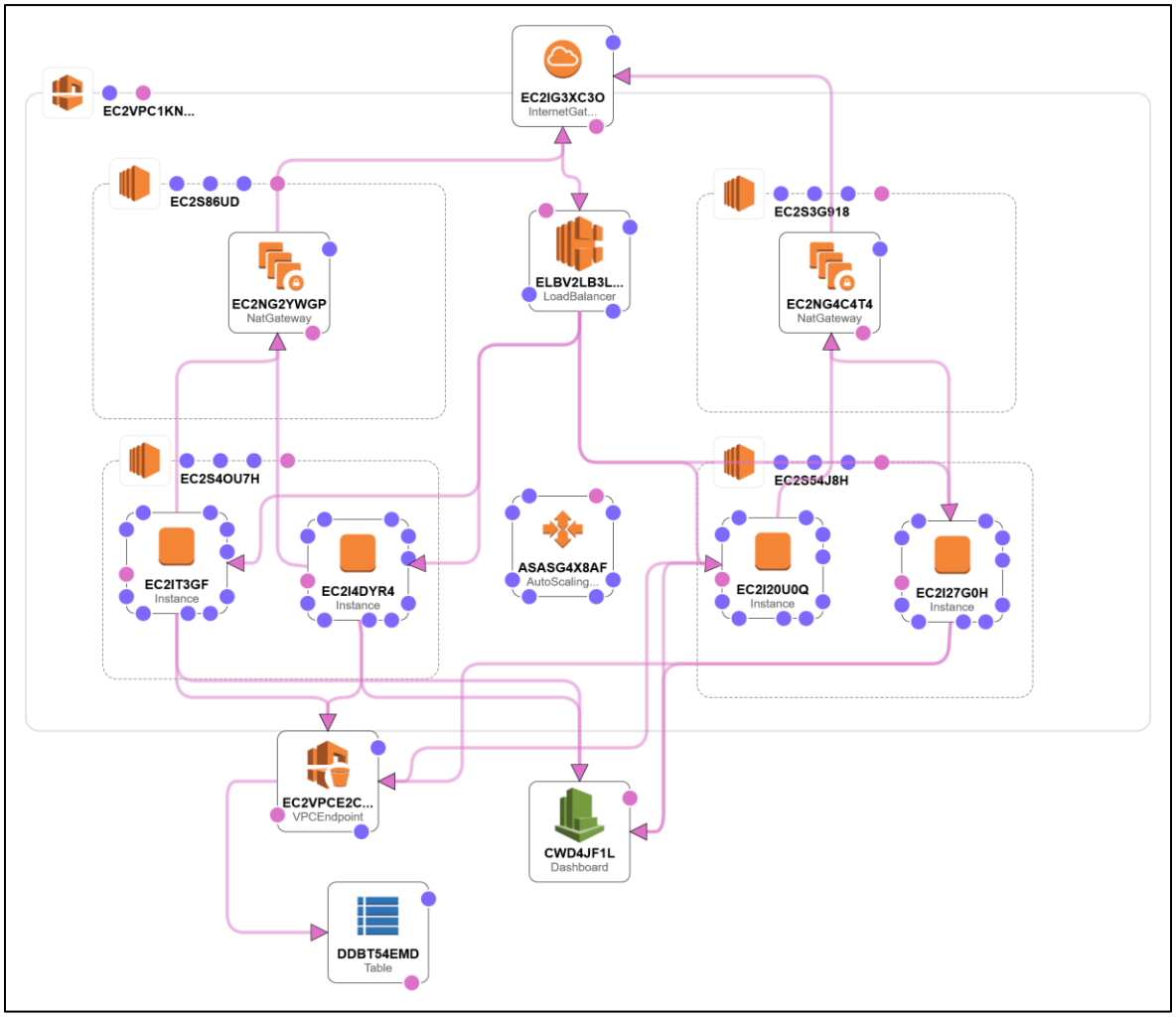
1.

Cloud formation design

Below diagram shows, the cloud formation design implemented using AWS cloud formation



Below shows the developed deployment diagram using AWS cloud formation designer



b. Component use in cloud formation designing.

Microservices, also known as microservice architectures are independently deployable and scalable. From a microservice, it can highly expect the resiliency, scalability and observability.

Resiliency: The ability of continues functioning of an application without occurring any failures is called resiliency. That mean even if the failures happen there should be fully functioning condition following failures. High availability and disaster recovery are main aspect in resiliency.

Scalability: Is the ability to change in volume according to the user requests. It can be scale-out or scale-in , in order to utilize the resources as per network traffic.

Observability : is the monitoring with time about the condition of microservice and their component performances and failures.

1. VPC (Virtual private cloud)

The created VPC of microservice is consist with two availability zones, two public and two private subnets.

- Availability zones.

Two availability zones have used in this cloud formation. If one availability zone goes down, then other availability zone will handle the requests. So application availability increase by using two availability zones. By using two availability zones, user can access to the application without any failures if one zone is down. Load balancer can direct the traffic to the available zones only without sending them to the failure zones. So this help to increase the resiliency also.

- Public and Private Subnet

There are two public subnets that routs through internet gateway and two private subnets that routs through Nat Gateways.

2. Internet gate way

Internet gate way allows the communication between instances in the VPC and the internet .Internet gateway provides target to VPC rout tables for internet routable traffic and serves network address translation(NAT) to instances.

3.NAT gateways

NAT gateways enables the instances in private subnets to the internets. Meanwhile NAT gateways can perform to prevent internet from initiating connection with instances.

4. Application load balancers

Application load balancer consist with three main component, Load balancers, Listeners and Target groups.

- Load Balancers

Application load balancer is a type of an elastic load balancer that act as a single point of contact for clients. This will distribute the incoming application traffic across EC2 instances in multiple availability zones.

- Listeners

Listener is a component of application load balancers .Listener will check the request from clients as per given protocol and ports and based on the define rules those request will forward to the targets groups.

- Target groups.

Target group is a component of application load balancer, this will routes request to registered target groups such as EC2 instances.

Application load balancer perform distributing the network traffic among the ECS host. Whenever instance scaling up load balancer will identify and direct the request to the new instance. Instance failure situation, load balancer will terminate the sending request to that particular instance. It helps to load SSL traffic and defend DDoS attacks Therefore, application load balancer perform to increase the availability and resiliency of application.

5. Auto-scaling group.

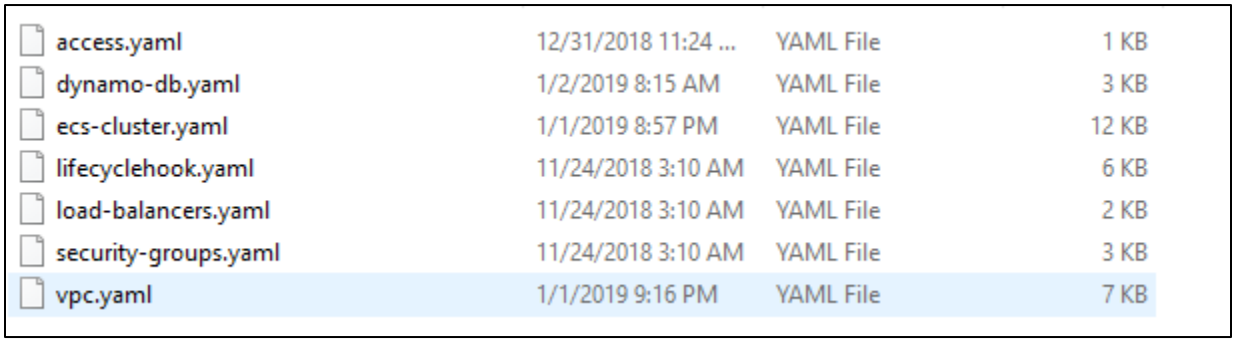
This will perform to create new EC2 instance (Scale up) and ECS task when in high network traffic. When traffic goes down again this will scale-In (reduce) the instances. So this will increase a high availability of the application

6. Cloud Watch

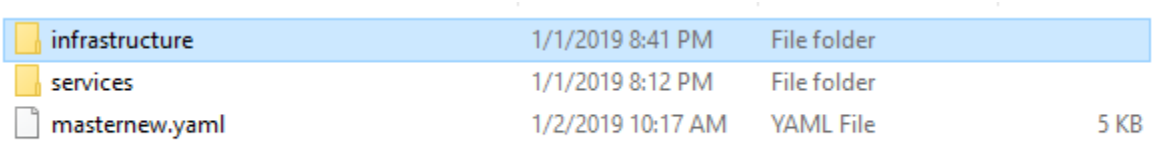
Cloud watch perform to monitor AWS cloud resources and the application running on it. This will collect track metrics, log files and set alarms. Cloud watch will provide a high observability of microservice resources and it can provide insights for running better application with high reaction and smoothly running.

2. Steps to run the AWS cloud formation script.

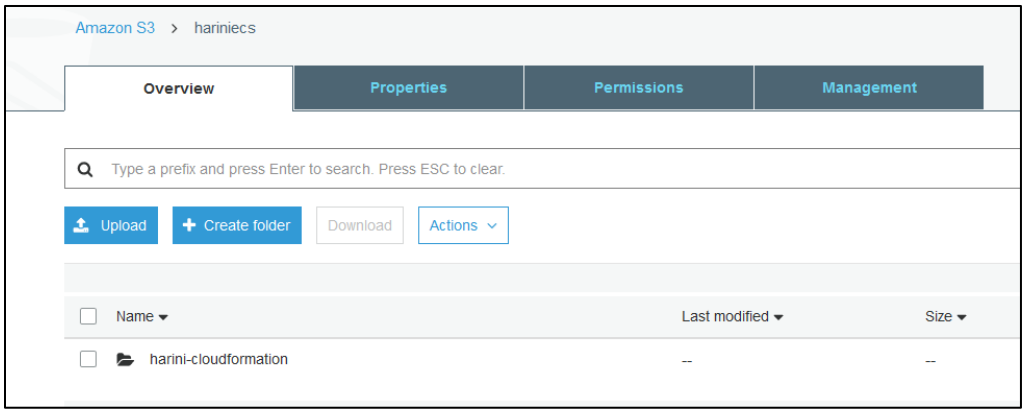
AWS cloud formation script include following yaml files for creating microservice infrastructure.



Masternew.yaml file will create the e above component accordingly and provides the required resources link for creating each stack.

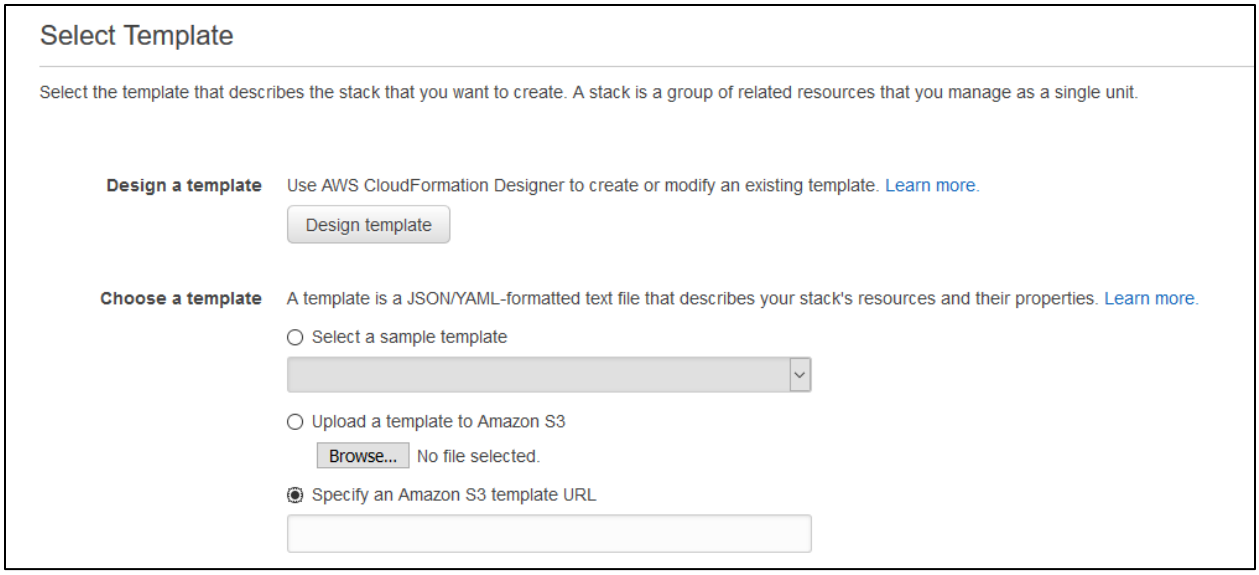


1. Upload the cloud formation file to S3 bucket

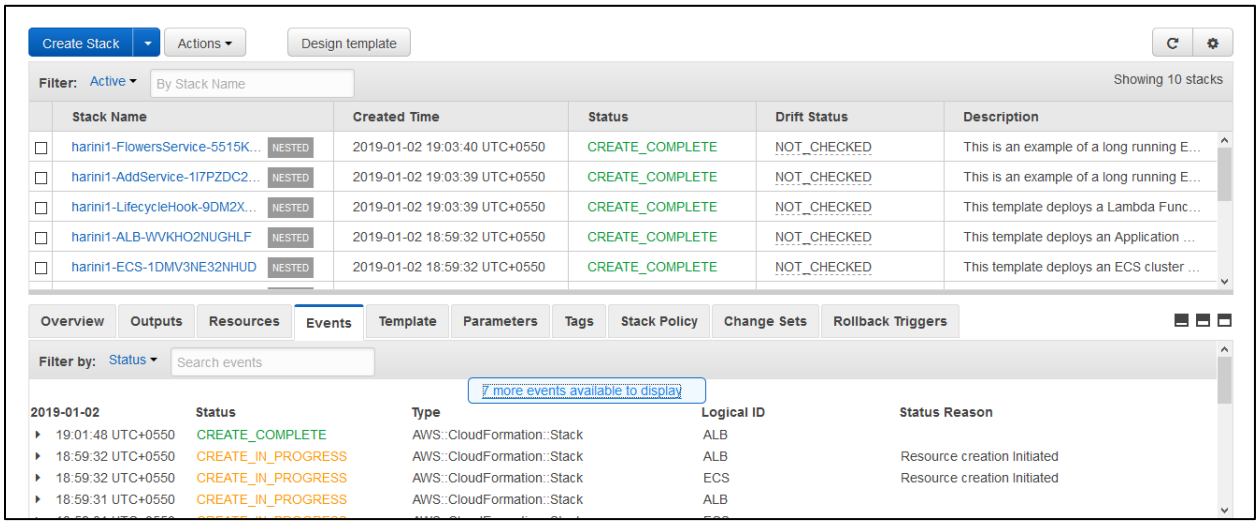


2. Creating a new stack

Under cloud formation, create new stack by providing Amazon S3 URL that contain **masternew.yaml**



Then create the stack with maternew.yaml and this will initiate to create other stacks as below,

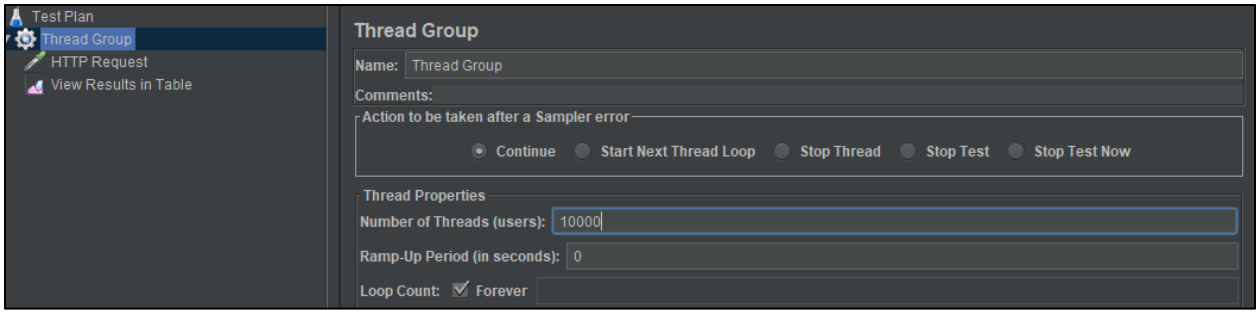


Above figure shows, the creating stacks using masternew.yaml

3. Apache Jmeter to test the environment.

To test the environment use the Apache Jmeter with following parameters,

- Tread Group

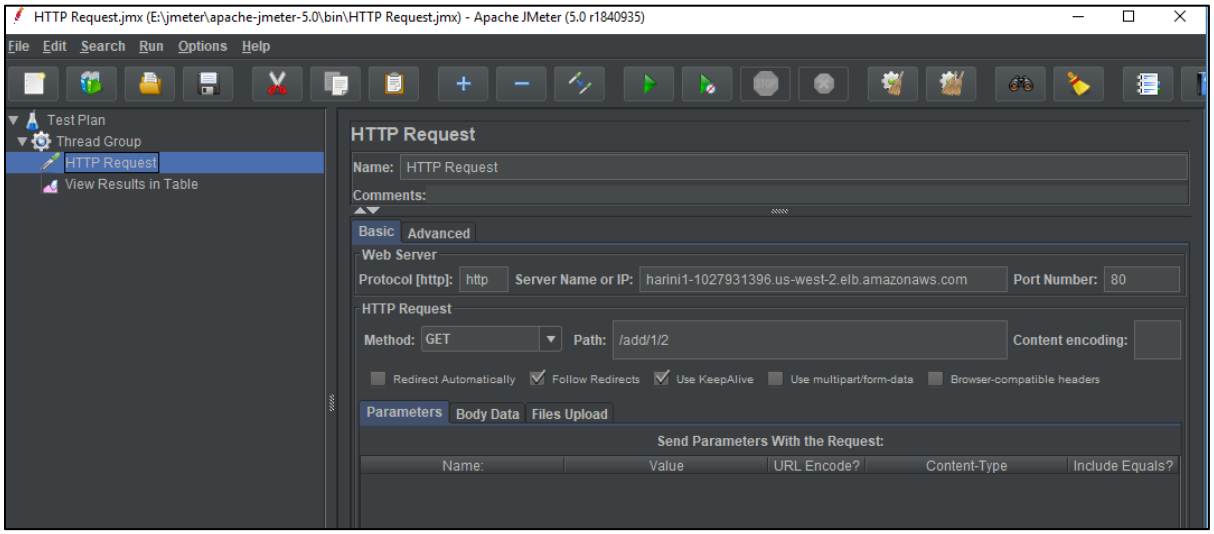


- HTTP Request.

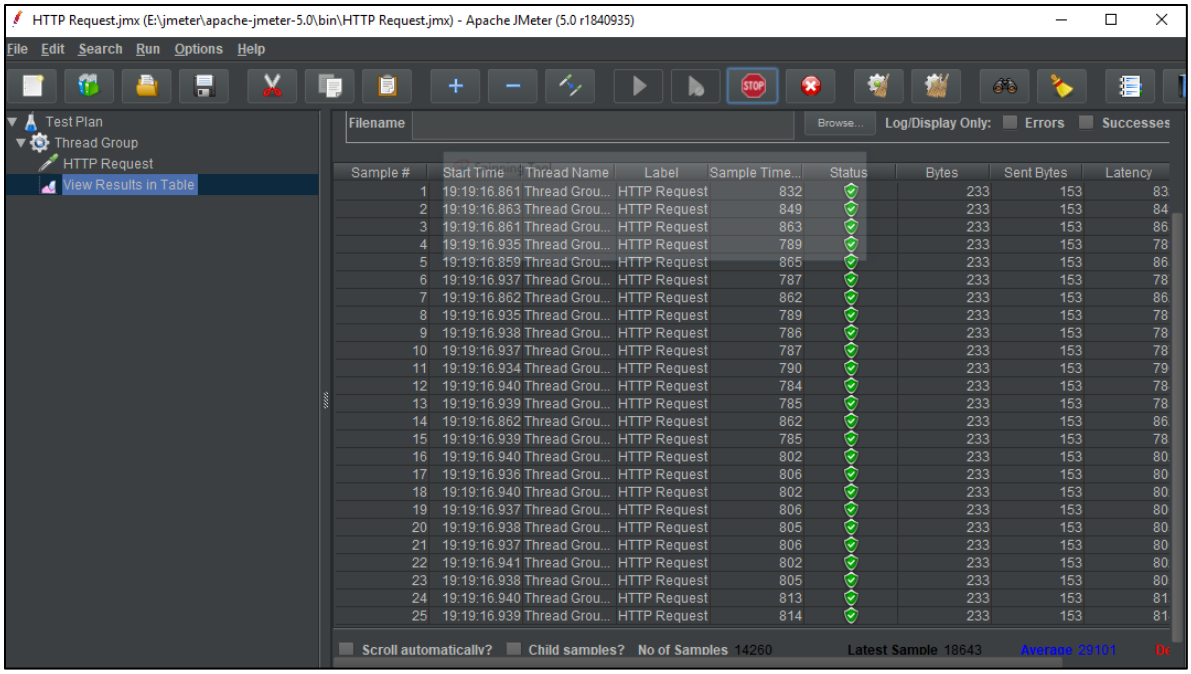
Protocol: http

Name or IP : harini1-1027931396.us-west-2.elb.amazonaws.com

Path : /add/1/2

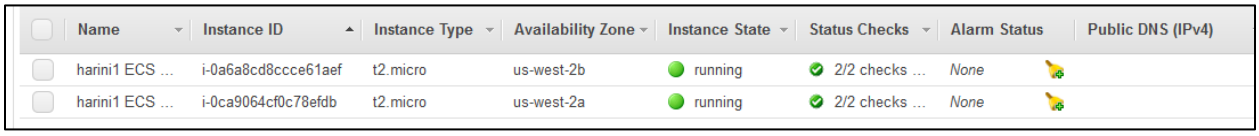


After setup the parameters s above run the Jmeter to send the request to the application. Below figure shows the sending request to the application.

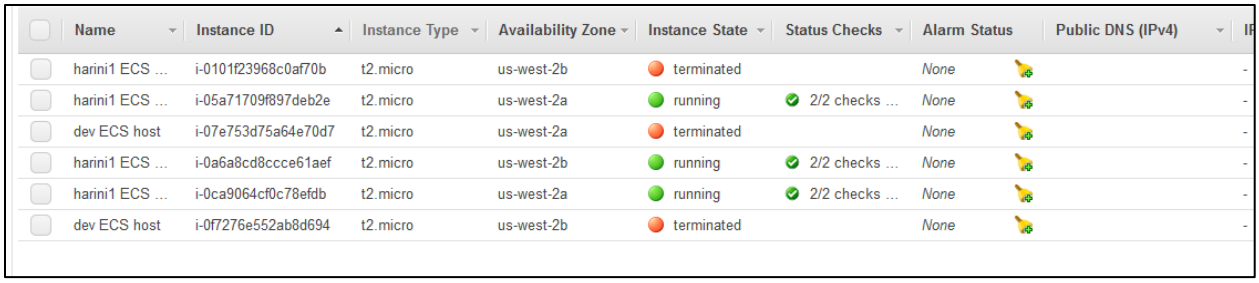


4. Interpreting the test results

a. Before sending the request to the application there are two instances running ,

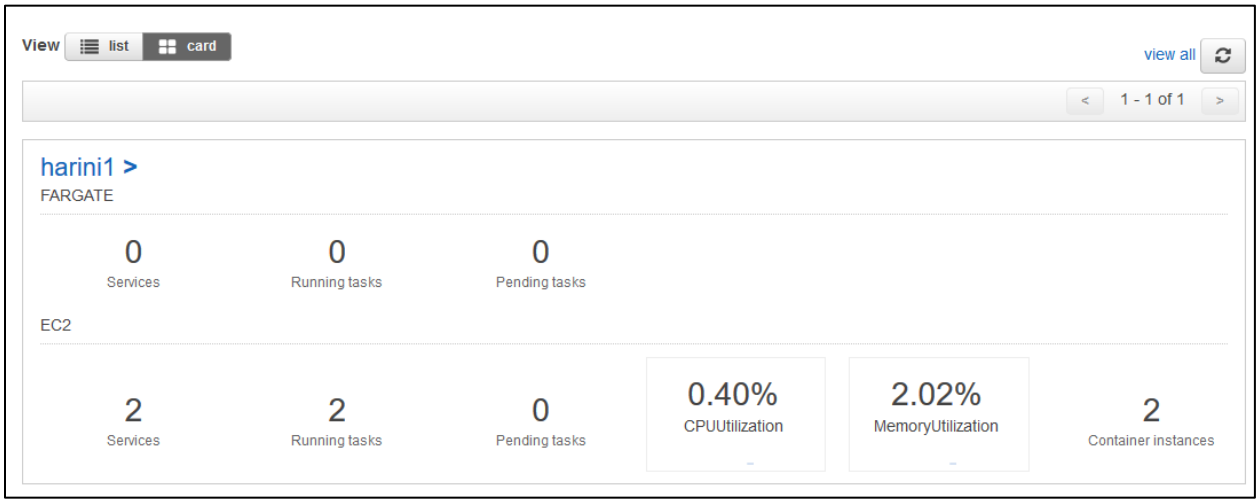


After sending, the requests there were three instance running.



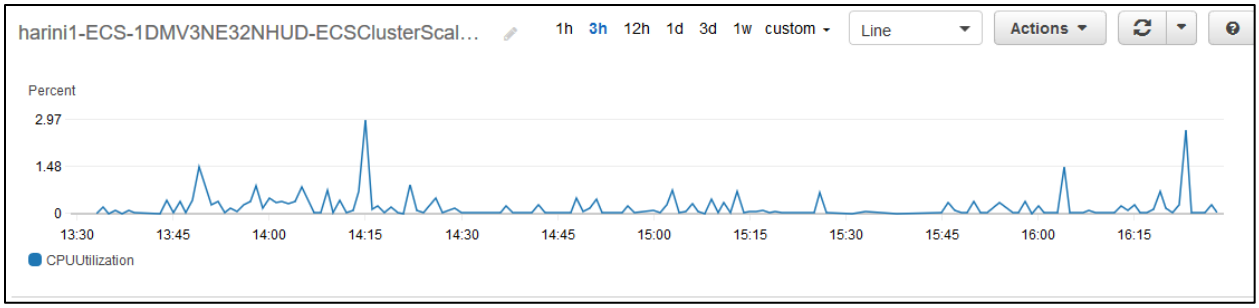
Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
harini1 ECS ...	i-0101f23968c0af70b	t2.micro	us-west-2b	terminated		None	-
harini1 ECS ...	i-05a71709f897deb2e	t2.micro	us-west-2a	running	2/2 checks ...	None	-
dev ECS host	i-07e753d75a64e70d7	t2.micro	us-west-2a	terminated		None	-
harini1 ECS ...	i-0a6a8cd8ccce61aef	t2.micro	us-west-2b	running	2/2 checks ...	None	-
harini1 ECS ...	i-0ca9064cf0c78efdb	t2.micro	us-west-2a	running	2/2 checks ...	None	-
dev ECS host	i-0f7276e552ab8d694	t2.micro	us-west-2b	terminated		None	-

From the jmeter test,it can be seen that when the traffic become high scale-up the instances by creating new instance. And ECS increase its task to perform in high traffic. After terminating the request from jmeter with time again instance become to scale-in by removing extra instance.

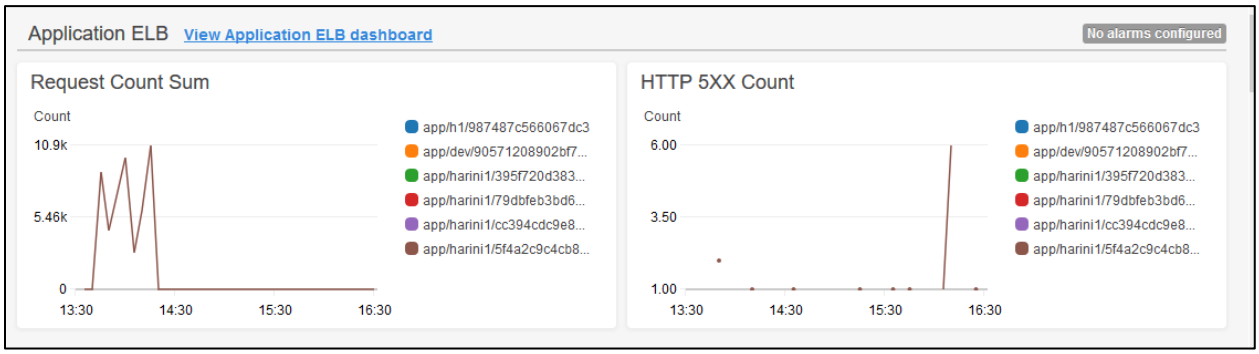


5. AWS Cloud watch graphs

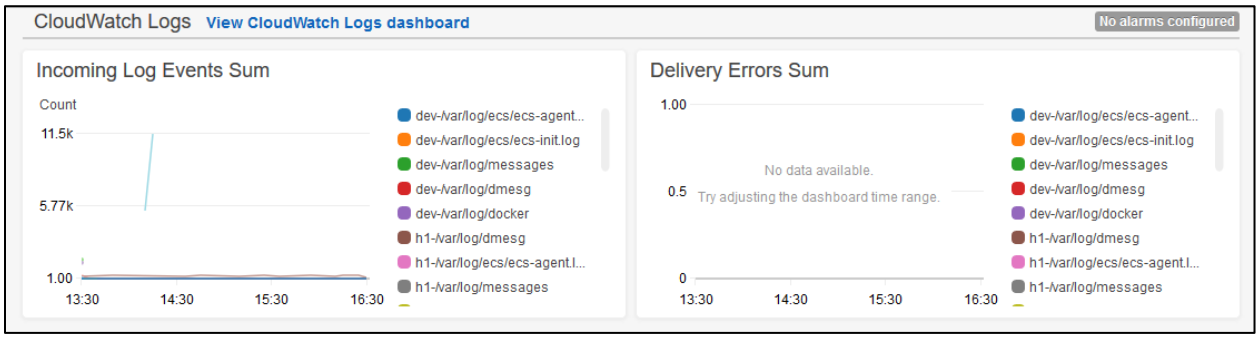
Below dashboards show the performance of each component during the application run time.



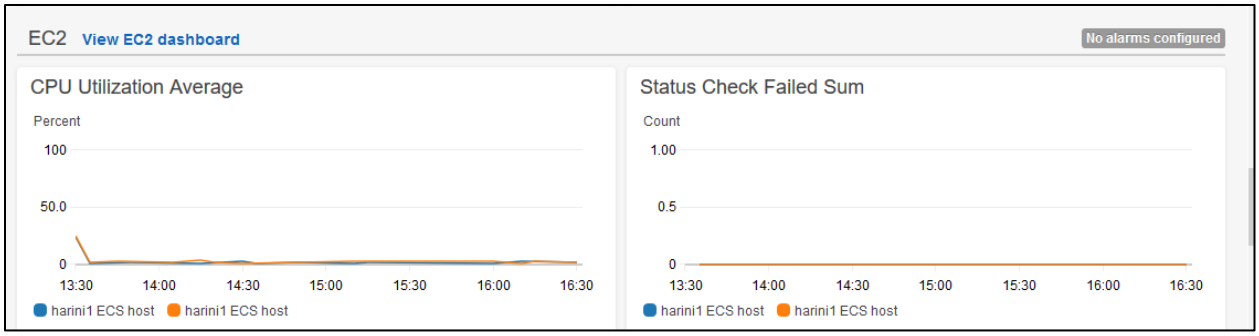
CPU utilization



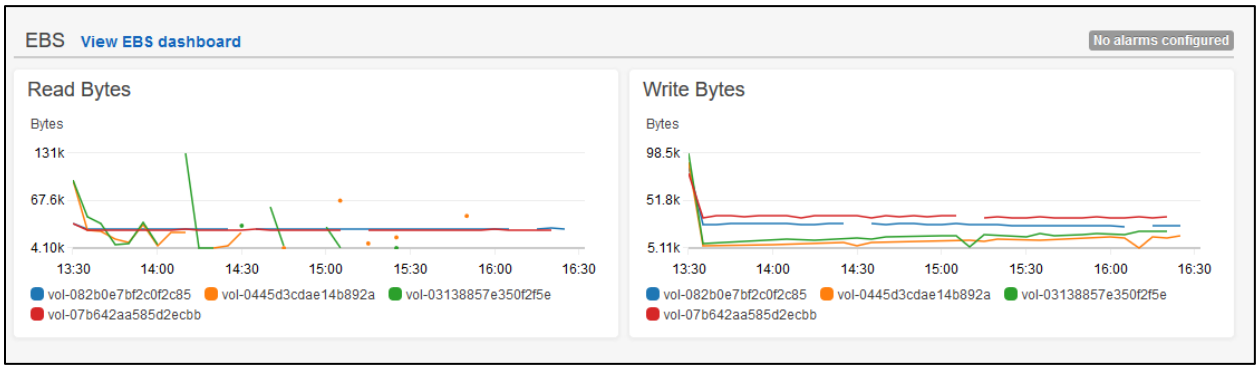
ELB dashboard



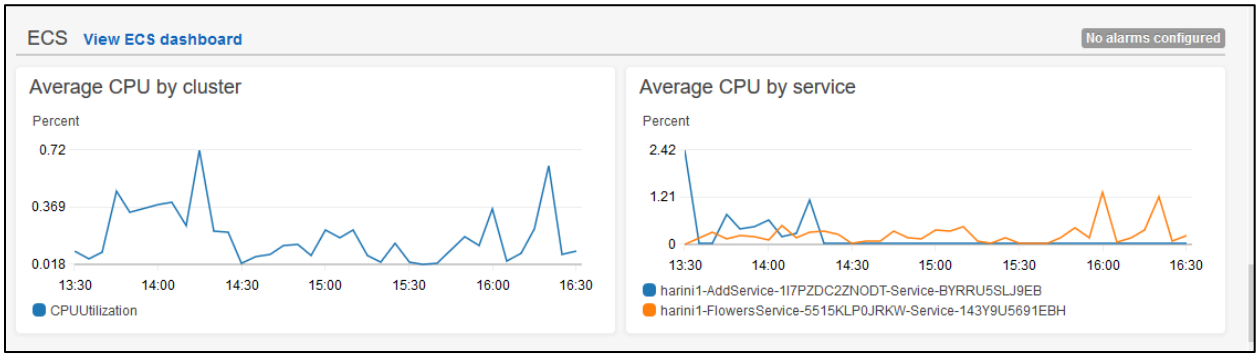
Cloud watch log dashboards



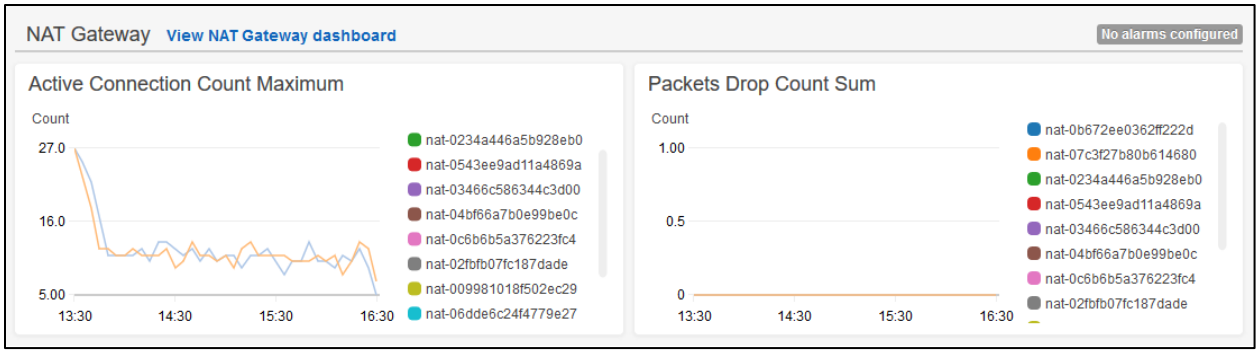
EC2 dashboard



EBS dashboard



ECS dashboard


NAT Gateway
View NAT Gateway dashboard
No alarms configured
Active Connection Count Maximum
Count
27.0
16.0
5.00
13:30
14:30
15:30
16:30
nat-0234a446a5b928eb0
nat-0543ee9ad11a4869a
nat-03466c586344c3d00
nat-04bf66a7b0e99be0c
nat-0c6b6b5a376223fc4
nat-02fbfb07fc187dade
nat-009981018f502ec29
nat-06dde6c24f4779e27
Packets Drop Count Sum
Count
1.00
0.5
0
13:30
14:30
15:30
16:30
nat-0b672ee0362ff222d
nat-07c3f27b80b614680
nat-0234a446a5b928eb0
nat-0543ee9ad11a4869a
nat-03466c586344c3d00
nat-04bf66a7b0e99be0c
nat-0c6b6b5a376223fc4
nat-02fbfb07fc187dade

NAT gateway