



AWS – CAPSTONE 5

Deploy Application using Kubernetes in AWS cluster (EC2)
using Cloud Formation

Rajeswari Subramanian | December,2022

Contents

Deploy Application using Kubernetes in AWS cluster (EC2) using Cloud Formation	1
1. Project Overview	4
2. Configure AWS CLI & kubectl	4
2.1. Install AWS CLI.....	4
2.2. Configure AWS Command Line using Security Credentials.....	4
2.3. Windows 10 - Install and configure kubectl.....	4
3. Architecture Diagram	5
AWS Components	7
4. Cluster Creation	8
4.1. Create Cluster.....	8
4.2. Create Node Group	13
4.3. Get Cluster, node groups, nodes.....	18
5. Auto Scaling Setup	21
5.1. Create IAM policy and role.....	22
5.2. Deploy the Cluster Autoscaler	24
5.3. View Cluster Autoscaler logs.....	27
6. Deploy Application	27
6.1. Start up the Redis Database.....	27
6.2. Set up and Expose the Guestbook Frontend	32
7. Auto Scaling.....	39
7.1. Scale the Pods	39
7.2. Scale the Nodes (EC2)	44
8. Cost Analysis	54
9. Lessons & Observations	55



1. Project Overview

Use Amazon EKS to deploy a highly available Kubernetes control plane. Configure 'kubectl', an open-source command line tool to interact with the Kubernetes infrastructure. Using AWS CloudFormation, launch a cluster of worker nodes on Amazon EC2, then launch a containerized guest book application onto the cluster.

2. Configure AWS CLI & kubectl

2.1. Install AWS CLI

- Download Binary: <https://awscli.amazonaws.com/AWSCLIV2.msi>
- Install the downloaded binary (standard windows install)

2.2. Configure AWS Command Line using Security Credentials

- Go to AWS Management Console --> Services --> IAM. select the IAM User
- Click on **Security credentials** tab
- Click on **Create access key**
- Copy Access ID and Secret access key
- Go to command line and provide the required details

```
aws configure
AWS Access Key ID [None]: xxxxxxxx (Replace your creds when prompted)
AWS Secret Access Key [None]:xxxxxxxxxxxx (Replace your creds when prompted)
Default region name [None]: us-east-1
Default output format [None]: json
```

2.3. Windows 10 - Install and configure kubectl

- Install kubectl on Windows 10

```
mkdir kubectlbinary
cd kubectlbinary
curl -o kubectl.exe https://amazon-eks.s3.us-west-2.amazonaws.com/1.16.8/2020-04-16/bin/windows/amd64/kubectl.exe
```

- Update the system **Path** environment variable

Edit environment variable X

C:\Users\agni1\Anaconda3\Library\bin
C:\Users\agni1\Anaconda3\Scripts
C:\Users\agni1\AppData\Local\Programs\Python\Python38\S...
C:\Users\agni1\AppData\Local\Programs\Python\Python38\
C:\Users\agni1\AppData\Local\Microsoft\WindowsApps
C:\Users\agni1\.dotnet\tools
C:\Users\agni1\AppData\Local\GitHubDesktop\bin
C:\Users\agni1\AppData\Local\atom\bin
C:\Program Files\Java\jdk-15.0.1
%USERPROFILE%\dotnet\tools
%IntelliJ IDEA Community Edition%
%USERPROFILE%\AppData\Local\Microsoft\WindowsApps
C:\Users\agni1\AppData\Roaming\npm
C:\Users\agni1\AppData\Local\Android\Sdk\platform-tools
C:\kafka_2.13-2.8.0\bin\windows
C:\apache-maven-3.8.1-bin\apache-maven-3.8.1\bin
C:\Program Files\Git\usr\bin
C:\Users\agni1\AppData\Local\Programs\Microsoft VS Code...
C:\Program Files\MongoDB\Tools\100\bin
C:\Program Files\MongoDB\Server\4.4\bin
C:\Users\agni1\.kube\config

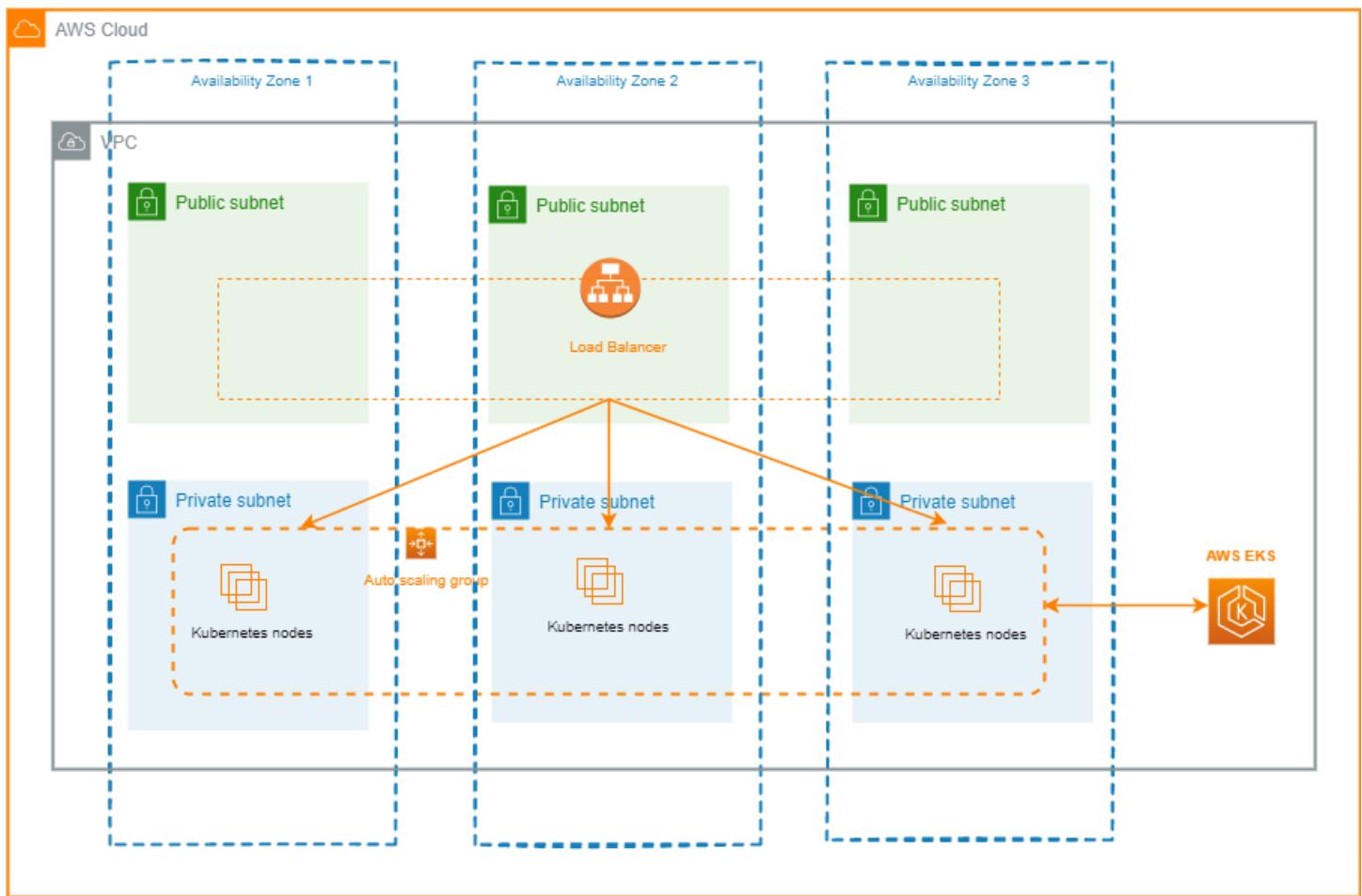
New
Edit
Browse...
Delete
Move Up
Move Down
Edit text...

OK Cancel

- Verify the kubectl client version

```
kubectl version --short --client  
kubectl version --client
```

3. Architecture Diagram



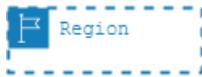
In this architecture, Amazon EKS cluster is created which provides the Kubernetes control plane. This EKS cluster spans across three Availability Zones. A virtual private cloud (VPC) configured with public and private subnets according to AWS best practices, to provide our own virtual network on AWS.

In the public subnets, load balancer is deployed to allow outbound internet access for ‘frontend’ pods running on EC2 nodes. EC2 nodes are present in private subnet to avoid direct access to internet

AWS Components



AWS Cloud



Region: It is a separate geographic area.



VPC: It is a virtual network that closely resembles a traditional network that we'd operate in our own data center



Availability Zone: Each Region has multiple, isolated locations known as Availability Zones. The code for Availability Zone is its Region code followed by a letter identifier.



Private Subnet: A subnet is a range of IP addresses in your VPC. Private subnet does not have a direct route to an internet gateway. Resources in a private subnet require a NAT device to access the public internet



Public Subnet: A subnet is a range of IP addresses in your VPC. Public subnet has a direct route to an internet gateway. Resources in a public subnet can access the public internet.



Amazon Elastic Kubernetes Service (Amazon EKS): It is a managed service that can be used to run Kubernetes on AWS without needing to install, operate, and maintain our own Kubernetes control plane or nodes. Kubernetes is an open-source system for automating the deployment, scaling, and management of containerized applications



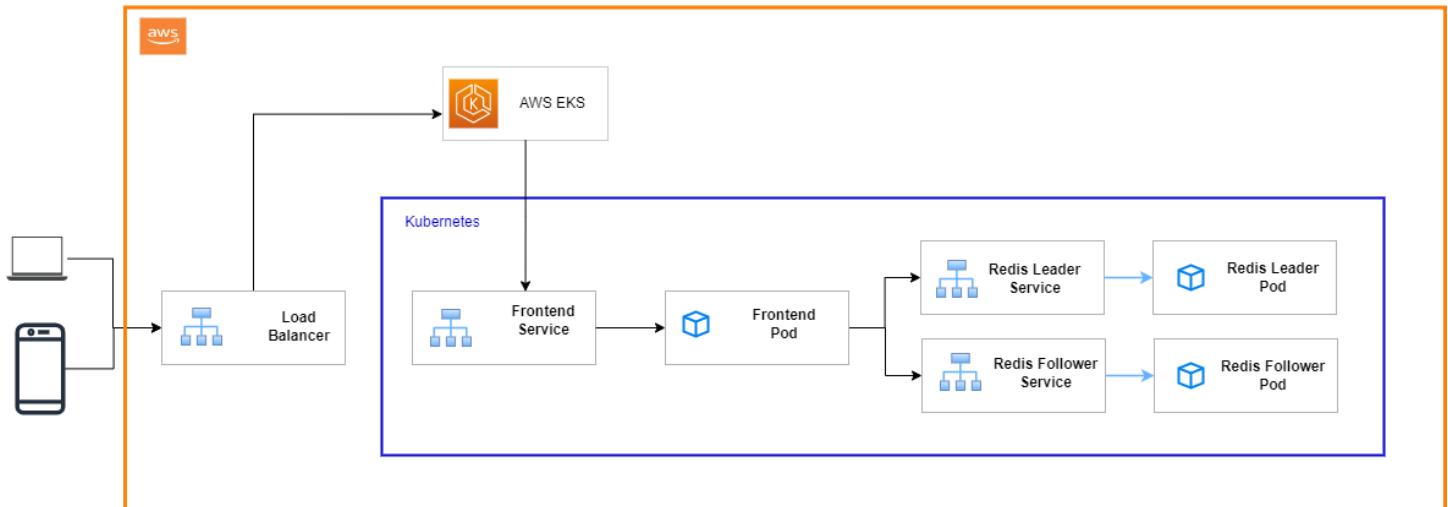
Autoscaling: It automatically scales up or down EC2 nodes in cluster , to meet changing demands. The cluster autoscaler uses AWS scaling groups which is used in this project



AWS Application Load Balancer (ALB): It is provisioned that load balances application traffic. ALBs can be used with pods that are deployed to nodes which are present in private subnets.



Amazon EKS managed node group: It can create, automatically update, or terminate nodes for our cluster with a single operation. Node updates and terminations automatically drain nodes to ensure that our applications stay available. Every managed node is provisioned as part of an Amazon EC2 Auto Scaling group that's managed for us by Amazon EKS. Every resource including the instances and Auto Scaling groups runs within our AWS account. Each node group runs across multiple Availability Zones that we define.



The guestbook app uses a PHP frontend. It is configured to communicate with either the Redis follower or leader Services, depending on whether the request is a read or a write. The frontend exposes a JSON interface, and serves a jQuery-Ajax-based UX.

Frontend is created using PHP, which connects to redis-leader & redis-follower to write and read data from redis database respectively. Pods and their corresponding services are running in AWS EKS cluster. Loadbalancer connects frontend pod to internet, through which we are able to access 'guestbook' application.

4. Cluster Creation

4.1. Create Cluster

Create cluster called 'capstone5' without node group on 'us-east-1' region which includes 3 availability zones (us-east1a, us-east1b, us-east1c)

```
eksctl create cluster --name capstone5 --region us-east-1 --zones=us-east-1a,us-east-1b,us-east-1c --without-nodegroup
```

--name → Name of the cluster

-- region → Region where cluster to be created

-- zones → list of availability zones

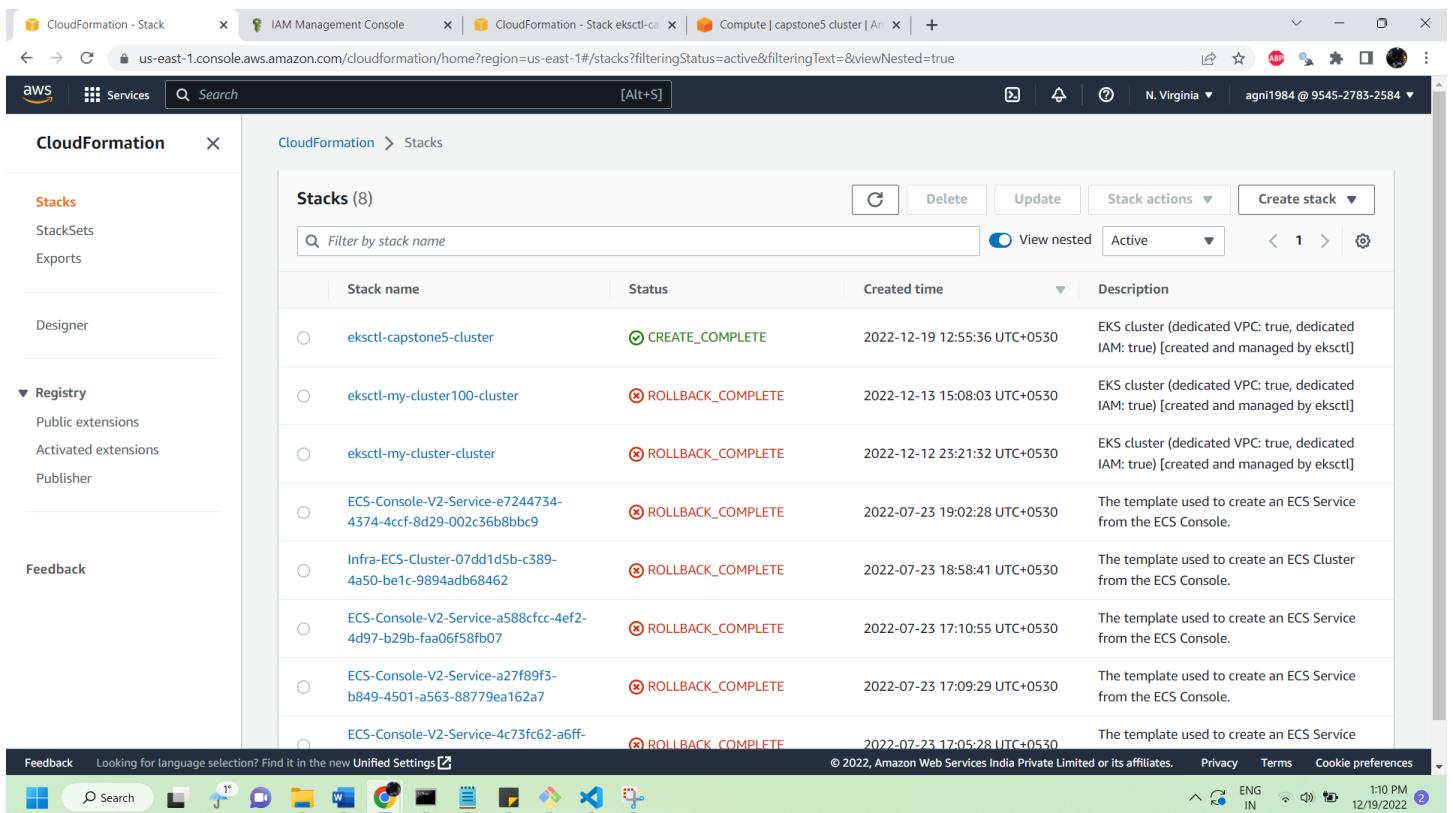
--without-nodegroup → create cluster without nodegroup

```

Administrator Command Prompt
2022-12-19 12:55:35 [ ] subnets for us-east-1c - public:192.168.64.0/19 private:192.168.160.0/19
2022-12-19 12:55:35 [ ] using Kubernetes version 1.23
2022-12-19 12:55:35 [ ] creating EKS cluster "capstone5" in "us-east-1" region with
2022-12-19 12:55:35 [ ] if you encounter any issues, check CloudFormation console or try 'eksctl utils describe-stacks --region=us-east-1 --cluster=capstone5'
2022-12-19 12:55:35 [ ] Kubernetes API endpoint access will use default of (publicAccess=true, privateAccess=false) for cluster "capstone5" in "us-east-1"
2022-12-19 12:55:35 [ ] CloudWatch logging will not be enabled for cluster "capstone5" in "us-east-1"
2022-12-19 12:55:35 [ ] you can enable it with 'eksctl utils update-cluster-logging --enable-types={SPECIFY-YOUR-LOG-TYPES-HERE (e.g. all)} --region=us-east-1 --cluster=capstone5'
2022-12-19 12:55:35 [ ] 2 sequential tasks: { create cluster control plane "capstone5", wait for control plane to become ready
}
2022-12-19 12:55:35 [ ] building cluster stack "eksctl-capstone5-cluster"
2022-12-19 12:55:37 [ ] deploying stack "eksctl-capstone5-cluster"
2022-12-19 12:56:07 [ ] waiting for CloudFormation stack "eksctl-capstone5-cluster"
2022-12-19 12:59:17 [ ] waiting for CloudFormation stack "eksctl-capstone5-cluster"
2022-12-19 13:00:18 [ ] waiting for CloudFormation stack "eksctl-capstone5-cluster"
2022-12-19 13:01:19 [ ] waiting for CloudFormation stack "eksctl-capstone5-cluster"
2022-12-19 13:02:20 [ ] waiting for CloudFormation stack "eksctl-capstone5-cluster"
2022-12-19 13:03:21 [ ] waiting for CloudFormation stack "eksctl-capstone5-cluster"
2022-12-19 13:04:22 [ ] waiting for CloudFormation stack "eksctl-capstone5-cluster"
2022-12-19 13:05:23 [ ] waiting for CloudFormation stack "eksctl-capstone5-cluster"
2022-12-19 13:06:24 [ ] waiting for CloudFormation stack "eksctl-capstone5-cluster"
2022-12-19 13:07:25 [ ] waiting for CloudFormation stack "eksctl-capstone5-cluster"
2022-12-19 13:09:36 [ ] waiting for the control plane to become ready
2022-12-19 13:09:37 [✓] saved kubeconfig as "C:\Users\agnil\kube\config"
2022-12-19 13:09:37 [ ] no tasks
2022-12-19 13:09:37 [✓] all EKS cluster resources for "capstone5" have been created
2022-12-19 13:09:41 [✓] kubectl command should work with "C:\Users\agnil\kube\config", try 'kubectl get nodes'
2022-12-19 13:09:41 [✓] EKS cluster "capstone5" in "us-east-1" region is ready

```

Cloud Formation for cluster creation is completed and all resources created as part of the EKS cluster listed in 'Resources' section.



The screenshot shows the AWS CloudFormation console interface. On the left, there's a navigation sidebar with sections for CloudFormation, Stacks, Designer, Registry, and Feedback. The 'Stacks' section is currently selected. The main area displays a table titled 'Stacks (8)' with columns for Stack name, Status, Created time, and Description. The 'eksctl-capstone5-cluster' stack is listed with a status of 'CREATE_COMPLETE' and a creation time of '2022-12-19 12:55:36 UTC+0530'. Other stacks listed include 'eksctl-my-cluster100-cluster', 'eksctl-my-cluster-cluster', and several ECS-related stacks like 'ECS-Console-V2-Service-e7244734-4374-4ccf-8d29-002c36b8bbc9' and 'Infra-ECS-Cluster-07dd1d5b-c389-4a50-be1c-9894adb68462'. The table includes filters for stack name, nested stacks, and active status, along with buttons for Create stack, Delete, Update, and Stack actions.

Stack name	Status	Created time	Description
eksctl-capstone5-cluster	CREATE_COMPLETE	2022-12-19 12:55:36 UTC+0530	EKS cluster (dedicated VPC: true, dedicated IAM: true) [created and managed by eksctl]
eksctl-my-cluster100-cluster	ROLLBACK_COMPLETE	2022-12-13 15:08:03 UTC+0530	EKS cluster (dedicated VPC: true, dedicated IAM: true) [created and managed by eksctl]
eksctl-my-cluster-cluster	ROLLBACK_COMPLETE	2022-12-12 23:21:32 UTC+0530	EKS cluster (dedicated VPC: true, dedicated IAM: true) [created and managed by eksctl]
ECS-Console-V2-Service-e7244734-4374-4ccf-8d29-002c36b8bbc9	ROLLBACK_COMPLETE	2022-07-23 19:02:28 UTC+0530	The template used to create an ECS Service from the ECS Console.
Infra-ECS-Cluster-07dd1d5b-c389-4a50-be1c-9894adb68462	ROLLBACK_COMPLETE	2022-07-23 18:58:41 UTC+0530	The template used to create an ECS Cluster from the ECS Console.
ECS-Console-V2-Service-a588cfcc-4ef2-4d97-b29b-faa06f58fb07	ROLLBACK_COMPLETE	2022-07-23 17:10:55 UTC+0530	The template used to create an ECS Service from the ECS Console.
ECS-Console-V2-Service-a27f89f3-b849-4501-a563-88779ea162a7	ROLLBACK_COMPLETE	2022-07-23 17:09:29 UTC+0530	The template used to create an ECS Service from the ECS Console.
ECS-Console-V2-Service-4c73fc62-a6ff-	ROLLBACK_COMPLETE	2022-07-23 17:05:28 UTC+0530	The template used to create an ECS Service

CloudFormation - Stack eksctl-capstone5-cluster | IAM Management Console | CloudFormation - Stack eksctl-capstone5-cluster | Compute | capstone5 cluster | +

aws Services Search [Alt+S]

CloudFormation

- Stacks
 - Stack details**
 - Drifts
 - StackSets
 - Exports
- Designer
- Registry
 - Public extensions
 - Activated extensions
 - Publisher
- Feedback

CloudFormation > Stacks > eksctl-capstone5-cluster

eksctl-capstone5-cluster

Stack info Events **Resources** Outputs Parameters Template Change sets

Resources (34)

Logical ID	Physical ID	Type	Status	Module
ClusterSharedNodeSecurityGroup	sg-0292e1c2c5823ed24	AWS::EC2::Security Group	CREATE_COMPLETE	-
ControlPlane	capstone5	AWS::EKS::Cluster	CREATE_COMPLETE	-
ControlPlaneSecurityGroup	sg-02a6bdb4673e1008e	AWS::EC2::Security Group	CREATE_COMPLETE	-
IngressDefaultClusterToNodeSG	IngressDefaultClusterToNodeSG	AWS::EC2::Security GroupIngress	CREATE_COMPLETE	-
IngressInterNodeGroupSG	IngressInterNodeGroupSG	AWS::EC2::Security GroupIngress	CREATE_COMPLETE	-

Feedback Looking for language selection? Find it in the new Unified Settings 

© 2022, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

ENG IN 1:1 PM 12/19/2022

CloudFormation - Stack eksctl-capstone5-cluster | IAM Management Console | CloudFormation - Stack eksctl-capstone5-cluster | Compute | capstone5 cluster | +

aws Services Search [Alt+S]

CloudFormation

- Stacks
 - Stack details**
 - Drifts
 - StackSets
 - Exports
- Designer
- Registry
 - Public extensions
 - Activated extensions
 - Publisher
- Feedback

CloudFormation > Stacks > eksctl-capstone5-cluster

Resources (34)

Logical ID	Physical ID	Type	Status	Module
InternetGateway	igw-0b032ce79d84a76bc	AWS::EC2::Internet Gateway	CREATE_COMPLETE	-
NATGateway	nat-05c76f78785b69fa1	AWS::EC2::NatGateway	CREATE_COMPLETE	-
NATIP	52.20.237.40	AWS::EC2::EIP	CREATE_COMPLETE	-
NATPrivateSubnetRouteUSEAST1A	eksct-NATProuteUSEAST1A-A91J2FO15A86	AWS::EC2::Route	CREATE_COMPLETE	-
NATPrivateSubnetRouteUSEAST1B	eksct-NATProuteUSEAST1B-FTSQY4G2DJ6U	AWS::EC2::Route	CREATE_COMPLETE	-
NATPrivateSubnetRouteUSEAST1C	eksct-NATProuteUSEAST1C-SBMPD6PFW350	AWS::EC2::Route	CREATE_COMPLETE	-
PolicyCloudWatchMetrics	eksct-Poli-QSG637YSWOAQ	AWS::IAM::Policy	CREATE_COMPLETE	-
PolicyELBPermissions	eksct-Poli-17AGQY7Q7LJ3O	AWS::IAM::Policy	CREATE_COMPLETE	-

Feedback Looking for language selection? Find it in the new Unified Settings 

© 2022, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

ENG IN 1:1 PM 12/19/2022

CloudFormation - Stack eksctl-capstone5-cluster | IAM Management Console | CloudFormation - Stack eksctl-capstone5-cluster | Compute | capstone5 cluster | +

CloudFormation Services Search [Alt+S]

CloudFormation

- Stacks
 - Stack details**
 - Drifts
 - StackSets
 - Exports
- Designer
- Registry
 - Public extensions
 - Activated extensions
 - Publisher
- Feedback

CloudFormation > Stacks > eksctl-capstone5-cluster

Stacks (8)

Filter by stack name Active View nested

Logical ID	Physical ID	Type	Status	Module
USEAST1A	6c	ble	CREATE_COMPLETE	-
PrivateRouteTable USEAST1B	rtb-0085539b3f59311be	AWS::EC2::RouteTable	CREATE_COMPLETE	-
PrivateRouteTable USEAST1C	rtb-0e00a1b7b9df5717d	AWS::EC2::RouteTable	CREATE_COMPLETE	-
PublicRouteTable	rtb-00575662996f090b4	AWS::EC2::RouteTable	CREATE_COMPLETE	-
PublicSubnetRoute	eksct-Publi-HZEZYM388W53	AWS::EC2::Route	CREATE_COMPLETE	-
RouteTableAssociationPrivateUSEAST1A	rtbassoc-0cb4e7b60f0017158	AWS::EC2::SubnetRouteTableAssociation	CREATE_COMPLETE	-
RouteTableAssociationPrivateUSEAST1B	rtbassoc-0fb5316400f8fcfef	AWS::EC2::SubnetRouteTableAssociation	CREATE_COMPLETE	-
RouteTableAssociationPrivateUSEAST1C	rtbassoc-0fb5316400f8fcfef	AWS::EC2::SubnetRouteTableAssociation	CREATE_COMPLETE	-

Feedback Looking for language selection? Find it in the new Unified Settings 

© 2022, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

ENG IN 1:1 PM 12/19/2022

CloudFormation - Stack eksctl-capstone5-cluster | IAM Management Console | CloudFormation - Stack eksctl-capstone5-cluster | Compute | capstone5 cluster | +

CloudFormation Services Search [Alt+S]

CloudFormation

- Stacks
 - Stack details**
 - Drifts
 - StackSets
 - Exports
- Designer
- Registry
 - Public extensions
 - Activated extensions
 - Publisher
- Feedback

CloudFormation > Stacks > eksctl-capstone5-cluster

Stacks (8)

Filter by stack name Active View nested

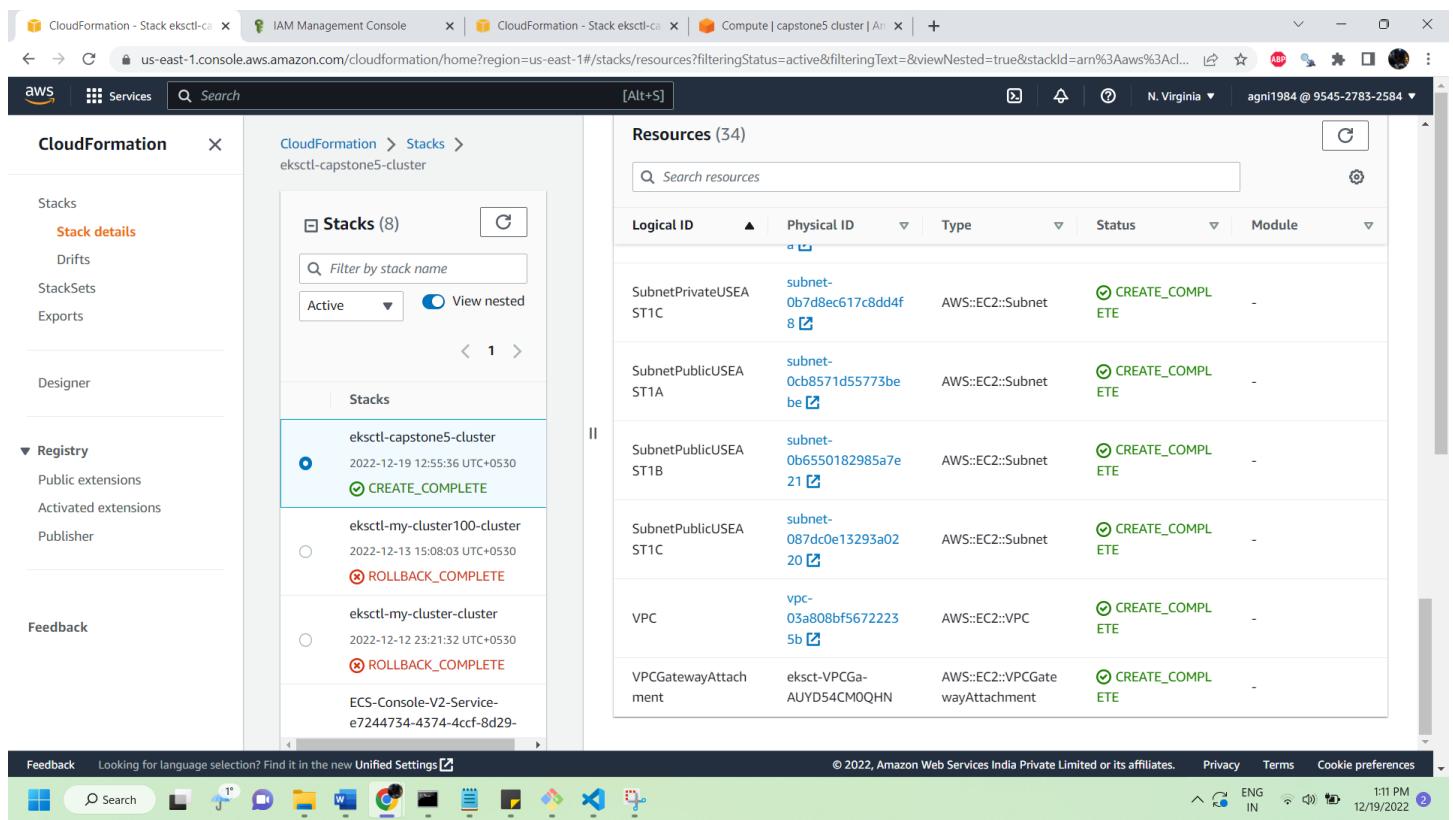
Stacks

Logical ID	Physical ID	Type	Status	Module
ionPublicUSEAST1C	0f99f7f7376a33fb	outeTableAssociation	CREATE_COMPLETE	-
eksctl-capstone5-cluster-ServiceRole-DPACYQDQBRVB		AWS::IAM::Role	CREATE_COMPLETE	-
SubnetPrivateUSEAST1A	subnet-0e474c33ba2e40fd6	AWS::EC2::Subnet	CREATE_COMPLETE	-
SubnetPrivateUSEAST1B	subnet-0b21a0fe9eecd6da	AWS::EC2::Subnet	CREATE_COMPLETE	-
SubnetPrivateUSEAST1C	subnet-0b7d8ec617c8dd4f8	AWS::EC2::Subnet	CREATE_COMPLETE	-
SubnetPublicUSEAST1A	subnet-0cb8571d55773bbe	AWS::EC2::Subnet	CREATE_COMPLETE	-
SubnetPublicUSEAST1B	subnet-0cb8571d55773bbe	AWS::EC2::Subnet	CREATE_COMPLETE	-

Feedback Looking for language selection? Find it in the new Unified Settings 

© 2022, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

ENG IN 1:1 PM 12/19/2022



The screenshot shows the AWS CloudFormation console. On the left, the navigation pane includes 'CloudFormation', 'Stacks', 'Drifts', 'StackSets', 'Exports', 'Designer', 'Registry' (with 'Public extensions' and 'Activated extensions'), and 'Feedback'. The main area displays the 'eksctl-capstone5-cluster' stack details, which lists 8 stacks. One stack, 'eksctl-capstone5-cluster', is selected and shows its status as 'CREATE_COMPLETE'. The 'Resources' table on the right lists 34 resources, including subnets, a VPC, and a VPC gateway attachment, all in 'CREATE_COMPLETE' status.

Query the status of our cluster with the following command.

```
aws eks describe-cluster --region us-east-1 --name capstone5 --query "cluster.status"
```

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>aws eks describe-cluster --region us-east-1 --name capstone5 --query "cluster.status"
"ACTIVE"
```

Enable kubectl to communicate with our cluster by adding a new context to the kubectl config file.

```
aws eks update-kubeconfig --region us-east-1 --name capstone5
```

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>aws eks update-kubeconfig --region us-east-1 --name capstone5
Updated context arn:aws:eks:us-east-1:954527832584:cluster/capstone5 in C:\Users\agni1\.kube\config
```

Confirm communication with our cluster by running the following command.

```
kubectl get svc
```

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>kubectl get svc
NAME      TYPE      CLUSTER-IP    EXTERNAL-IP    PORT(S)    AGE
kubernetes  ClusterIP  10.100.0.1   <none>        443/TCP   24m
```

Create an IAM OIDC identity provider for our cluster with the following command.

```
eksctl utils associate-iam-oidc-provider --region us-east-1 --cluster capstone5 --approve
```

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>eksctl utils associate-iam-oidc-provider --region us-east-1 --cluster capstone5 --approve
2022-12-19 13:18:41 [!] will create IAM Open ID Connect provider for cluster "capstone5" in "us-east-1"
2022-12-19 13:18:43 [✓] created IAM Open ID Connect provider for cluster "capstone5" in "us-east-1"
```

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5>aws eks update-kubeconfig --name capstone5
Added new context arn:aws:eks:us-east-1:954527832584:cluster/capstone5 to C:\Users\agni1\.kube\config
```

4.2. Create Node Group

```
eksctl create nodegroup --cluster=capstone5 --region=us-east-1 --name=capstone5-ng --node-type=t3.medium --nodes=2 --nodes-min=2 --nodes-max=5 --node-volume-size=20 --ssh-access --ssh-public-key=capstonekey --managed --asg-access --external-dns-access --full-ecr-access --node-private-networking
```

--cluster → cluster name

--region → region where cluster presented

--name → name of the node group

--node-type → EC2 instance type in node group

--nodes → number of default nodes (EC2 instances)

--nodes-min → minimum number of nodes (EC2) in cluster

--nodes-max → maximum number of nodes (EC2) in cluster

--node-volume → EBS volume in GB

--ssh access → allow SSH access

--ssh-public-key → public key to link with EC2 instance. Using this key, we can access EC2 instance through SSH. Ensure that key is already created in AWS

--asg-access → include Auto scaling

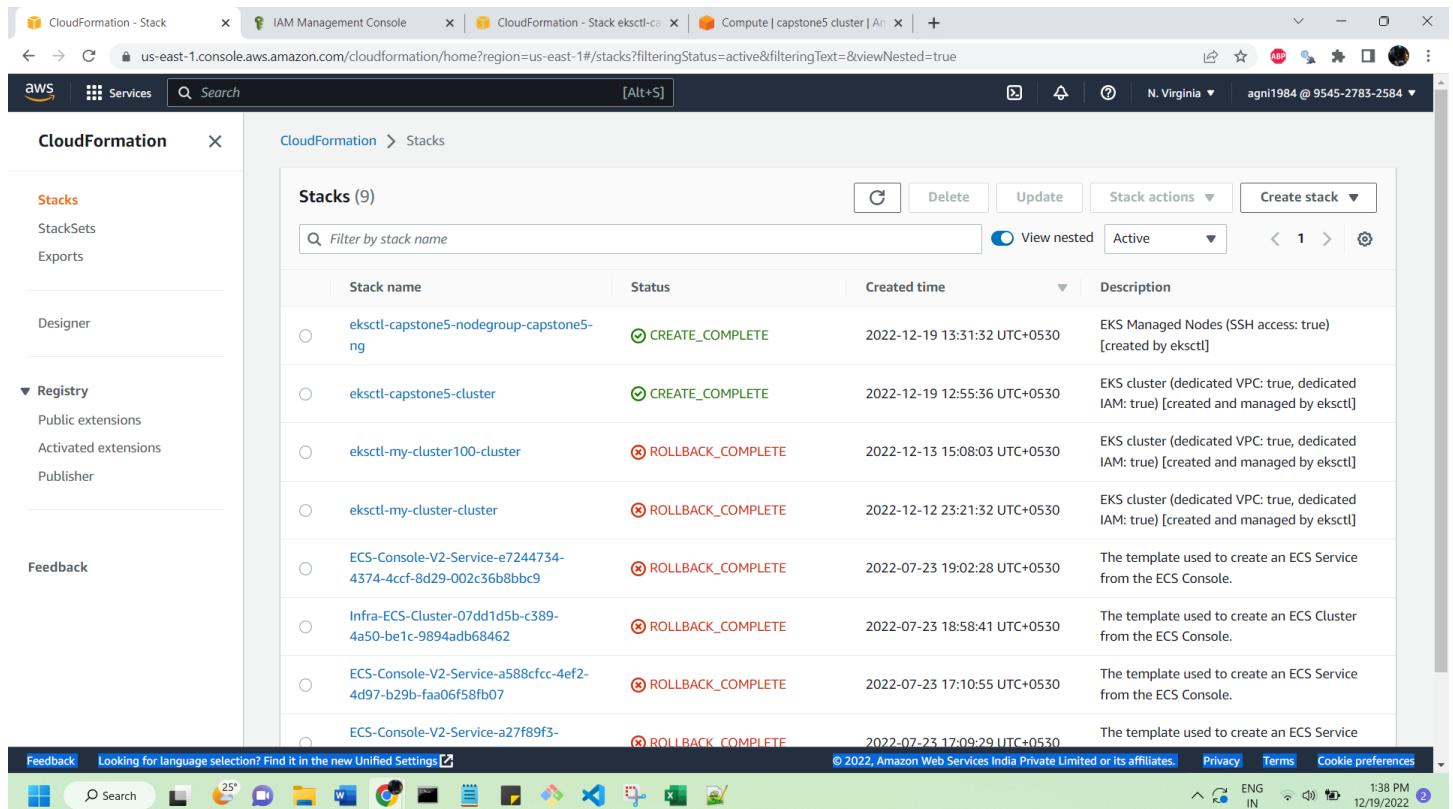
--external-dns-access →

--full-ecr-access → (optional) not required for this project

--node-private-networking → node group created in private subnet

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>eksctl create nodegroup --cluster=capstone5 --region=us-east-1 --name=capstone5-ng --node-type=t3.medium --nodes=2 --nodes-min=2 --nodes-max=5 --node-volume-size=20 --ssh-access --ssh-public-key=capstonekey --managed --asg-access --external-dns-access --full-ecr-access --node-private-networking
2022-12-19 13:31:22 [ ] will use version 1.23 for new nodegroup(s) based on control plane version
2022-12-19 13:31:27 [ ] nodegroup "capstone5-ng" will use "AmazonLinux2/1.23"
2022-12-19 13:31:28 [ ] using EC2 key pair %q(string=<nil>)
2022-12-19 13:31:29 [ ] 1 nodegroup (capstone5-ng) was included (based on the include/exclude rules)
2022-12-19 13:31:29 [ ] will create a CloudFormation stack for each of 1 managed nodegroups in cluster "capstone5"
2022-12-19 13:31:29 [ ] 2 sequential tasks: { fix cluster compatibility, 1 task: { create managed nodegroup "capstone5-ng" } }
2022-12-19 13:31:29 [ ] checking cluster stack for missing resources
2022-12-19 13:31:30 [ ] cluster stack has all required resources
2022-12-19 13:31:31 [ ] building managed nodegroup stack "eksctl-capstone5-nodegroup-capstone5-ng"
2022-12-19 13:31:32 [ ] deploying stack "eksctl-capstone5-nodegroup-capstone5-ng"
2022-12-19 13:31:32 [ ] waiting for CloudFormation stack "eksctl-capstone5-nodegroup-capstone5-ng"
2022-12-19 13:32:03 [ ] waiting for CloudFormation stack "eksctl-capstone5-nodegroup-capstone5-ng"
2022-12-19 13:32:40 [ ] waiting for CloudFormation stack "eksctl-capstone5-nodegroup-capstone5-ng"
2022-12-19 13:34:06 [ ] waiting for CloudFormation stack "eksctl-capstone5-nodegroup-capstone5-ng"
2022-12-19 13:35:53 [ ] waiting for CloudFormation stack "eksctl-capstone5-nodegroup-capstone5-ng"
2022-12-19 13:35:54 [ ] no tasks
2022-12-19 13:35:54 [✓] created 0 nodegroup(s) in cluster "capstone5"
2022-12-19 13:35:55 [ ] nodegroup "capstone5-ng" has 2 node(s)
2022-12-19 13:35:55 [ ] node "ip-192-168-135-51.ec2.internal" is ready
2022-12-19 13:35:55 [ ] node "ip-192-168-164-85.ec2.internal" is ready
2022-12-19 13:35:55 [ ] waiting for at least 2 node(s) to become ready in "capstone5-ng"
2022-12-19 13:35:55 [ ] nodegroup "capstone5-ng" has 2 node(s)
2022-12-19 13:35:55 [ ] node "ip-192-168-135-51.ec2.internal" is ready
2022-12-19 13:35:55 [ ] node "ip-192-168-164-85.ec2.internal" is ready
2022-12-19 13:35:55 [✓] created 1 managed nodegroup(s) in cluster "capstone5"
2022-12-19 13:35:56 [ ] checking security group configuration for all nodegroups
2022-12-19 13:35:56 [ ] all nodegroups have up-to-date cloudformation templates
```

Cloud formation for node group creation is completed.



Stack name	Status	Created time	Description
eksctl-capstone5-nodegroup-capstone5-ng	CREATE_COMPLETE	2022-12-19 13:31:32 UTC+0530	EKS Managed Nodes (SSH access: true) [created by eksctl]
eksctl-capstone5-cluster	CREATE_COMPLETE	2022-12-19 12:55:36 UTC+0530	EKS cluster (dedicated VPC: true, dedicated IAM: true) [created and managed by eksctl]
eksctl-my-cluster100-cluster	ROLLBACK_COMPLETE	2022-12-13 15:08:03 UTC+0530	EKS cluster (dedicated VPC: true, dedicated IAM: true) [created and managed by eksctl]
eksctl-my-cluster-cluster	ROLLBACK_COMPLETE	2022-12-12 23:21:32 UTC+0530	EKS cluster (dedicated VPC: true, dedicated IAM: true) [created and managed by eksctl]
ECS-Console-V2-Service-e7244734-4374-4ccf-8d29-002c36b8bbc9	ROLLBACK_COMPLETE	2022-07-23 19:02:28 UTC+0530	The template used to create an ECS Service from the ECS Console.
Infra-ECS-Cluster-07dd1d5b-c389-4a50-be1c-9894adb68462	ROLLBACK_COMPLETE	2022-07-23 18:58:41 UTC+0530	The template used to create an ECS Cluster from the ECS Console.
ECS-Console-V2-Service-a588cfcc-4ef2-4d97-b29b-faa06f58fb07	ROLLBACK_COMPLETE	2022-07-23 17:10:55 UTC+0530	The template used to create an ECS Service from the ECS Console.
ECS-Console-V2-Service-a27f89f3-	ROLLBACK_COMPLETE	2022-07-23 17:09:29 UTC+0530	The template used to create an ECS Service

list of resources as part of the node group creation, are shown below

Screenshot of the AWS CloudFormation console showing the creation of an EKS cluster named 'capstone5'. The 'Resources' table lists seven items, all in 'CREATE_COMPLETE' status:

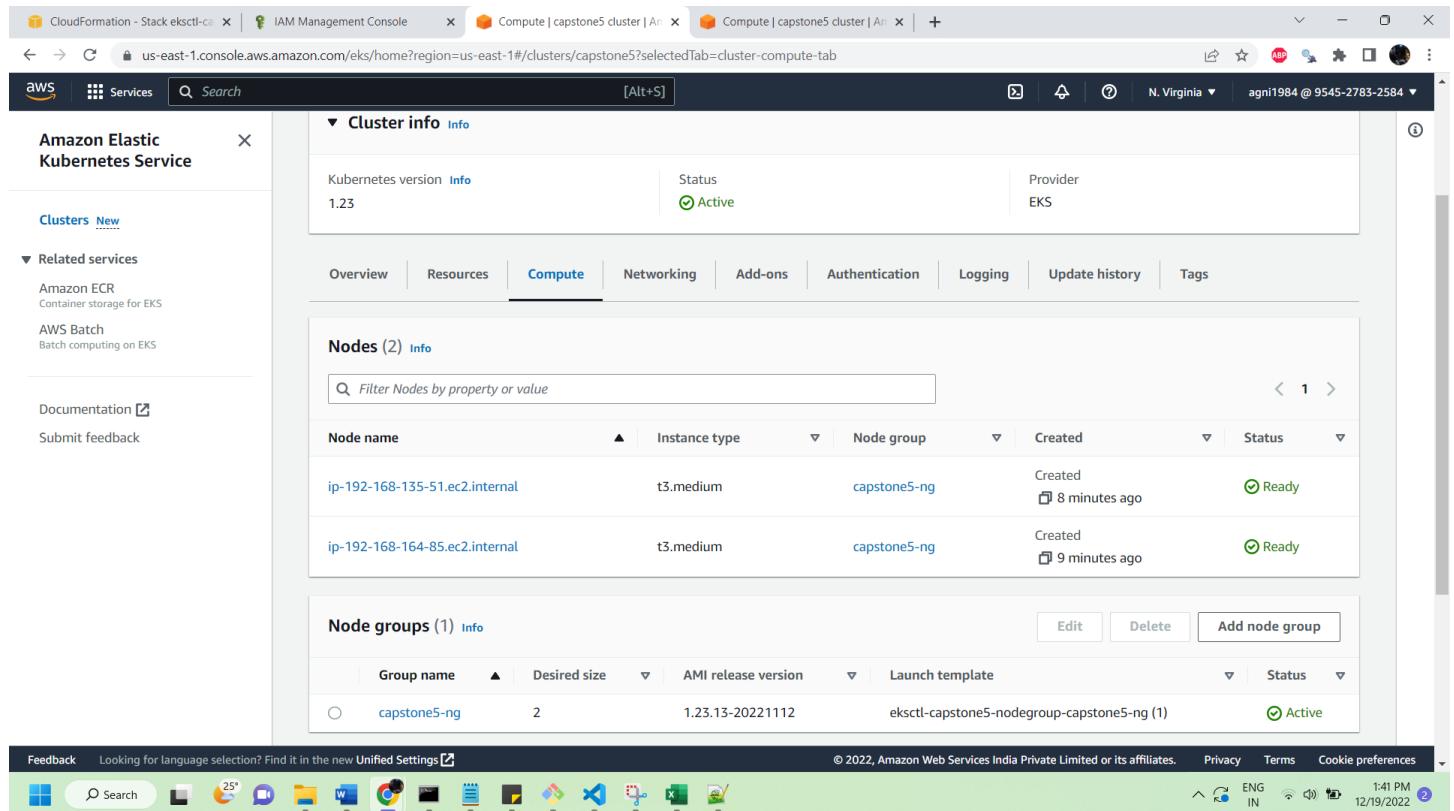
Logical ID	Physical ID	Type	Status	Module
LaunchTemplate	eksctl-launchtemplate-7	LaunchTemplate	CREATE_COMPLETE	-
ManagedNodeGroup	capstone5/capstone5-ng	AWS::EKS::Nodegroup	CREATE_COMPLETE	-
NodeInstanceRole	eksctl-capstone5-nodegroup-capstone5-NodeInstanceRole-15ZLOY162DRYG	AWS::IAM::Role	CREATE_COMPLETE	-
PolicyAutoScaling	eksctl-PolicyAutoScaling-Q1CT3WTCBIU2	AWS::IAM::Policy	CREATE_COMPLETE	-
PolicyExternalDNSChangeSet	eksctl-PolicyExternalDNSChangeSet-17GKG9H4TA6IQ	AWS::IAM::Policy	CREATE_COMPLETE	-
PolicyExternalDNSHostedZones	eksctl-PolicyExternalDNSHostedZones-ESG2DXPAON66	AWS::IAM::Policy	CREATE_COMPLETE	-
SSH	sg-0134a3f539135869	AWS::EC2::SecurityGroup	CREATE_COMPLETE	-

Go to “Elastic Kubernetes service” in AWS console. Created cluster is listed under ‘cluster’ section

Screenshot of the AWS EKS console showing the created cluster 'capstone5'. The 'Clusters' table shows one cluster in 'Active' status:

Cluster name	Status	Kubernetes version	Provider
capstone5	Active	1.23	EKS

Got to compute section in cluster. We can see the nodegroup details and EC2 instances details which are part of node group.



Cluster info

Kubernetes version Info 1.23	Status  Active	Provider EKS
---	---	--------------

Nodes (2) Info

Filter Nodes by property or value					
Node name	Instance type	Node group	Created	Status	
ip-192-168-135-51.ec2.internal	t3.medium	capstone5-ng	Created 8 minutes ago	 Ready	
ip-192-168-164-85.ec2.internal	t3.medium	capstone5-ng	Created 9 minutes ago	 Ready	

Node groups (1) Info

Group name	Desired size	AMI release version	Launch template	Status
capstone5-ng	2	1.23.13-20221112	eksctl-capstone5-nodegroup-capstone5-ng (1)	 Active

Click on 'node name'. we can see node details and its capacity allocation details.

Screenshot of the AWS CloudFormation console showing multiple tabs open. The main tab displays the details of a node in an EKS cluster named 'capstone5'.

Node Details:

- Status:** Ready
- Kernel version:** 5.4.219-126.411.amzn2.x86_64
- Created:** 12 minutes ago
- Last transition time:** 12 minutes ago
- Node group:** capstone5-ng
- Container runtime:** docker://20.10.17
- OS (Architecture):** linux (amd64)
- Instance:** i-0ff959beb64cf916c
- Kubelet version:** v1.23.13-eks-fb459a0
- OS image:** Amazon Linux 2
- Instance type:** t3.medium

Capacity allocation:



Resource Type	Available	Used	Percentage
Cores	1805 m	2 pods	90%, 12%
Memory	3.25 GB		86%

List of resources as part of cluster are shown under 'resources' section in AWS console.

Screenshot of the AWS EKS console showing the 'Resources' tab for the 'capstone5' cluster.

Cluster Info:

- Kubernetes version:** 1.23
- Status:** Active
- Provider:** EKS

Resource types:

- Workloads
- Cluster
- Service and networking
- Config and secrets
- Storage

Cluster: Nodes (2):

Node name	Instance type	Node group	Created	Status
ip-192-168-135-51.ec2.internal	t3.medium	capstone5-ng	9 minutes ago	Ready

4.3. Get Cluster, node groups, nodes

```
eksctl get cluster
```

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>eksctl get cluster
NAME      REGION      EKSCTL CREATED
capstone5  us-east-1  True
```

```
eksctl get nodegroup --cluster=capstone5
```

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>eksctl get nodegroup --cluster=capstone5
CLUSTER      NODEGROUP      STATUS      CREATED      MIN SIZE      MAX SIZE      DESIRED CAPACITY      INSTANCE TYPE      IMAGE ID      ASG NAME      TY
E           capstone5-ng    ACTIVE     2022-12-19T08:02:01Z    2             5             2                  t3.medium      AL2_x86_64      eks-capstone5-ng-60c294b3-b163-6a21-782f-293de5eddf7  ma
aged
```

```
kubectl get nodes -o wide
```

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>kubectl get nodes -o wide
NAME          STATUS  ROLES   AGE   VERSION      INTERNAL-IP      EXTERNAL-IP      OS-IMAGE      KERNEL-VERSION      CONTAINER-RUNTIME
ip-192-168-135-51.ec2.internal  Ready  <none>  11m  v1.23.13-eks-fb459a0  192.168.135.51  <none>        Amazon Linux 2  5.4.219-126.411.amzn2.x86_64  docker://20.10.17
ip-192-168-164-85.ec2.internal  Ready  <none>  11m  v1.23.13-eks-fb459a0  192.168.164.85  <none>        Amazon Linux 2  5.4.219-126.411.amzn2.x86_64  docker://20.10.17
```

As nodegroup is created in private subnet, “External IP Address” won’t be assigned to EC2 instances



Screenshot of the AWS CloudFormation console showing the Instances page. The URL is [https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#InstancesinstanceState=running:v=3;\\$case=true%5C;client=false;\\$regex=tags:false%5C;client=false](https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#InstancesinstanceState=running:v=3;$case=true%5C;client=false;$regex=tags:false%5C;client=false). The page displays two running t3.medium instances named 'capstone5-capstone5-Node'. The sidebar includes sections for EC2 Dashboard, New EC2 Experience, Instances, Images, Elastic Block Store, Network & Security, and Load Balancing.

Screenshot of the AWS CloudFormation console showing the Instance details page for instance i-0f00dc0da42f436a2. The URL is <https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#InstanceDetails:instanceId=i-0f00dc0da42f436a2>. The page provides detailed information about the instance, including its configuration, monitoring status, and security settings. The sidebar is identical to the previous screenshot.



CloudFormati... IAM Manager Resources | ca... ip-192-168-1... ip-192-168-1... Compute | ca... Instances | EC... Instance detail... Instance detail...

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#InstanceDetails:instanceId=i-0ff959beb64cf916c

AWS Services Search [Alt+5]

New EC2 Experience Tell us what you think

EC2 Dashboard EC2 Global View Events Tags Limits

Instances Instances Instance Types Launch Templates Spot Requests Savings Plans Reserved Instances Dedicated Hosts Scheduled Instances Capacity Reservations

Images AMIs AMI Catalog

Elastic Block Store Volumes Snapshots Lifecycle Manager

Network & Security Security Groups Elastic IPs Placement Groups Key Pairs Network Interfaces

Load Balancing Load Balancers

Feedback Looking for language selection? Find it in the new Unified Settings

Instance summary for i-0ff959beb64cf916c (capstone5-capstone5-Node)

Updated less than a minute ago

Public IPv4 address: 192.168.135.51

Private IP DNS name (IPv4 only): ip-192-168-135-51.ec2.internal

Instance state: Running

Hostname type: IP name: ip-192-168-135-51.ec2.internal

Instance type: t3.medium

VPC ID: vpc-03a808bf56722235b (eksctl-capstone5-cluster/VPC)

Subnet ID: subnet-0b21a0fe9eed6daa (eksctl-capstone5-cluster/SubnetPrivateUSEAST1B)

Tags

Details Security Networking Storage Status checks Monitoring Tags

AMI ID: ami-0c942a408e18bcc9

AMI name: amazon-eks-node-1.23-v20221112

Launch time: Mon Dec 19 2022 13:32:23 GMT+0530 (India Standard Time) (16 minutes)

Lifecycle: normal

Key pair name: capstonekey

Kernel ID: -

RAM disk ID: -

Enclaves Support: Disabled

Monitoring: disabled

Termination protection: Disabled

AMI location: amazon/amazon-eks-node-1.23-v20221112

Stop-hibernate behavior: disabled

State transition reason: -

State transition message: -

Owner: 954527832584

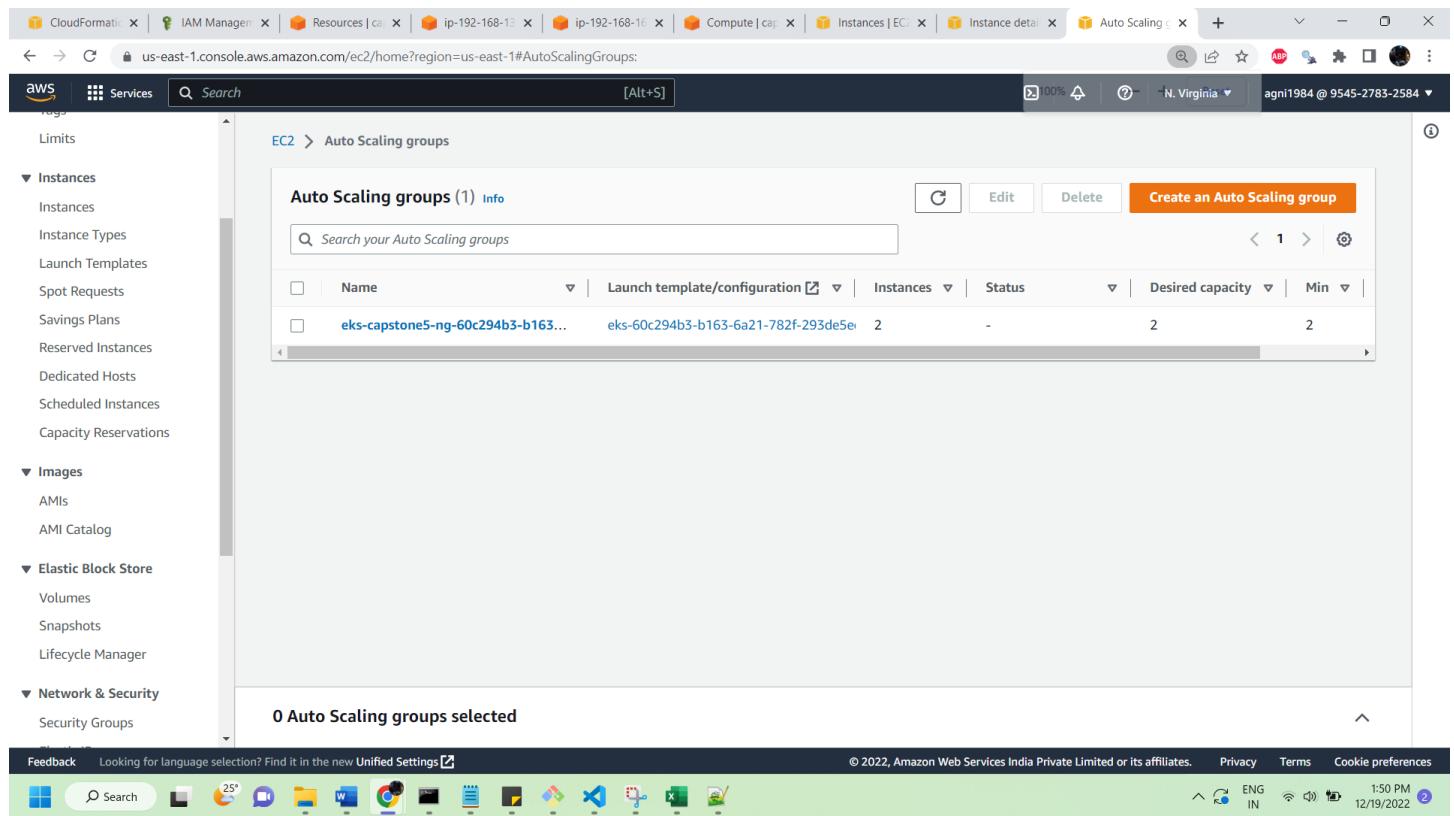
Boot mode: -

© 2022, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

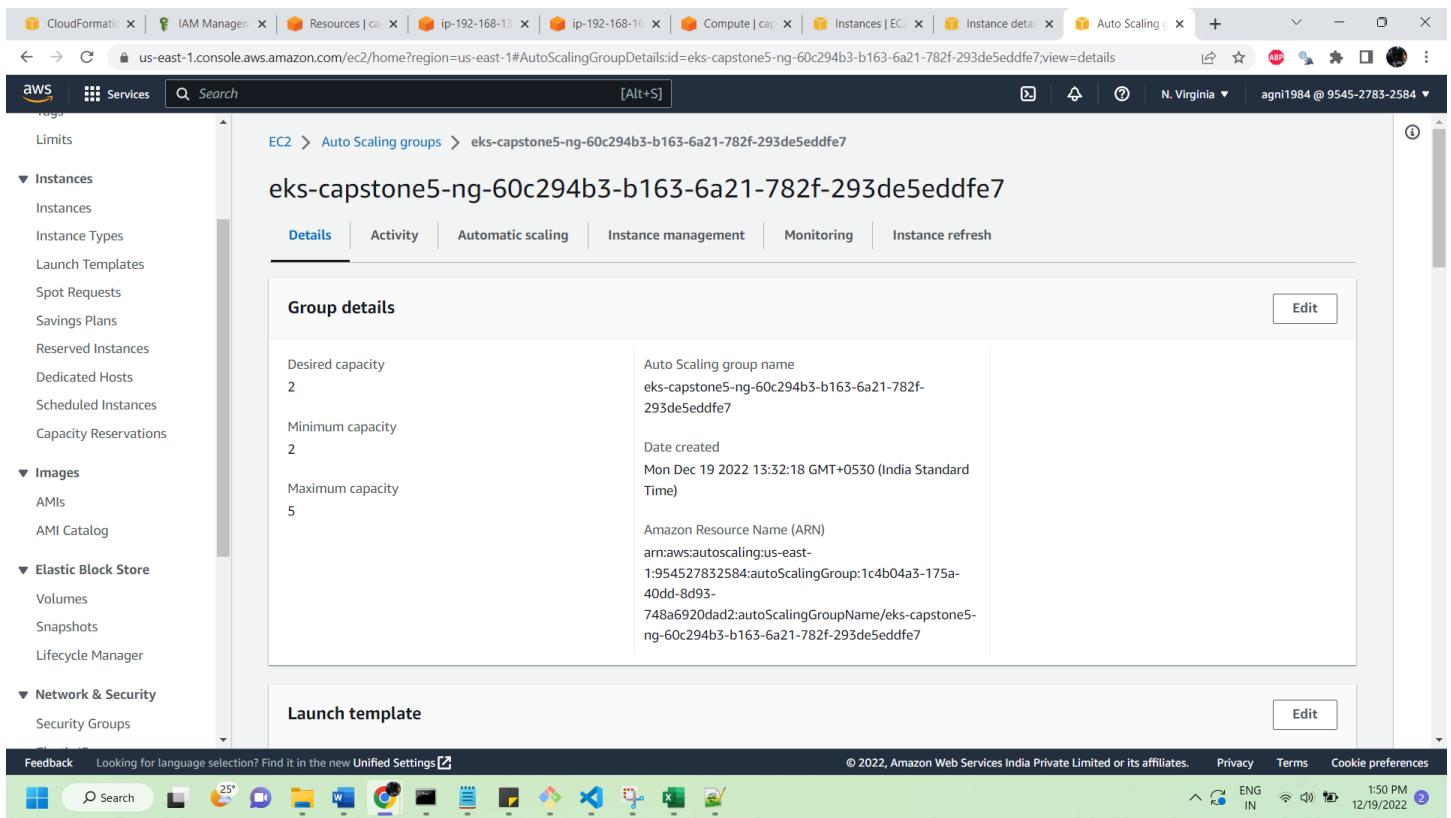
ENG IN 12:48 PM 12/19/2022

5. Auto Scaling Setup

As we created node group with ‘–asg-access’ command ‘Auto Scaling’ group is created automatically.



Name	Launch template/configuration	Instances	Status	Desired capacity	Min
eks-capstone5-ng-60c294b3-b163...	eks-60c294b3-b163-6a21-782f-293de5e...	2	-	2	2



The screenshot shows the AWS CloudFormation console with a stack named "eks-capstone5-ng-60c294b3-b163-6a21-782f-293de5eddf7". The "Group details" section displays the following configuration:

Setting	Value
Desired capacity	2
Minimum capacity	2
Maximum capacity	5

Other details shown include:

- Auto Scaling group name: eks-capstone5-ng-60c294b3-b163-6a21-782f-293de5eddf7
- Date created: Mon Dec 19 2022 13:32:18 GMT+0530 (India Standard Time)
- Amazon Resource Name (ARN): arn:aws:autoscaling:us-east-1:954527832584:autoScalingGroup:1c4b04a3-175a-40dd-8d93-748a6920dad2:autoScalingGroupName/eks-capstone5-ng-60c294b3-b163-6a21-782f-293de5eddf7

However above auto scaling group increase/decreases number of pods in existing node group. It won't add/remove EC2 node. "Cluster AutoScaler" should be configured to do this.

Cluster Autoscaler automatically adjusts the number of nodes in cluster when pods fail or are rescheduled onto other nodes. The Cluster Autoscaler is typically installed as a Deployment in the cluster. It uses leader election to ensure high availability, but scaling is done by only one replica at a time. Follow below step to configure cluster

5.1. Create IAM policy and role

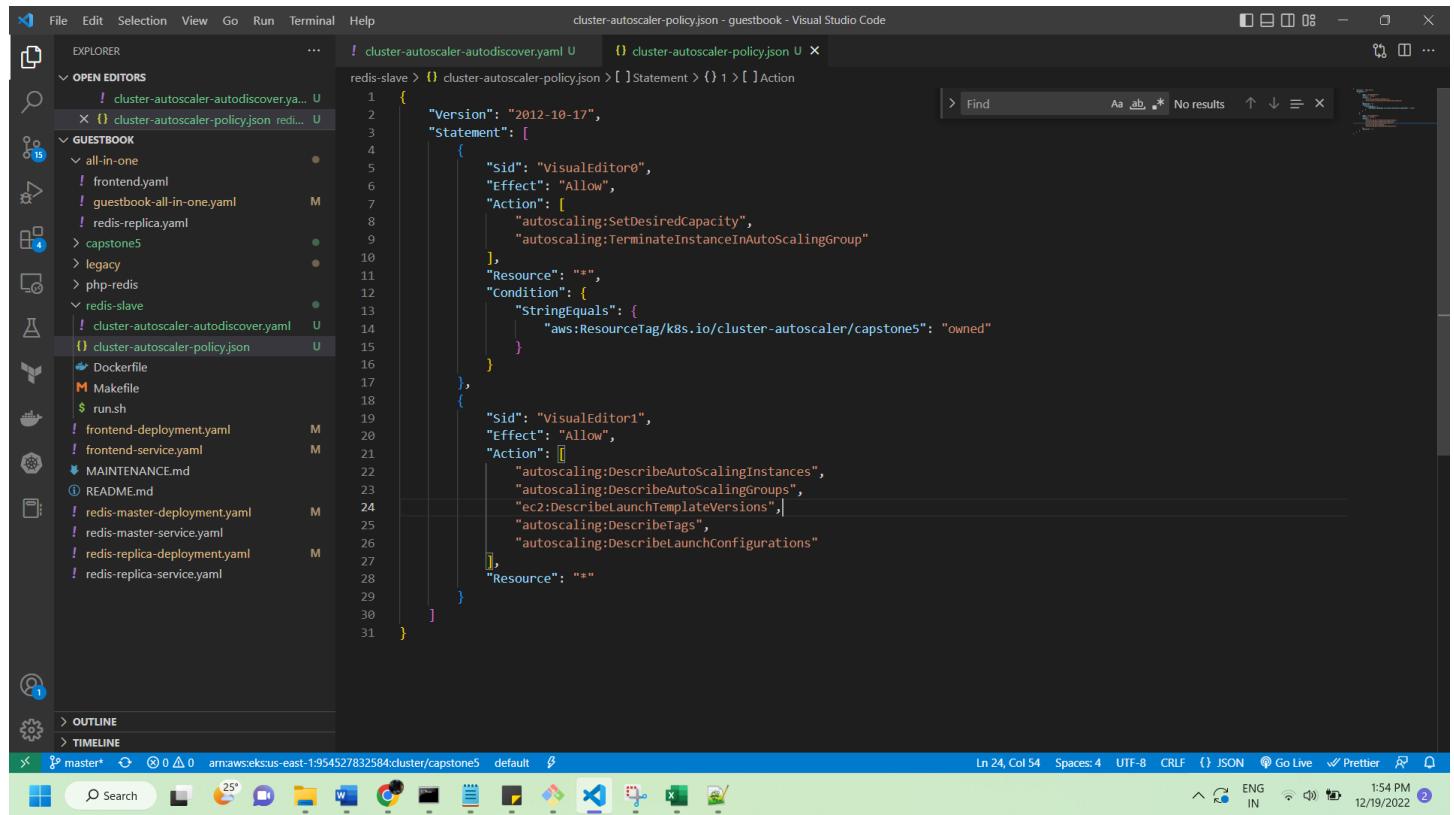
Save the following contents to a file that's named cluster-autoscaler-policy.json

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "autoscaling:SetDesiredCapacity",
                "autoscaling:TerminateInstanceInAutoScalingGroup"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/k8s.io/cluster-autoscaler/capstone5": "owned"
                }
            }
        },
        {
            "Sid": "VisualEditor1",
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogStream",
                "logs:PutLogEvents"
            ],
            "Resource": "arn:aws:logs:*:*/*"
        }
    ]
}
```

```

        "Sid": "VisualEditor1",
        "Effect": "Allow",
        "Action": [
            "autoscaling:DescribeAutoScalingInstances",
            "autoscaling:DescribeAutoScalingGroups",
            "ec2:DescribeLaunchTemplateVersions",
            "autoscaling:DescribeTags",
            "autoscaling:DescribeLaunchConfigurations"
        ],
        "Resource": "*"
    }
}

```



Create the policy with the following command.

```
aws iam create-policy --policy-name AmazonEKSClusterAutoscalerPolicy --policy-document file://cluster-autoscaler-policy.json
```

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\redis-slave>aws iam create-policy --policy-name AmazonEKSClusterAutoscalerPolicy --policy-document file://cluster-autosca
ler-policy.json
{
  "Policy": {
    "PolicyName": "AmazonEKSClusterAutoscalerPolicy",
    "PolicyId": "ANP45PSK3IE0QFEBY2U",
    "Arn": "arn:aws:iam::954527832584:policy/AmazonEKSClusterAutoscalerPolicy",
    "Path": "/",
    "DefaultVersionId": "v1",
    "AttachmentCount": 0,
    "PermissionsBoundaryUsageCount": 0,
    "IsAttachable": true,
    "CreateDate": "2022-12-19T08:27:48+00:00",
    "UpdateDate": "2022-12-19T08:27:48+00:00"
  }
}
```

Create an IAM role and attach an IAM policy to it using eksctl

```
eksctl create iamserviceaccount --cluster=capstone5 --namespace=kube-system --
name=cluster-autoscaler --attach-policy-
arn=arn:aws:iam::954527832584:policy/AmazonEKSClusterAutoscalerPolicy --override-
existing-serviceaccounts --approve
```

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\redis-slave>eksctl create iamserviceaccount --cluster=capstone5 --namespace=kube-system --name=cluster-autoscaler --a
ttach-policy-arn=arn:aws:iam::954527832584:policy/AmazonEKSClusterAutoscalerPolicy --override-existing-serviceaccounts --approve
2022-12-19 14:00:20 [+] 1 iamserviceaccount (kube-system/cluster-autoscaler) was included (based on the include/exclude rules)
2022-12-19 14:00:20 [+] metadata of serviceaccounts that exist in Kubernetes will be updated, as --override-existing-serviceaccounts was set
2022-12-19 14:00:20 [+] 1 task:
  2 sequential sub-tasks:
    create IAM role for serviceaccount "kube-system/cluster-autoscaler",
    create serviceaccount "kube-system/cluster-autoscaler",
  )2022-12-19 14:00:20 [+] building iamserviceaccount stack "eksctl-capstone5-addon-iamserviceaccount-kube-system-cluster-autoscaler"
2022-12-19 14:00:20 [+] deploying stack "eksctl-capstone5-addon-iamserviceaccount-kube-system-cluster-autoscaler"
2022-12-19 14:00:21 [+] waiting for CloudFormation stack "eksctl-capstone5-addon-iamserviceaccount-kube-system-cluster-autoscaler"
2022-12-19 14:00:51 [+] waiting for CloudFormation stack "eksctl-capstone5-addon-iamserviceaccount-kube-system-cluster-autoscaler"
2022-12-19 14:00:53 [+] created serviceaccount "kube-system/cluster-autoscaler"
```

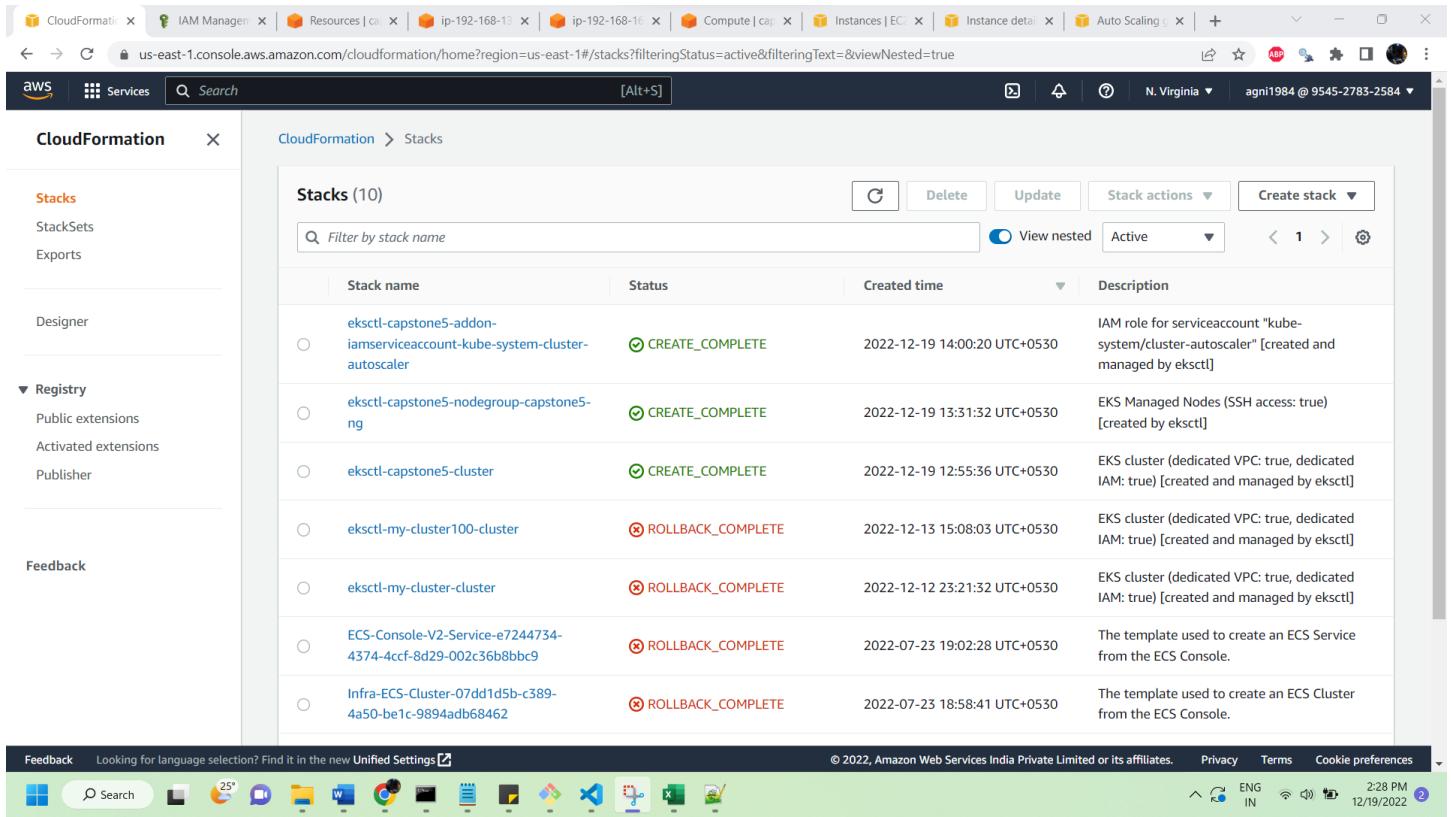
5.2. Deploy the Cluster Autoscaler

Complete the following steps to deploy the Cluster Autoscaler.

- Download the Cluster Autoscaler YAML file.

```
kubectl apply -f https://raw.githubusercontent.com/kubernetes/autoscaler/master/cluster-
autoscaler/cloudprovider/aws/examples/cluster-autoscaler-autodiscover.yaml
```

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\redis-slave>kubectl apply -f https://raw.githubusercontent.com/kubernetes/autoscaler/master/cluster-autoscaler/cloudprovi
der/aws/examples/cluster-autoscaler-autodiscover.yaml
Warning: resource serviceaccounts/cluster-autoscaler is missing the kubectl.kubernetes.io/last-applied-configuration annotation which is required by kubectl apply. kubectl apply should only be used on resources
created declaratively by either kubectl create --save-config or kubectl apply. The missing annotation will be patched automatically.
serviceaccount/cluster-autoscaler configured
clusterrole.rbac.authorization.k8s.io/cluster-autoscaler created
role.rbac.authorization.k8s.io/cluster-autoscaler created
clusterrolebinding.rbac.authorization.k8s.io/cluster-autoscaler created
rolebinding.rbac.authorization.k8s.io/cluster-autoscaler created
deployment.apps/cluster-autoscaler created
```



Stacks (10)

Stack name	Status	Created time	Description
eksctl-capstone5-addon-iamserviceaccount-kube-system-cluster-autoscaler	CREATE_COMPLETE	2022-12-19 14:00:20 UTC+0530	IAM role for serviceaccount "kube-system/cluster-autoscaler" [created and managed by eksctl]
eksctl-capstone5-nodegroup-capstone5-ing	CREATE_COMPLETE	2022-12-19 13:31:32 UTC+0530	EKS Managed Nodes (SSH access: true) [created by eksctl]
eksctl-capstone5-cluster	CREATE_COMPLETE	2022-12-19 12:55:36 UTC+0530	EKS cluster (dedicated VPC: true, dedicated IAM: true) [created and managed by eksctl]
eksctl-my-cluster100-cluster	ROLLBACK_COMPLETE	2022-12-13 15:08:03 UTC+0530	EKS cluster (dedicated VPC: true, dedicated IAM: true) [created and managed by eksctl]
eksctl-my-cluster-cluster	ROLLBACK_COMPLETE	2022-12-12 23:21:32 UTC+0530	EKS cluster (dedicated VPC: true, dedicated IAM: true) [created and managed by eksctl]
ECS-Console-V2-Service-e7244734-4374-4ccf-8d29-002c36b8bbc9	ROLLBACK_COMPLETE	2022-07-23 19:02:28 UTC+0530	The template used to create an ECS Service from the ECS Console.
Infra-ECS-Cluster-07dd1d5b-c389-4a50-be1c-9894adb68462	ROLLBACK_COMPLETE	2022-07-23 18:58:41 UTC+0530	The template used to create an ECS Cluster from the ECS Console.

- Modify the cluster name in autoscaler, by replacing <YOUR CLUSTER NAME> with capstone5

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\redis-slave>kubectl -n kube-system edit deployment.apps/cluster-autoscaler
deployment.apps/cluster-autoscaler edited
```

- Annotate the cluster-autoscaler service account with the ARN of the IAM role that we created previously.

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\redis-slave>kubectl annotate serviceaccount cluster-autoscaler -n kube-system eks.amazonaws.com/role-arn=arn:aws:iam::27832584:role/AmazonEKSClusterAutoscalerRole
error: --overwrite is false but found the following declared annotation(s): 'eks.amazonaws.com/role-arn' already has a value (arn:aws:iam::954527832584:role/eksctl-capstone5-addon-iamserviceaccount-kub-Role1-FBB0RPX8B176)
```

- Patch the deployment to add the cluster-autoscaler.kubernetes.io/safe-to-evict annotation to the Cluster Autoscaler pods with the following command.

```
kubectl patch deployment cluster-autoscaler -n kube-system -p
'{"spec": {"template": {"metadata": {"annotations": {"cluster-autoscaler.kubernetes.io/safe-to-evict": "false"} }}}}'
```

```
agni1@LAPTOP-BSMIH6S8 MINGW64 ~
$ kubectl patch deployment cluster-autoscaler -n kube-system -p '{"spec":{"template":{"metadata":{"annotations":{"cluster-autoscaler.kubernetes.io/safe-to-evict": "false"}}}}}' --type=JSONPath
deployment.apps/cluster-autoscaler patched
```

- Edit the Cluster Autoscaler deployment with the following command.

```
kubectl -n kube-system edit deployment.apps/cluster-autoscaler
```

Edit the cluster-autoscaler container command to add the following options. --balance-similar-node-groups ensures that there is enough available compute across all availability zones. --skip-nodes-with-system-pods=false ensures that there are no problems with scaling to zero.

--balance-similar-node-groups
--skip-nodes-with-system-pods=false

```
spec:
  containers:
    - command:
        - ./cluster-autoscaler
        - --v=4
        - --stderrthreshold=info
        - --cloud-provider=aws
        - --skip-nodes-with-local-storage=false
        - --expander=least-waste
        - --node-group-auto-discovery=asg:tag=k8s.io/cluster-autoscaler/enabled,k8s.io/cluster-autoscaler/capstone5
          - --balance-similar-node-groups
          - --skip-nodes-with-system-pods=false
```

- Save and close the file to apply the changes.

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\redis-slave>
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\redis-slave>kubectl -n kube-system edit deployment.apps/cluster-autoscaler
deployment.apps/cluster-autoscaler edited
```

- Set the Cluster Autoscaler image tag to the version 1.25.0

```
kubectl set image deployment cluster-autoscaler -n kube-system cluster-autoscaler=k8s.gcr.io/autoscaling/cluster-autoscaler:v1.25.0
```

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\redis-slave>kubectl set image deployment cluster-autoscaler -n kube-system cluster-autoscaler=k8s.gcr.io/autoscaling/cluster-autoscaler:v1.25.0
deployment.apps/cluster-autoscaler image updated
```

5.3. View Cluster Autoscaler logs

After we have deployed the Cluster Autoscaler, we can view the logs and verify that it's monitoring our cluster load.

View Cluster Autoscaler logs with the following command.

```
kubectl -n kube-system logs -f deployment.apps/cluster-autoscaler
```

The example output is as follows.

```
ca| Administrator: Command Prompt - kubectl -n kube-system logs -f deployment.apps/cluster-autoscaler
I1219 09:33:23.009164 1 pre_filtering_processor.go:66] Skipping ip-192-168-164-85.ec2.internal - node group min size reached
I1219 09:33:23.009170 1 pre_filtering_processor.go:66] Skipping ip-192-168-135-51.ec2.internal - node group min size reached
I1219 09:33:23.009202 1 static_autoscaler.go:35] Scale down status: lastScaleUpTime=2022-12-19 08:31:42.812891855 +0000 UTC m=-3581.007598434 lastScaleDownDeleteTime=2022-12-19 08:31:42.812891855 +0000 UTC m=-3581.007598434 scaleDownForbidden=false scaleDownInCooldown=false
I1219 09:33:23.009228 1 static_autoscaler.go:54] Starting scale down
I1219 09:33:23.009235 1 legacy.go:727] No candidates for scale down
I1219 09:33:23.009246 1 static_autoscaler.go:243] Starting main loop
I1219 09:33:23.072872 1 filter_out_schedulable.go:65] Filtering out schedulables
I1219 09:33:23.072893 1 filter_out_schedulable.go:137] Filtered out 0 pods using hints
I1219 09:33:23.072898 1 filter_out_schedulable.go:176] 0 pods were kept as unschedulable based on caching
I1219 09:33:23.072902 1 filter_out_schedulable.go:177] 0 pods marked as unschedulable can be scheduled.
I1219 09:33:23.072908 1 filter_out_schedulable.go:87] No schedulable pods
I1219 09:33:23.072923 1 static_autoscaler.go:445] No unschedulable pods
I1219 09:33:23.072934 1 static_autoscaler.go:492] Calculating unneeded nodes
I1219 09:33:23.072942 1 pre_filtering_processor.go:66] Skipping ip-192-168-164-85.ec2.internal - node group min size reached
I1219 09:33:23.072949 1 pre_filtering_processor.go:66] Skipping ip-192-168-135-51.ec2.internal - node group min size reached
I1219 09:33:23.072959 1 static_autoscaler.go:35] Scale down status: lastScaleUpTime=2022-12-19 08:31:42.812891855 +0000 UTC m=-3581.007598434 lastScaleDownDeleteTime=2022-12-19 08:31:42.812891855 +0000 UTC m=-3581.007598434 scaleDownForbidden=false scaleDownInCooldown=false
I1219 09:33:23.073005 1 static_autoscaler.go:54] Starting scale down
I1219 09:33:23.073027 1 legacy.go:727] No candidates for scale down
I1219 09:33:23.073045 1 reflector.go:257] Listing and Watching *v1.CSISStorageCapacity from k8s.io/client-go/informers/factory.go:134
I1219 09:33:23.073048 1 reflector.go:424] k8s.io/client-go/informers/factory.go:134: failed to list *v1.CSISStorageCapacity: the server could not find the requested resource
I1219 09:33:23.073049 1 reflector.go:140] k8s.io/client-go/informers/factory.go:134: Failed to watch *v1.CSISStorageCapacity: failed to list *v1.CSISStorageCapacity: the server could not find the requested resource
I1219 09:33:42.813830 1 node_instances_cache.go:156] Start refreshing cloud provider node instances cache
I1219 09:33:42.813857 1 node_instances_cache.go:168] Refresh cloud provider node instances cache finished, refresh took 10.821µs
I1219 09:33:43.008474 1 static_autoscaler.go:243] Starting main loop
I1219 09:33:43.0084811 1 auto_scaling_groups.go:360] Regenerating instance to ASG map for ASG names: []
I1219 09:33:43.0084865 1 auto_scaling_groups.go:367] Regenerating instance to ASG map for ASG tags: map[k8s.io/cluster-autoscaler/capstone5: k8s.io/cluster-autoscaler/enabled:]
I1219 09:33:43.008665 1 aws_wrapper.go:689] 0 launch configurations to query
I1219 09:33:43.200885 1 aws_wrapper.go:690] 0 launch templates to query
I1219 09:33:43.200992 1 aws_wrapper.go:706] Successfully queried 0 launch configurations
I1219 09:33:43.200997 1 aws_wrapper.go:717] Successfully queried 0 launch templates
I1219 09:33:43.2009983 1 aws_wrapper.go:728] Successfully queried instance requirements for 0 ASGs
I1219 09:33:43.2009993 1 aws_manager.go:266] Refreshed ASG list, next refresh after 2022-12-19 09:34:43.200991053 +0000 UTC m+=199.380420241
I1219 09:33:43.201242 1 filter_out_schedulable.go:65] Calculating unneeded nodes
I1219 09:33:43.201259 1 filter_out_schedulable.go:137] Filtered out 0 pods using hints
I1219 09:33:43.201266 1 filter_out_schedulable.go:176] 0 pods were kept as unschedulable based on caching
I1219 09:33:43.201272 1 filter_out_schedulable.go:177] 0 pods marked as unschedulable can be scheduled.
I1219 09:33:43.201288 1 filter_out_schedulable.go:87] No schedulable pods
I1219 09:33:43.201302 1 static_autoscaler.go:445] No unschedulable pods
I1219 09:33:43.201325 1 static_autoscaler.go:492] Calculating unneeded nodes
I1219 09:33:43.201339 1 pre_filtering_processor.go:66] Skipping ip-192-168-164-85.ec2.internal - node group min size reached
I1219 09:33:43.201353 1 pre_filtering_processor.go:66] Skipping ip-192-168-135-51.ec2.internal - node group min size reached
I1219 09:33:43.201388 1 static_autoscaler.go:35] Scale down status: lastScaleUpTime=2022-12-19 08:31:42.812891855 +0000 UTC m=-3581.007598434 lastScaleDownDeleteTime=2022-12-19 08:31:42.812891855 +0000 UTC m=-3581.007598434 scaleDownForbidden=false scaleDownInCooldown=false
I1219 09:33:43.201422 1 static_autoscaler.go:54] Starting scale down
I1219 09:33:43.201460 1 legacy.go:727] No candidates for scale down
```

6. Deploy Application

In this module, we build and deploy a multi-tier web application using Kubernetes and Docker. It consists of the following components:

- A single-instance Redis to store guestbook entries
- Multiple web frontend instances

6.1. Start up the Redis Database

The guestbook application uses Redis to store its data.

6.1.1. Creating the Redis Deployment

The manifest file, included below, specifies a Deployment controller that runs a single replica Redis Pod.

```
# SOURCE: ... \redis-Leader-deployment.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: redis-leader
  labels:
    app: redis
    role: leader
    tier: backend
spec:
  replicas: 1
  selector:
    matchLabels:
      app: redis
  template:
    metadata:
      labels:
        app: redis
        role: leader
        tier: backend
    spec:
      containers:
        - name: leader
          image: "docker.io/redis:6.0.5"
          resources:
            requests:
              cpu: 100m
              memory: 100Mi
          ports:
            - containerPort: 6379
```

- Launch a terminal window in the directory of downloaded the manifest files.
- Apply the Redis Deployment from the redis-leader-deployment.yaml file:

```
kubectl apply -f redis-leader-deployment.yaml
```

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\redis-slave\capstone5>kubectl apply -f redis-leader-deployment.yaml
deployment.apps/redis-leader created
```

- Query the list of Pods to verify that the Redis Pod is running:

```
kubectl get pods
```

The response should be similar to this:

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>kubectl get pods
NAME           READY   STATUS    RESTARTS   AGE
redis-leader-766465cd9c-5nd9r   1/1     Running   0          7s
```

- Run the following command to view the logs from the Redis leader Pod:

```
kubectl logs -f deployment/redis-leader
```

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>kubectl logs -f deployment/redis-leader
1:C 19 Dec 2022 09:34:40.102 # o000o000o000 Redis is starting o000o000o000
1:C 19 Dec 2022 09:34:40.102 # Redis version=6.0.5, bits=64, commit=00000000, modified=0, pid=1, just started
1:C 19 Dec 2022 09:34:40.102 # Warning: no config file specified, using the default config. In order to specify a config file use redis-server /path/to/redis.conf
1:M 19 Dec 2022 09:34:40.103 * Running mode=standalone, port=6379.
1:M 19 Dec 2022 09:34:40.104 # Server initialized
1:M 19 Dec 2022 09:34:40.104 # WARNING you have Transparent Huge Pages (THP) support enabled in your kernel. This will create latency and memory usage issues with Redis. To fix this issue run the command 'echo never > /sys/kernel/mm/transparent_hugepage/enabled' as root, and add it to your /etc/rc.local in order to retain the setting after a reboot. Redis must be restarted after THP is disabled.
1:M 19 Dec 2022 09:34:40.104 * Ready to accept connections
```

6.1.2. Creating the Redis leader Service

The guestbook application needs to communicate to the Redis to write its data. We need to apply a Service to proxy the traffic to the Redis Pod. A Service defines a policy to access the Pods.

```
# SOURCE: .../redis-leader-service.yaml
apiVersion: v1
kind: Service
metadata:
  name: redis-leader
  labels:
    app: redis
    role: leader
    tier: backend
spec:
  ports:
  - port: 6379
    targetPort: 6379
  selector:
    app: redis
    role: leader
    tier: backend
```

- Apply the Redis Service from the following redis-leader-service.yaml file:

```
kubectl apply -f redis-leader-service.yaml
```

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\redis-slave\capstone5>kubectl apply -f redis-leader-service.yaml
service/redis-leader created
```

- Query the list of Services to verify that the Redis Service is running:

```
kubectl get service
```

The response should be similar to this:

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>kubectl get service
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
kubernetes   ClusterIP   10.100.0.1    <none>        443/TCP     122m
redis-leader   ClusterIP   10.100.78.253  <none>        6379/TCP    8s
```

Note: This manifest file creates a Service named **redis-leader** with a set of labels that match the labels previously defined, so the Service routes network traffic to the Redis Pod.

6.1.3. Set up Redis followers

Although the Redis leader is a single Pod, we can make it highly available and meet traffic demands by adding a few Redis followers, or replicas.

```
# SOURCE: .../redis-follower-deployment.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: redis-follower
  labels:
    app: redis
    role: follower
    tier: backend
spec:
  replicas: 2
  selector:
    matchLabels:
      app: redis
  template:
    metadata:
      labels:
        app: redis
        role: follower
        tier: backend
    spec:
      containers:
        - name: follower
          image: gcr.io/google_samples/gb-redis-follower:v2
          resources:
            requests:
              cpu: 100m
              memory: 100Mi
          ports:
            - containerPort: 6379
```

- Apply the Redis Deployment from the following redis-follower-deployment.yaml file:

```
kubectl apply -f redis-follower-deployment.yaml
```

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>kubectl apply -f redis-follower-deployment.apps/redis-follower created
```

- Verify that the two Redis follower replicas are running by querying the list of Pods:

```
kubectl get pods
```

The response should be similar to this:

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>kubectl get pods
NAME           READY   STATUS    RESTARTS   AGE
redis-follower-84fcc94dfc-6zz5s   1/1     Running   0          15s
redis-follower-84fcc94dfc-z19f5   1/1     Running   0          15s
redis-leader-766465cd9c-5nd9r    1/1     Running   0          2m14s
```

6.1.4. Creating the Redis follower service

The guestbook application needs to communicate with the Redis followers to read data. To make the Redis followers discoverable, we must set up another Service.

```
# SOURCE: .../redis-follower-service.yaml
apiVersion: v1
kind: Service
metadata:
  name: redis-follower
  labels:
    app: redis
    role: follower
    tier: backend
spec:
  ports:
    # the port that this service should serve on
    - port: 6379
  selector:
    app: redis
    role: follower
    tier: backend
```

- Apply the Redis Service from the following redis-follower-service.yaml file:

```
kubectl apply -f redis-follower-service.yaml
```

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>kubectl apply -f redis-follower-service.yaml
service/redis-follower created
```

- Query the list of Services to verify that the Redis Service is running:

```
kubectl get service
```

The response should be similar to this:

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>kubectl get service
NAME         TYPE      CLUSTER-IP    EXTERNAL-IP   PORT(S)      AGE
kubernetes   ClusterIP 10.100.0.1    <none>        443/TCP     124m
redis-follower ClusterIP 10.100.208.247 <none>        6379/TCP    5s
redis-leader  ClusterIP 10.100.78.253  <none>        6379/TCP    100s
```

Note: This manifest file creates a Service named **redis-follower** with a set of labels that match the labels previously defined, so the Service routes network traffic to the Redis Pod.

6.2. Set up and Expose the Guestbook Frontend

Now Redis storage of our guestbook up and running, start the guestbook web servers. Like the Redis followers, the frontend is also deployed using a Kubernetes Deployment.

The guestbook app uses a PHP frontend. It is configured to communicate with either the Redis follower or leader Services, depending on whether the request is a read or a write. The frontend exposes a JSON interface, and serves a jQuery-Ajax-based UX.

6.2.1. Creating the Guestbook Frontend Deployment

```
# SOURCE: .../frontend-deployment.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: frontend
spec:
  replicas: 3
  selector:
    matchLabels:
      app: guestbook
      tier: frontend
  template:
    metadata:
      labels:
        app: guestbook
```

```

tier: frontend
spec:
  containers:
    - name: php-redis
      image: gcr.io/google_samples/gb-frontend:v5
      env:
        - name: GET_HOSTS_FROM
          value: "dns"
      resources:
        requests:
          cpu: 100m
          memory: 100Mi
      ports:
        - containerPort: 80
  
```

- Apply the frontend Deployment from the frontend-deployment.yaml file:

```
kubectl apply -f frontend-deployment.yaml
```

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>kubectl apply -f frontend-deployment.yaml
deployment.apps/frontend created
```

- Query the list of Pods to verify that the three frontend replicas are running:

```
kubectl get pods -l app=guestbook -l tier=frontend
```

The response should be similar to this:

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>kubectl get pods -l app=guestbook -l tier=frontend
NAME           READY   STATUS    RESTARTS   AGE
frontend-57df59b89c-4lp74  1/1     Running   0          25s
frontend-57df59b89c-kwpcj  1/1     Running   0          25s
frontend-57df59b89c-mjkbt  1/1     Running   0          25s
```

6.2.2. Creating the Frontend Service

The Redis Services we applied is only accessible within the Kubernetes cluster because the default type for a Service is ClusterIP. ClusterIP provides a single IP address for the set of Pods the Service is pointing to. This IP address is accessible only within the cluster.

If we want guests to be able to access our guestbook, we must configure the frontend Service to be externally visible, so a client can request the Service from outside the Kubernetes cluster. As AWS supports external load balancers, use LoadBalancer as type. It helps to create AWS application load balancer through which we can access ‘guestbook’ application

```
# SOURCE: .../frontend-service.yaml
```

```

apiVersion: v1
kind: Service
metadata:
  name: frontend
  labels:
    app: guestbook
    tier: frontend
spec:
  # if your cluster supports it, uncomment the following to automatically create
  # an external Load-balanced IP for the frontend service.
  # type: LoadBalancer
  type: LoadBalancer
  ports:
    # the port that this service should serve on
    - port: 80
  selector:
    app: guestbook
    tier: frontend

```

- Apply the frontend Service from the frontend-service.yaml file:

```
kubectl apply -f frontend-service.yaml
```

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>kubectl apply -f frontend-service.yaml
service/frontend created
```

- Query the list of Services to verify that the frontend Service is running:

```
kubectl get services
```

The response should be similar to this:

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>kubectl get services
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
frontend   LoadBalancer  10.100.248.229  aeec240d803a84e7fb7f95a4a920f6e0-391405583.us-east-1.elb.amazonaws.com  80:30147/TCP  28s
kubernetes ClusterIP  10.100.0.1      <none>          443/TCP     128m
redis-follower ClusterIP  10.100.208.247  <none>          6379/TCP    3m41s
redis-leader   ClusterIP  10.100.78.253   <none>          6379/TCP    5m16s
```

AWS Load balancer is created and shown in AWS console. In listener section, we can see loadbalancer's port 80 route the traffic to container port 30147

CloudForm x | IAM Manag x | IAM Manag x | Resources x | ip-192-168-1-100.x | ip-192-168-1-100.x | Compute x | Load balanc x | Instance de x | Auto Scal x | + | - | X

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LoadBalancers:

aws Services Search [Alt+S]

EC2 > Load balancers

Load balancers (1)
Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.

Name	DNS name	State	VPC ID	Availability Zones	Type	Create
aeec240d803a84e7fb7f95a4a920f6e0	aeec240d803a84e7fb7f95a4a920f6e0-391405583.us-east-1.elb.amazonaws.com	-	vpc-03a808bf56722235b	3 Availability Zones	classic	Decem 15:10

Feedback Looking for language selection? Find it in the new Unified Settings 

© 2022, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

ENG IN 3:12 PM 12/19/2022

CloudForm x | IAM Manag x | IAM Manag x | Resources x | ip-192-168-1-100.x | ip-192-168-1-100.x | Compute x | Load balanc x | Details Po x | Instance de x | Auto Scal x | + | - | X

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LBDetails:clbName=aeec240d803a84e7fb7f95a4a920f6e0

aws Services Search [Alt+S]

EC2 | Load balancers | aeec240d803a84e7fb7f95a4a920f6e0

Load balancer: aeec240d803a84e7fb7f95a4a920f6e0

Description Instances Health check Listeners Monitoring Tags Migration

Basic Configuration

Name	aeec240d803a84e7fb7f95a4a920f6e0	Creation time	December 19, 2022 at 3:10:41 PM UTC+5:30
* DNS name	aeec240d803a84e7fb7f95a4a920f6e0-391405583.us-east-1.elb.amazonaws.com (A Record)	Hosted zone	Z35SXDOTRQ7X7K
Type	Classic (Migrate Now)	Status	2 of 2 instances in service
Scheme	internet-facing	VPC	vpc-03a808bf56722235b
Availability Zones	subnet-087dc0e13293a0220 - us-east-1c, subnet-0b6550182985a7e21 - us-east-1b, subnet-0cb8571d55773bebe - us-east-1a		

Port Configuration

Port Configuration	80 (TCP) forwarding to 30147 (TCP)
Info Slickness options not available for TCP protocols	

Security

Source Security Group	sg-0d02448ae4e4da798, k8s-elb-aeec240d803a84e7fb7f95a4a920f6e0
• Security group for Kubernetes ELB aeec240d803a84e7fb7f95a4a920f6e0 (default/frontend)	
Edit security groups	

Feedback Looking for language selection? Find it in the new Unified Settings 

© 2022, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

ENG IN 3:12 PM 12/19/2022



Screenshot of the AWS CloudFormation console showing the creation of a new stack named 'CloudFormation'. The 'Outputs' tab is selected, displaying the output key 'BucketName' with the value 'arn:aws:s3:::cloudformation-eu-central-1-123456789012'. Other tabs include 'Template', 'Parameters', 'Resources', and 'Logs'.

Screenshot of the AWS EC2 Load Balancers console showing the configuration of a load balancer named 'aeec240d803a84e7fb7f95a4a920f6e0'. The 'Listeners' tab is selected, showing a single listener configuration:

Load Balancer Protocol	Load Balancer Port	Instance Protocol	Instance Port	Cipher	SSL Certificate
TCP	80	TCP	30147	N/A	N/A

Screenshot of a Microsoft Edge browser window showing a guestbook application. The URL is 'Not secure | aeec240d803a84e7fb7f95a4a920f6e0-391405583.us-east-1.elb.amazonaws.com'. The page displays a form with a text input field containing 'Hello, Rajeswari' and a 'Submit' button.

Screenshot of a Microsoft Edge browser window showing a guestbook application. The URL is 'Not secure | aeec240d803a84e7fb7f95a4a920f6e0-391405583.us-east-1.elb.amazonaws.com'. The page displays a form with a text input field containing 'Hello, Rajeswari' and a 'Submit' button.

Guestbook x +

Not secure | aeec240d803a84e7fb7f95a4a920f6e0-391405583.us-east-1.elb.amazonaws.com

Guestbook

Messages

Submit

Hello
Rajeswari



Guestbook x +

Not secure | aeec240d803a84e7fb7f95a4a920f6e0-391405583.us-east-1.elb.amazonaws.com

Guestbook

Hi, Guest

Submit

Hello
Rajeswari





Guestbook

Messages

Submit

Hello
Rajeswari
Hi, Guest



Guestbook

Messages

Submit

Hello
Rajeswari
Hi
Gues



7. Auto Scaling

7.1. Scale the Pods

We can scale up or down as needed because servers are defined as a Service that uses a Deployment controller.

- Run the following command to scale up the number of frontend Pods:

```
kubectl scale deployment frontend --replicas=5
```

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>kubectl scale deployment frontend --replicas=5
deployment.apps/frontend scaled
```

- Query the list of Pods to verify the number of frontend Pods running:

```
kubectl get pods
```

The response should look similar to this:

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
frontend-57df59b89c-4lp74   1/1     Running   0          6m7s
frontend-57df59b89c-9lntw   1/1     Running   0          26s
frontend-57df59b89c-kwpcj   1/1     Running   0          6m7s
frontend-57df59b89c-mjkbt   1/1     Running   0          6m7s
frontend-57df59b89c-rjclw   1/1     Running   0          26s
redis-follower-84fcc94dfc-6zz5s 1/1     Running   0          7m46s
redis-follower-84fcc94dfc-zl9f5 1/1     Running   0          7m46s
redis-leader-766465cd9c-5nd9r  1/1     Running   0          9m45s
```

To view, cpu and memory & pod allocation details of each node, follow below steps,

1. Go to EKS in AWS console and open ‘capston5’ cluster
2. Got ‘Compute’ tab and select node

Screenshot of the AWS CloudFormation console showing the CloudFormation stack named 'capstone5' with a status of 'CREATE_COMPLETE'. The stack contains resources like 'AWS::Lambda::Function', 'AWS::SQS::Queue', and 'AWS::DynamoDB::Table'.

Cluster info

Kubernetes version	Info 1.23	Status	Active	Provider	EKS
--------------------	--------------	--------	--------	----------	-----

Compute tab selected.

Nodes (2)

Node name	Instance type	Node group	Created	Status
ip-192-168-135-51.ec2.internal	t3.medium	capstone5-ng	Created 2 hours ago	Ready
ip-192-168-164-85.ec2.internal	t3.medium	capstone5-ng	Created 2 hours ago	Ready

Node groups (1)

Group name	Desired size	AMI release version	Launch template	Status
capstone5-ng	2	1.23.13-20221112	eksctl-capstone5-nodegroup-capstone5-ng (1)	Active

Feedback Looking for language selection? Find it in the new Unified Settings 

© 2022, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

CloudFormation | IAM Management | IAM Management | Compute | ip-192-168-135-51 | ip-192-168-135-51 | Compute | Load balancer | Details | Instances | Auto Scaling | + | N. Virginia | agni1984 @ 9545-2783-2584 | 3:16 PM 12/19/2022

Screenshot of the AWS CloudFormation console showing the CloudFormation stack named 'capstone5' with a status of 'CREATE_COMPLETE'. The stack contains resources like 'AWS::Lambda::Function', 'AWS::SQS::Queue', and 'AWS::DynamoDB::Table'.

Amazon Elastic Kubernetes Service

Clusters

Related services

- Amazon ECR
- AWS Batch

Documentation

Feedback Looking for language selection? Find it in the new Unified Settings 

© 2022, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

CloudFormation | IAM Management | IAM Management | Compute | ip-192-168-135-51 | ip-192-168-135-51 | Compute | Load balancer | Details | Instances | Auto Scaling | + | N. Virginia | agni1984 @ 9545-2783-2584 | 3:16 PM 12/19/2022

Amazon Elastic Kubernetes Service

Clusters

Related services

- Amazon ECR
- AWS Batch

Documentation

Feedback Looking for language selection? Find it in the new Unified Settings 

© 2022, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

CloudFormation | IAM Management | IAM Management | Compute | ip-192-168-135-51 | ip-192-168-135-51 | Compute | Load balancer | Details | Instances | Auto Scaling | + | N. Virginia | agni1984 @ 9545-2783-2584 | 3:17 PM 12/19/2022

Amazon Elastic Kubernetes Service

Clusters

Related services

- Amazon ECR
- AWS Batch

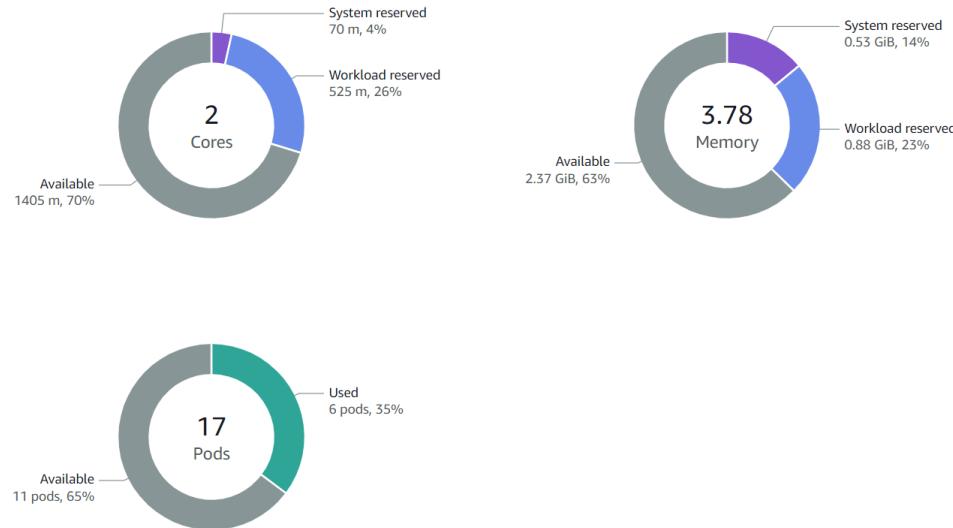
Documentation

Feedback Looking for language selection? Find it in the new Unified Settings 

© 2022, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

CloudFormation | IAM Management | IAM Management | Compute | ip-192-168-135-51 | ip-192-168-135-51 | Compute | Load balancer | Details | Instances | Auto Scaling | + | N. Virginia | agni1984 @ 9545-2783-2584 | 3:17 PM 12/19/2022

Core Metrics



Core Metrics

2 Cores

- Available: 1405 m, 70%
- System reserved: 70 m, 4%
- Workload reserved: 525 m, 26%

3.78 Memory

- Available: 2.37 GiB, 63%
- System reserved: 0.53 GiB, 14%
- Workload reserved: 0.88 GiB, 23%

17 Pods

- Used: 6 pods, 35%
- Available: 11 pods, 65%



The screenshot shows the AWS EKS (Amazon Elastic Kubernetes Service) console. The left sidebar includes links for Clusters, Related services (Amazon ECR, AWS Batch), Documentation, and Submit feedback. The main content area displays the 'Pods (6)' section, which lists the following pods:

Name	Status	Created	IP
frontend-57df59b89c-4lp74	Running	9 minutes ago	192.168.137.218
frontend-57df59b89c-mjkbt	Running	9 minutes ago	192.168.141.62
redis-follower-84fcc94dfc-6zz5s	Running	10 minutes ago	192.168.137.18
aws-node-m62bs	Running	2 hours ago	192.168.135.51
cluster-autoscaler-5c6d5bdd74-bgj97	Running	16 minutes ago	192.168.150.33
kube-proxy-f6w9r	Running	2 hours ago	192.168.135.51

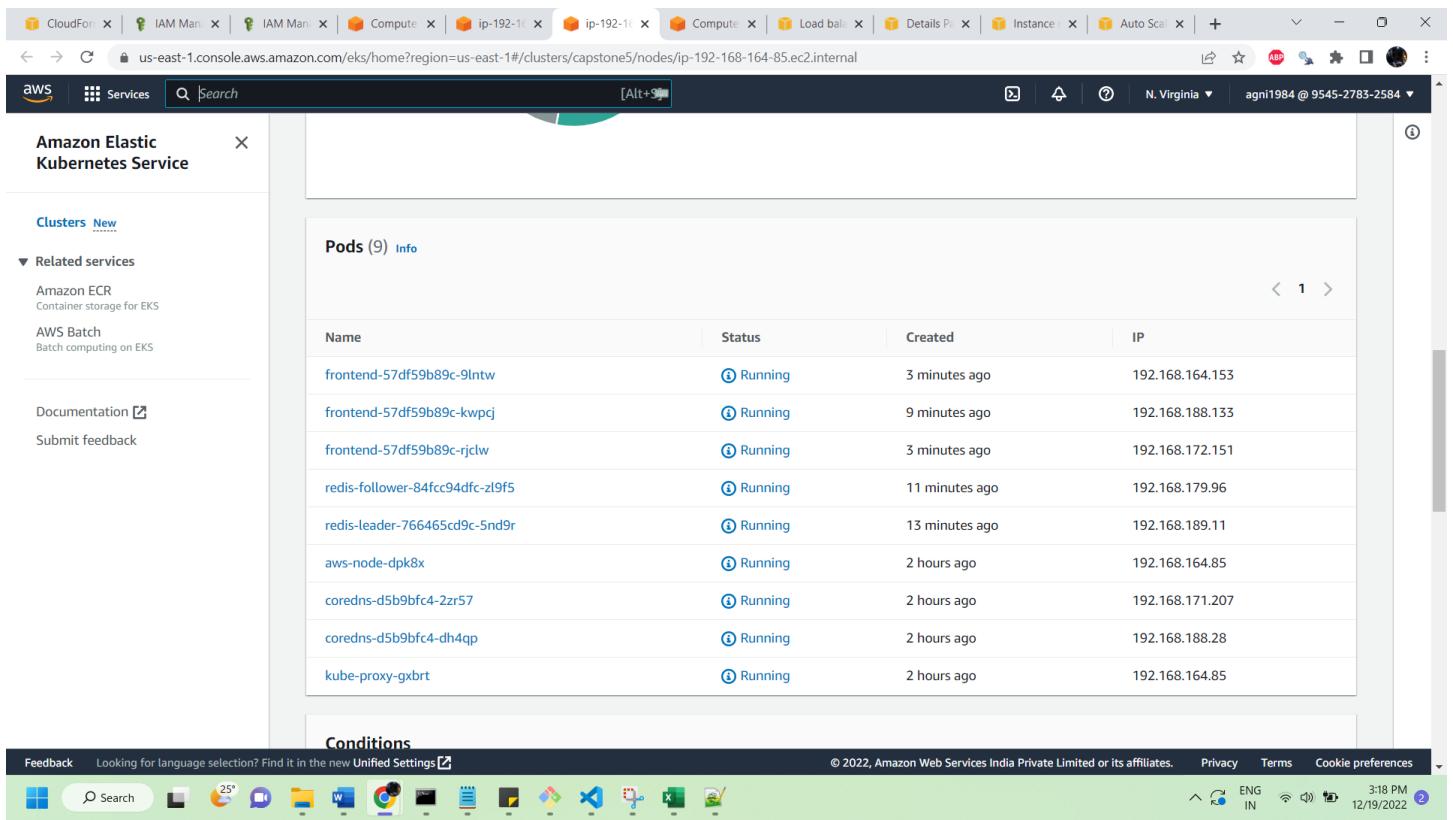
Below the pods table is a 'Conditions' section, which shows a single entry:

Name	Status	Message
MemoryPressure	False	kubelet has sufficient memory available

The screenshot shows the Amazon Elastic Kubernetes Service (EKS) home page. On the left, a sidebar displays navigation links for CloudFront, IAM Manager, Compute, ip-192-168-164-85.ec2.internal, Load balancer, Details Page, Instance, Auto Scale, Services, and a search bar. The main content area features three donut charts:

- CPU Utilization:** Shows 2 Cores. Data: Available 1105 m, 55%; Workload reserved 825 m, 41%; System reserved 70 m, 4%.
- Memory Utilization:** Shows 3.78 Memory. Data: Available 2.63 GiB, 69%; Workload reserved 0.63 GiB, 17%; System reserved 0.53 GiB, 14%.
- Pods Utilization:** Shows 17 Pods. Data: Available 8 pods, 47%; Used 9 pods, 53%.

On the right side, there is a user profile for agni1984 @ 9545-2783-2584, indicating they are in N. Virginia. The bottom of the screen includes a toolbar with icons for Feedback, Search, and various system status indicators.



Clusters [New](#)

Related services

- Amazon ECR Container storage for EKS
- AWS Batch Batch computing on EKS

Documentation [Submit feedback](#)

Name	Status	Created	IP
frontend-57df59b89c-9ntw	Running	3 minutes ago	192.168.164.153
frontend-57df59b89c-kwpcj	Running	9 minutes ago	192.168.188.133
frontend-57df59b89c-rjclw	Running	3 minutes ago	192.168.172.151
redis-follower-84fcc94dfc-zl9f5	Running	11 minutes ago	192.168.179.96
redis-leader-766465cd9c-5nd9r	Running	13 minutes ago	192.168.189.11
aws-node-dpk8x	Running	2 hours ago	192.168.164.85
coredns-d5b9bfc4-2zr57	Running	2 hours ago	192.168.171.207
coredns-d5b9bfc4-dh4qp	Running	2 hours ago	192.168.188.28
kube-proxy-gxbrt	Running	2 hours ago	192.168.164.85

Conditions

Feedback Looking for language selection? Find it in the new Unified Settings [?](#)

© 2022, Amazon Web Services India Private Limited or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

ENG IN 3:18 PM 12/19/2022

- Run the following command to scale down the number of frontend Pods:

```
kubectl scale deployment frontend --replicas=2
```

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>kubectl scale deployment frontend --replicas=2
deployment.apps/frontend scaled
```

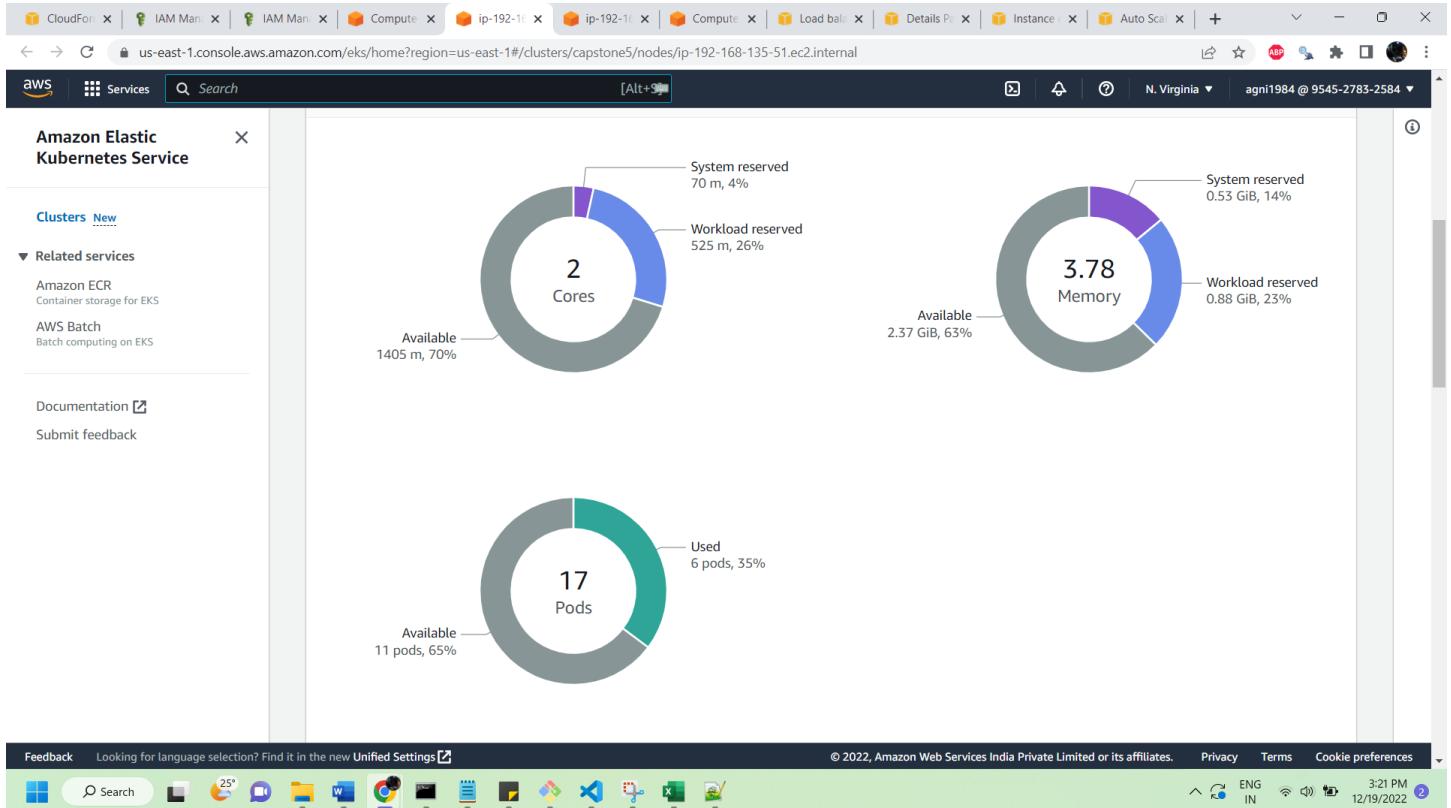
- Query the list of Pods to verify the number of frontend Pods running:

```
kubectl get pods
```

The response should look similar to this:

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
frontend-57df59b89c-4lp74  1/1     Running   0          11m
frontend-57df59b89c-mjkbt  1/1     Running   0          11m
redis-follower-84fcc94dfc-6zz5s  1/1     Running   0          12m
redis-follower-84fcc94dfc-zl9f5  1/1     Running   0          12m
redis-leader-766465cd9c-5nd9r   1/1     Running   0          14m
```

Front end pos would be reduced to 2 and can see the details in AWS console.



Feedback Looking for language selection? Find it in the new Unified Settings

© 2022, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

CloudFront IAM Manager IAM Manager Compute ip-192-168-135-51.ec2.internal ip-192-168-135-51.ec2.internal Compute Load bal. Details Page Instances Auto Scaling +

aws Services Search [Alt+Shift]

N. Virginia agni1984 @ 9545-2783-2584

Amazon Elastic Kubernetes Service

Clusters New

Related services

- Amazon ECR Container storage for EKS
- AWS Batch Batch computing on EKS

Documentation **Submit feedback**

Pods (6)

Name	Status	Created	IP
frontend-57df59b89c-4lp74	Running	13 minutes ago	192.168.137.218
frontend-57df59b89c-mjkbt	Running	13 minutes ago	192.168.141.62
redis-follower-84fcc94dfc-6zz5s	Running	14 minutes ago	192.168.137.18
aws-node-m62bs	Running	2 hours ago	192.168.135.51
cluster-autoscaler-5c6d5bdd74-bgj97	Running	20 minutes ago	192.168.150.33
kube-proxy-f6w9r	Running	2 hours ago	192.168.135.51

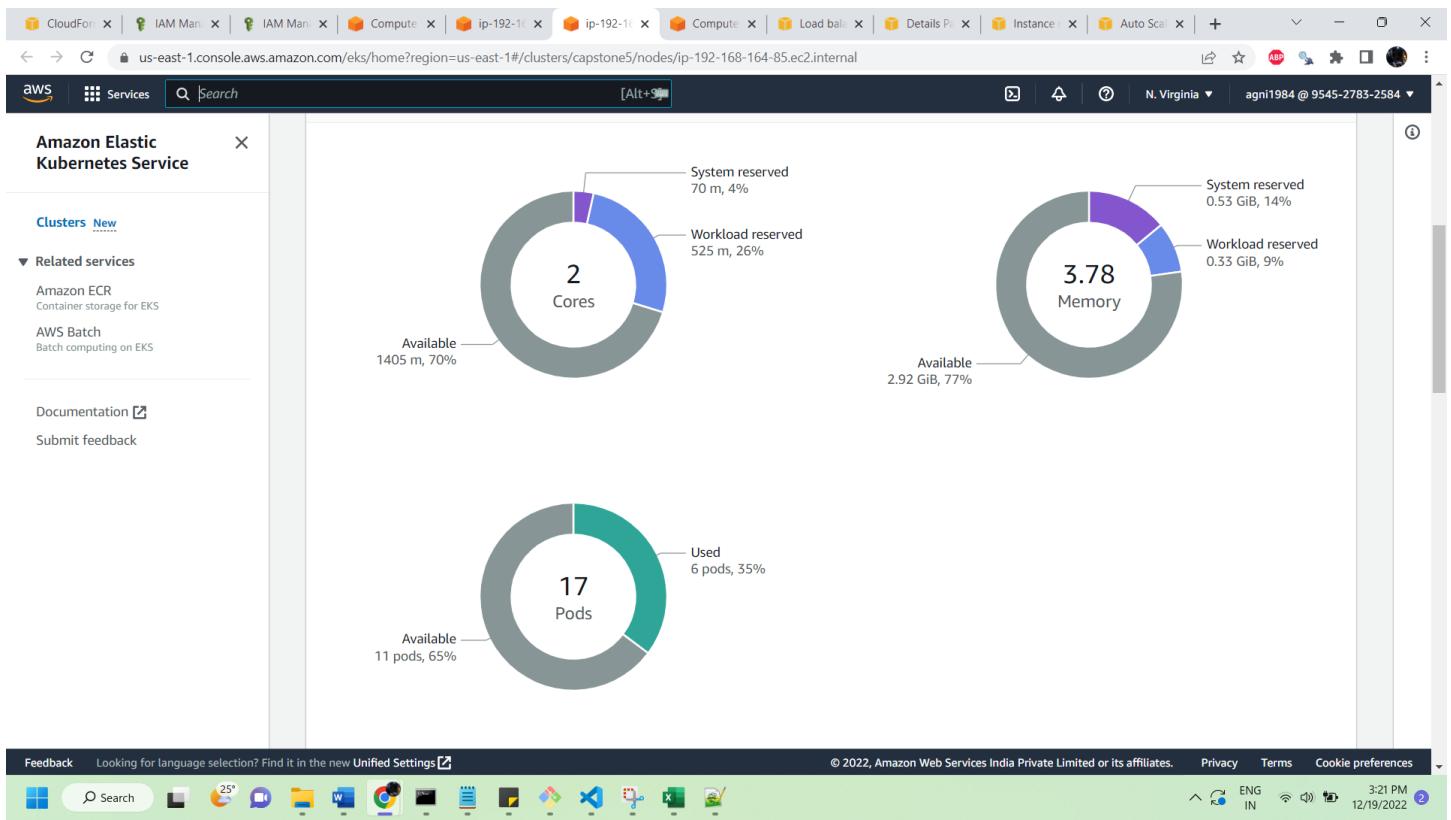
Conditions

Name	Status	Message
MemoryPressure	False	kubelet has sufficient memory available

CloudFront IAM Manager IAM Manager Compute ip-192-168-135-51.ec2.internal ip-192-168-135-51.ec2.internal Compute Load bal. Details Page Instances Auto Scaling +

aws Services Search [Alt+Shift]

N. Virginia agni1984 @ 9545-2783-2584



CloudFront | IAM Manager | IAM Manager | Compute | ip-192-168-164-85.ec2.internal | Compute | Load balancer | Details | Instances | Auto Scaling | + | - | X | ↗ | ⚙ | ⓘ | N. Virginia | agni1984 @ 9545-2783-2584 | + | ↗ | ⚙ | ⓘ | 1

Amazon Elastic Kubernetes Service

Clusters New

Related services

- Amazon ECR Container storage for EKS
- AWS Batch Batch computing on EKS

Documentation  [Submit feedback](#)

Feedback Looking for language selection? Find it in the new Unified Settings 

© 2022, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

Pods (6) 			
Name	Status	Created	IP
redis-follower-84fcc94dfc-zl9f5	 Running	15 minutes ago	192.168.179.96
redis-leader-766465cd9c-5nd9r	 Running	17 minutes ago	192.168.189.11
aws-node-dpk8x	 Running	2 hours ago	192.168.164.85
coredns-d5b9bfc4-2zr57	 Running	2 hours ago	192.168.171.207
coredns-d5b9bfc4-dh4qp	 Running	2 hours ago	192.168.188.28
kube-proxy-gxbt	 Running	2 hours ago	192.168.164.85

Conditions		
Name	Status	Message
MemoryPressure	False	kubelet has sufficient memory available

ENG IN 3:21 PM 12/19/2022

7.2. Scale the Nodes (EC2)

As two EC2 node capacity is enough for currently running pods, it is not required to scale up the EC2 node.

Increase number pods for frontend to 30 and redis-follower to 12 and check again

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>kubectl scale deployment frontend --replicas=30
deployment.apps/frontend scaled
```

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>kubectl get pods -l app=guestbook -l tier=frontend
NAME          READY   STATUS    RESTARTS   AGE
frontend-57df59b89c-486hj  1/1     Running   0          2m8s
frontend-57df59b89c-4lp74  1/1     Running   0          18m
frontend-57df59b89c-548js  1/1     Running   0          2m8s
frontend-57df59b89c-59sb7  1/1     Running   0          2m8s
frontend-57df59b89c-512bb  1/1     Running   0          2m8s
frontend-57df59b89c-6wggf  1/1     Running   0          2m8s
frontend-57df59b89c-7b891  1/1     Running   0          2m8s
frontend-57df59b89c-999bj  1/1     Running   0          2m8s
frontend-57df59b89c-91mlc  1/1     Running   0          2m8s
frontend-57df59b89c-9v4jd  1/1     Running   0          2m8s
frontend-57df59b89c-b5plh  1/1     Running   0          2m8s
frontend-57df59b89c-cksl1  1/1     Running   0          2m8s
frontend-57df59b89c-c164c  1/1     Running   0          2m8s
frontend-57df59b89c-dbzwk  1/1     Running   0          2m8s
frontend-57df59b89c-f5dz6  1/1     Running   0          2m8s
frontend-57df59b89c-fgphc  1/1     Running   0          2m8s
frontend-57df59b89c-frtlk  1/1     Running   0          2m8s
frontend-57df59b89c-fv92b  1/1     Running   0          2m8s
frontend-57df59b89c-g19hn  1/1     Running   0          2m8s
frontend-57df59b89c-htfgh  1/1     Running   0          2m8s
frontend-57df59b89c-j661k  1/1     Running   0          2m8s
frontend-57df59b89c-jx5ns  1/1     Running   0          2m8s
frontend-57df59b89c-knqpp  1/1     Running   0          2m8s
frontend-57df59b89c-njkbt  1/1     Running   0          18m
frontend-57df59b89c-n2qb6  1/1     Running   0          2m8s
frontend-57df59b89c-qmvhg  1/1     Running   0          2m8s
frontend-57df59b89c-rq7xr  1/1     Running   0          2m8s
frontend-57df59b89c-tqgpm  1/1     Running   0          2m8s
frontend-57df59b89c-v5zq5  1/1     Running   0          2m8s
frontend-57df59b89c-z6fls  1/1     Running   0          2m8s
```

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>kubectl scale deployment redis-follower --replicas=12
deployment.apps/redis-follower scaled
```

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>kubectl get pods -l app=redis -l tier=backend -l role=follower
NAME          READY   STATUS    RESTARTS   AGE
redis-follower-84fcc94dfc-4h6jx  1/1     Running   0          56s
redis-follower-84fcc94dfc-6zz5s  1/1     Running   0          21m
redis-follower-84fcc94dfc-842r5  1/1     Running   0          56s
redis-follower-84fcc94dfc-dbrpg  1/1     Running   0          56s
redis-follower-84fcc94dfc-g2vts  1/1     Running   0          56s
redis-follower-84fcc94dfc-hlgh6  1/1     Running   0          56s
redis-follower-84fcc94dfc-jhvbx 1/1     Running   0          56s
redis-follower-84fcc94dfc-mwzg8  1/1     Running   0          56s
redis-follower-84fcc94dfc-qlzwn  1/1     Running   0          56s
redis-follower-84fcc94dfc-wx5df  0/1     Pending   0          56s
redis-follower-84fcc94dfc-xlgwr  1/1     Running   0          56s
redis-follower-84fcc94dfc-z19fs  1/1     Running   0          21m
```

Number of nodes in node group is increased to 4, as four EC2 instances are required to run all pods.

CloudFor x IAM Man x IAM Man x Compute x ip-192-1 x ip-192-1 x Compute x Load bal x Details Po x Instance x Auto Sc x + us-east-1.console.aws.amazon.com/eks/home?region=us-east-1#/clusters/capstone5?selectedTab=cluster-compute-tab N. Virginia agni1984 @ 9545-2783-2584 [Alt+S]

Amazon Elastic Kubernetes Service

Clusters New

Related services

- Amazon ECR Container storage for EKS
- AWS Batch Batch computing on EKS

Documentation  Submit feedback

Cluster info 

Kubernetes version  1.23 Status  Active Provider EKS

Compute Overview Resources Networking Add-ons Authentication Logging Update history Tags

Nodes (4) Info 

Filter Nodes by property or value < 1 >

Node name	Instance type	Node group	Created	Status
ip-192-168-114-62.ec2.internal	t3.medium	capstone5-ng	Created 4 minutes ago	 Ready
ip-192-168-135-51.ec2.internal	t3.medium	capstone5-ng	Created 2 hours ago	 Ready
ip-192-168-164-85.ec2.internal	t3.medium	capstone5-ng	Created 2 hours ago	 Ready
ip-192-168-182-186.ec2.internal	t3.medium	capstone5-ng	Created a minute ago	 Ready

Node groups (1) 

Feedback Looking for language selection? Find it in the new Unified Settings 

© 2022, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

ENG IN 3:29 PM 12/19/2022

CloudFor x IAM Man x IAM Man x Compute x ip-192-1 x ip-192-1 x ip-192-1 x Compute x Load bal x Details Po x Instance x Auto Sc x + us-east-1.console.aws.amazon.com/eks/home?region=us-east-1#/clusters/capstone5/nodes/ip-192-168-114-62.ec2.internal N. Virginia agni1984 @ 9545-2783-2584 [Alt+S]

Amazon Elastic Kubernetes Service

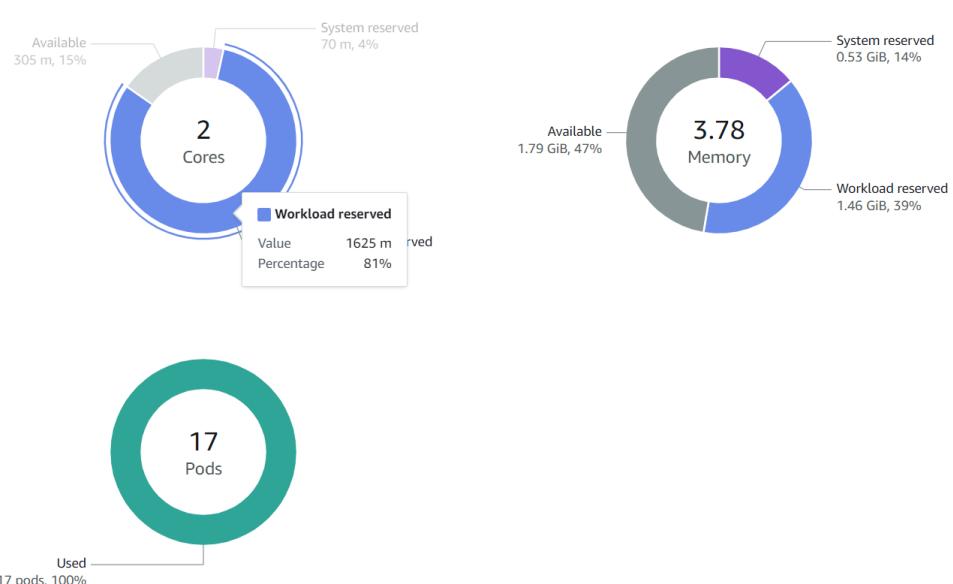
Clusters New

Related services

- Amazon ECR Container storage for EKS
- AWS Batch Batch computing on EKS

Documentation  Submit feedback

Compute



System reserved

Cores: Available 305 m, 15%, System reserved 70 m, 4%. Workload reserved 1625 m, 81%.

Memory: Available 1.79 GiB, 47%, System reserved 0.53 GiB, 14%. Workload reserved 1.46 GiB, 39%.

Pods: Used 17 pods, 100%.

Feedback Looking for language selection? Find it in the new Unified Settings 

© 2022, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

ENG IN 3:29 PM 12/19/2022

Cloud | IAM M | IAM N | Comp | ip-192-168-114-62.ec2.internal | ip-192-168-114-62.ec2.internal | ip-192-168-114-62.ec2.internal | Comp | Load | Detail | Instances | Auto | + | - | _ | ↻ | ⚙ | ? | N. Virginia | agni1984 @ 9545-2783-2584 | 9:00 AM | 12/19/2022

Amazon Elastic Kubernetes Service

Clusters New

Related services

- Amazon ECR Container storage for EKS
- AWS Batch Batch computing on EKS

Documentation **Submit feedback**

Name	Status	Created	IP
frontend-57df59b89c-548js	Running	6 minutes ago	192.168.122.254
frontend-57df59b89c-ck54l	Running	6 minutes ago	192.168.118.248
frontend-57df59b89c-dbwzk	Running	6 minutes ago	192.168.121.232
frontend-57df59b89c-htfgh	Running	6 minutes ago	192.168.122.244
frontend-57df59b89c-n2qb6	Running	6 minutes ago	192.168.103.19
frontend-57df59b89c-qmvhg	Running	6 minutes ago	192.168.124.237
redis-follower-84fcc94dfc-4h6jx	Running	3 minutes ago	192.168.115.198
redis-follower-84fcc94dfc-842r5	Running	3 minutes ago	192.168.120.108
redis-follower-84fcc94dfc-dbrpg	Running	3 minutes ago	192.168.127.5
redis-follower-84fcc94dfc-g2vts	Running	3 minutes ago	192.168.118.46

Conditions

Name	Status	Message
MemoryPressure	False	kubelet has sufficient memory available
DiskPressure	False	kubelet has no disk pressure

Feedback Looking for language selection? Find it in the new Unified Settings

© 2022, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

ENG IN 3:30 PM 12/19/2022

Cloud | IAM M | IAM N | Comp | ip-192-168-114-62.ec2.internal | ip-192-168-114-62.ec2.internal | ip-192-168-114-62.ec2.internal | Comp | Load | Detail | Instances | Auto | + | - | _ | ↻ | ⚙ | ? | N. Virginia | agni1984 @ 9545-2783-2584 | 9:00 AM | 12/19/2022

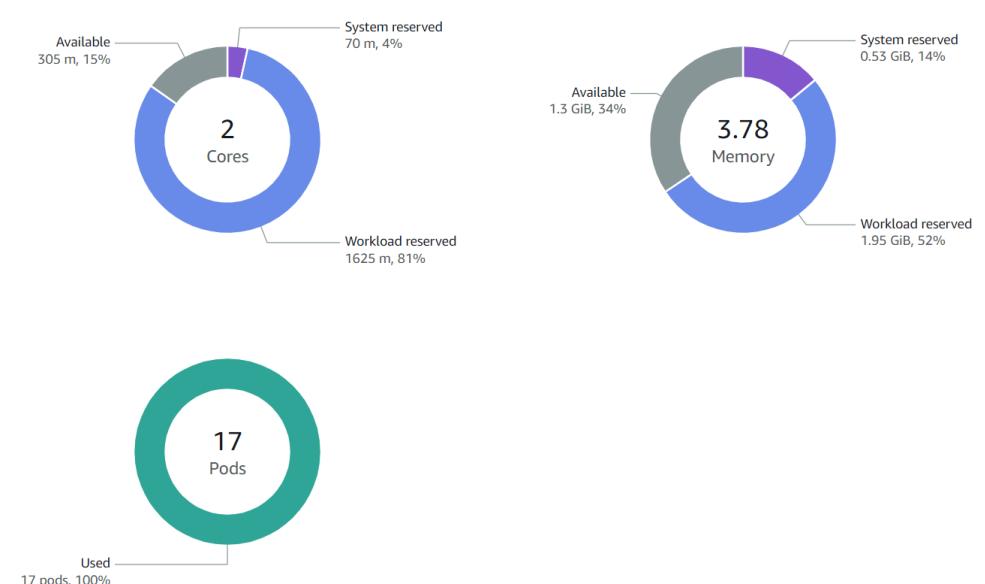
Amazon Elastic Kubernetes Service

Clusters New

Related services

- Amazon ECR Container storage for EKS
- AWS Batch Batch computing on EKS

Documentation **Submit feedback**



The dashboard displays resource usage statistics:

- Cores:** 2 Cores. Available: 305 m, 15%. System reserved: 70 m, 4%. Workload reserved: 1625 m, 81%.
- Memory:** 3.78 GiB. Available: 1.3 GiB, 34%. System reserved: 0.53 GiB, 14%. Workload reserved: 1.95 GiB, 52%.
- Pods:** 17 Pods. Used: 17 pods, 100%.

Feedback Looking for language selection? Find it in the new Unified Settings

© 2022, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

ENG IN 3:30 PM 12/19/2022

Cloud | IAM M | IAM N | Comp | ip-192-168-135-51.ec2.internal | ip-192-168-141-168 | ip-192-168-140-81 | ip-192-168-137-218 | Load | Detail | Instances | Auto | + | - | X | ↗ | ⚙ | ⓘ | N. Virginia | agni1984 @ 9545-2783-2584 | 3:30 PM | 12/19/2022

Amazon Elastic Kubernetes Service

Clusters New

Related services

- Amazon ECR Container storage for EKS
- AWS Batch Batch computing on EKS

Documentation  Submit feedback

Used – 17 pods, 100%

Pods (17) Info

Name	Status	Created	IP
frontend-57df59b89c-4lp74	Running	22 minutes ago	192.168.137.218
frontend-57df59b89c-59sb7	Running	6 minutes ago	192.168.141.168
frontend-57df59b89c-5l2bb	Running	6 minutes ago	192.168.140.81
frontend-57df59b89c-7b89l	Running	6 minutes ago	192.168.137.84
frontend-57df59b89c-cl64c	Running	6 minutes ago	192.168.156.20
frontend-57df59b89c-frtlk	Running	6 minutes ago	192.168.145.51
frontend-57df59b89c-gl9hn	Running	6 minutes ago	192.168.145.88
frontend-57df59b89c-knqpp	Running	6 minutes ago	192.168.152.115
frontend-57df59b89c-nmjkb	Running	22 minutes ago	192.168.141.62
frontend-57df59b89c-rq7xr	Running	6 minutes ago	192.168.142.37

Feedback Looking for language selection? Find it in the new Unified Settings 

© 2022, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

Cloud | IAM M | IAM N | Comp | ip-192-168-135-51.ec2.internal | ip-192-168-141-168 | ip-192-168-140-81 | ip-192-168-137-218 | Load | Detail | Instances | Auto | + | - | X | ↗ | ⚙ | ⓘ | N. Virginia | agni1984 @ 9545-2783-2584 | 3:30 PM | 12/19/2022

Amazon Elastic Kubernetes Service

Clusters New

Related services

- Amazon ECR Container storage for EKS
- AWS Batch Batch computing on EKS

Documentation  Submit feedback

Available – 305 m, 15% System reserved 70 m, 4%

2 Cores

Workload reserved 1625 m, 81%

Available – 1.85 GiB, 49% System reserved 0.53 GiB, 14%

3.78 Memory

Workload reserved 1.41 GiB, 37%

Used – 17 pods, 100%

17 Pods

Cloud | IAM | Comp | ip-192-168-164-85.ec2.internal | ip-192-168-164-86.ec2.internal | ip-192-168-164-87.ec2.internal | ip-192-168-164-88.ec2.internal | Load | Detail | Instances | Auto Scaling | + | - | X | ↻ | ⚙️ | ? | N. Virginia | agni1984 @ 9545-2783-2584 | 9545-2783-2584

Amazon Elastic Kubernetes Service

Clusters New

Related services

- Amazon ECR Container storage for EKS
- AWS Batch Batch computing on EKS

Documentation  

Submit feedback

Name	Status	Created	IP
frontend-57df59b89c-486hj	Running	6 minutes ago	192.168.172.151
frontend-57df59b89c-gwggf	Running	6 minutes ago	192.168.162.1
frontend-57df59b89c-999bj	Running	6 minutes ago	192.168.164.153
frontend-57df59b89c-9lmlc	Running	6 minutes ago	192.168.164.147
frontend-57df59b89c-9v4jd	Running	6 minutes ago	192.168.162.213
frontend-57df59b89c-b5plh	Running	6 minutes ago	192.168.183.245
frontend-57df59b89c-f5dz6	Running	6 minutes ago	192.168.188.133
frontend-57df59b89c-fgphc	Running	6 minutes ago	192.168.173.171
frontend-57df59b89c-fv92b	Running	6 minutes ago	192.168.184.249
frontend-57df59b89c-j66lk	Running	6 minutes ago	192.168.186.242

Conditions

Name	Status	Message
MemoryPressure	False	kubelet has sufficient memory available
DiskPressure	False	kubelet has no disk pressure

Feedback Looking for language selection? Find it in the new Unified Settings 

© 2022, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

ENG IN 3:30 PM 12/19/2022

Cloud | IAM | Comp | ip-192-168-164-85.ec2.internal | ip-192-168-164-86.ec2.internal | ip-192-168-164-87.ec2.internal | ip-192-168-164-88.ec2.internal | Load | Detail | Instances | Auto Scaling | + | - | X | ↻ | ⚙️ | ? | N. Virginia | agni1984 @ 9545-2783-2584 | 9545-2783-2584

Amazon Elastic Kubernetes Service

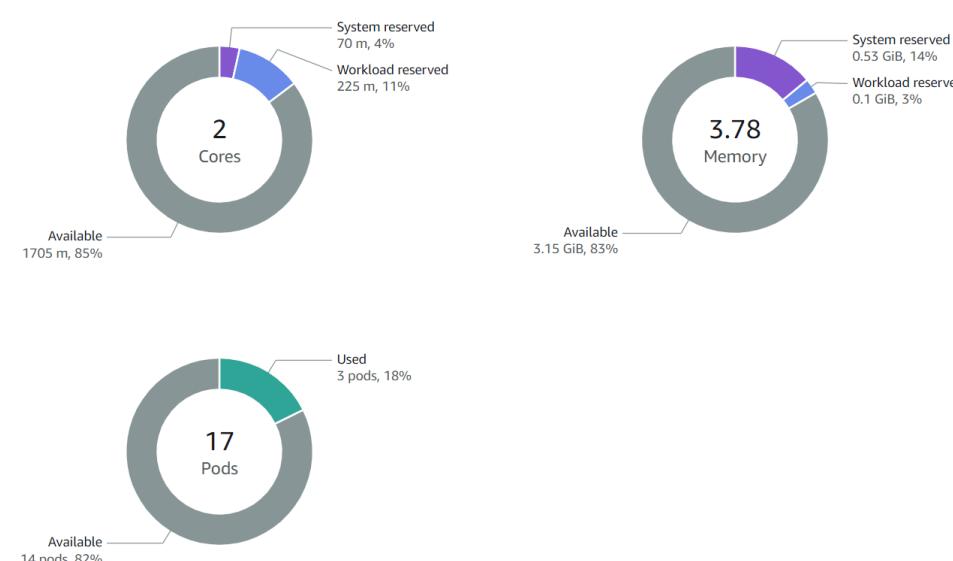
Clusters New

Related services

- Amazon ECR Container storage for EKS
- AWS Batch Batch computing on EKS

Documentation  

Submit feedback



2 Cores

- System reserved: 70 m, 4%
- Workload reserved: 225 m, 11%
- Available: 1705 m, 85%

3.78 Memory

- System reserved: 0.53 GiB, 14%
- Workload reserved: 0.1 GiB, 3%
- Available: 3.15 GiB, 83%

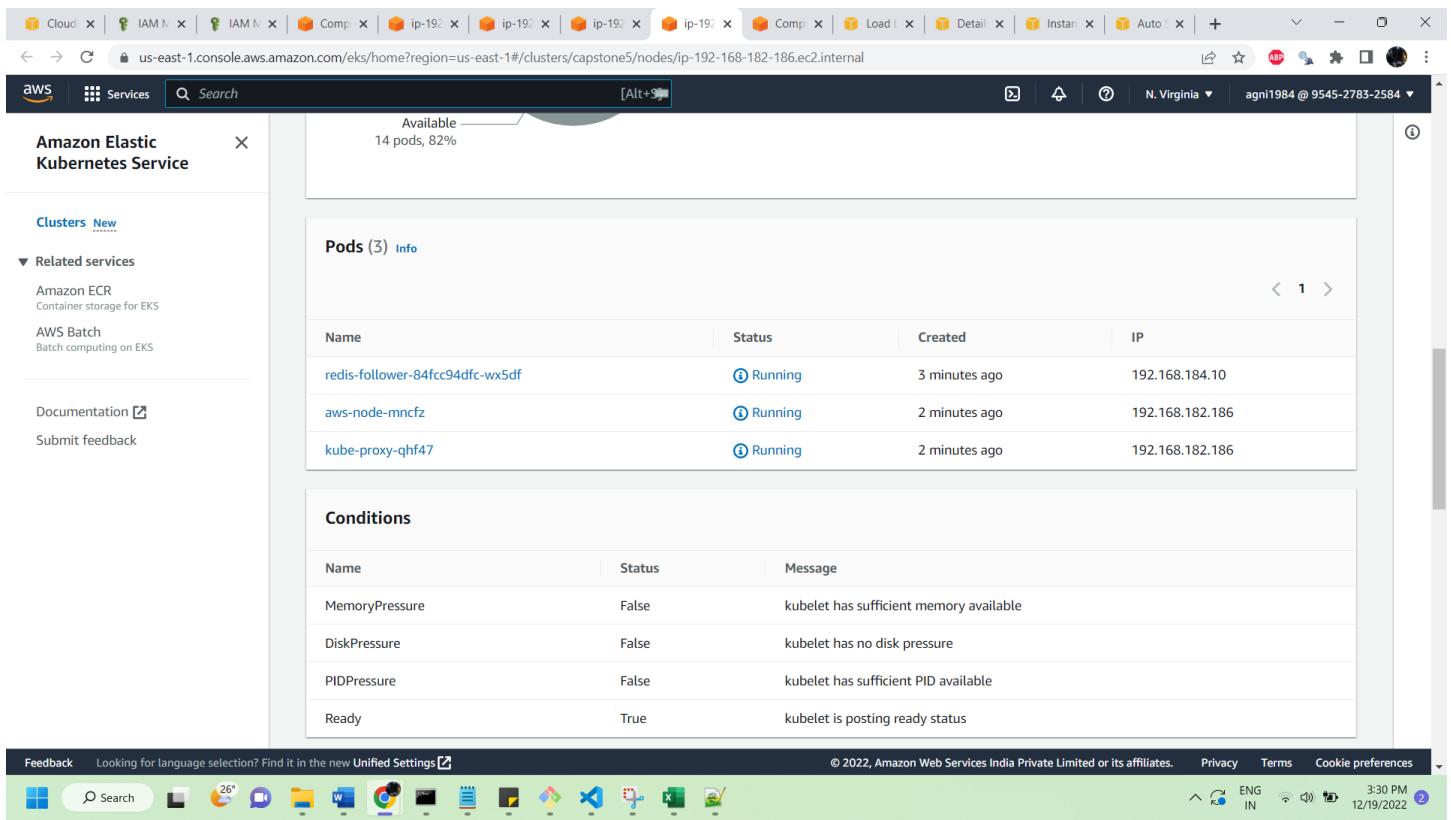
17 Pods

- Used: 3 pods, 18%
- Available: 14 pods, 82%

Feedback Looking for language selection? Find it in the new Unified Settings 

© 2022, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

ENG IN 3:30 PM 12/19/2022



The screenshot shows the AWS CloudWatch Metrics interface for an EKS cluster. The top navigation bar includes tabs for CloudWatch Metrics, IAM Policies, IAM Roles, Compromised Credentials, IP Addresses, CloudWatch Metrics Insights, CloudWatch Metrics Insights Details, CloudWatch Metrics Insights Instances, CloudWatch Metrics Insights Auto Scaling, and CloudWatch Metrics Insights Metrics. The main content area displays metrics for the Amazon Elastic Kubernetes Service, showing 14 pods available at 82% utilization.

Clusters (New)

- Related services
 - Amazon ECR: Container storage for EKS
 - AWS Batch: Batch computing on EKS
- Documentation
- Submit feedback

Pods (3) info

Name	Status	Created	IP
redis-follower-84fcc94dfc-wx5df	Running	3 minutes ago	192.168.184.10
aws-node-mncfz	Running	2 minutes ago	192.168.182.186
kube-proxy-qhf47	Running	2 minutes ago	192.168.182.186

Conditions

Name	Status	Message
MemoryPressure	False	kubelet has sufficient memory available
DiskPressure	False	kubelet has no disk pressure
PIDPressure	False	kubelet has sufficient PID available
Ready	True	kubelet is posting ready status

Feedback Looking for language selection? Find it in the new Unified Settings 

© 2022, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

ENG IN 3:30 PM 12/19/2022

Based on above screenshots, 17 pods are running in each three EC2 instances and 3 pods are running in last EC2 instance.

Scale down pods to original value,

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>kubectl scale deployment frontend --replicas=3
deployment.apps/frontend scaled
```

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>kubectl scale deployment redis-follower --replicas=2
deployment.apps/redis-follower scaled
```

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>kubectl get pods -l app=guestbook -l tier=frontend
NAME          READY   STATUS    RESTARTS   AGE
frontend-57df59b89c-cks4l  1/1     Running   0          11m
frontend-57df59b89c-n2qb6  1/1     Running   0          11m
frontend-57df59b89c-qmvhg  1/1     Running   0          11m
```

```
E:\GreatLakes_Cloud\CapstoneProject\aws-capstone5\aws_sample_projects\examples\guestbook\capstone5>kubectl get pods -l app=redis -l tier=backend -l role=follower
NAME          READY   STATUS    RESTARTS   AGE
redis-follower-84fcc94dfc-6zz5s  1/1     Running   0          29m
redis-follower-84fcc94dfc-z19f5  1/1     Running   0          29m
```

Number of EC2 instances reduced back to two instances

Screenshot of the AWS CloudFormation console showing the status of two EC2 instances in a Kubernetes cluster.

Nodes (2) Info

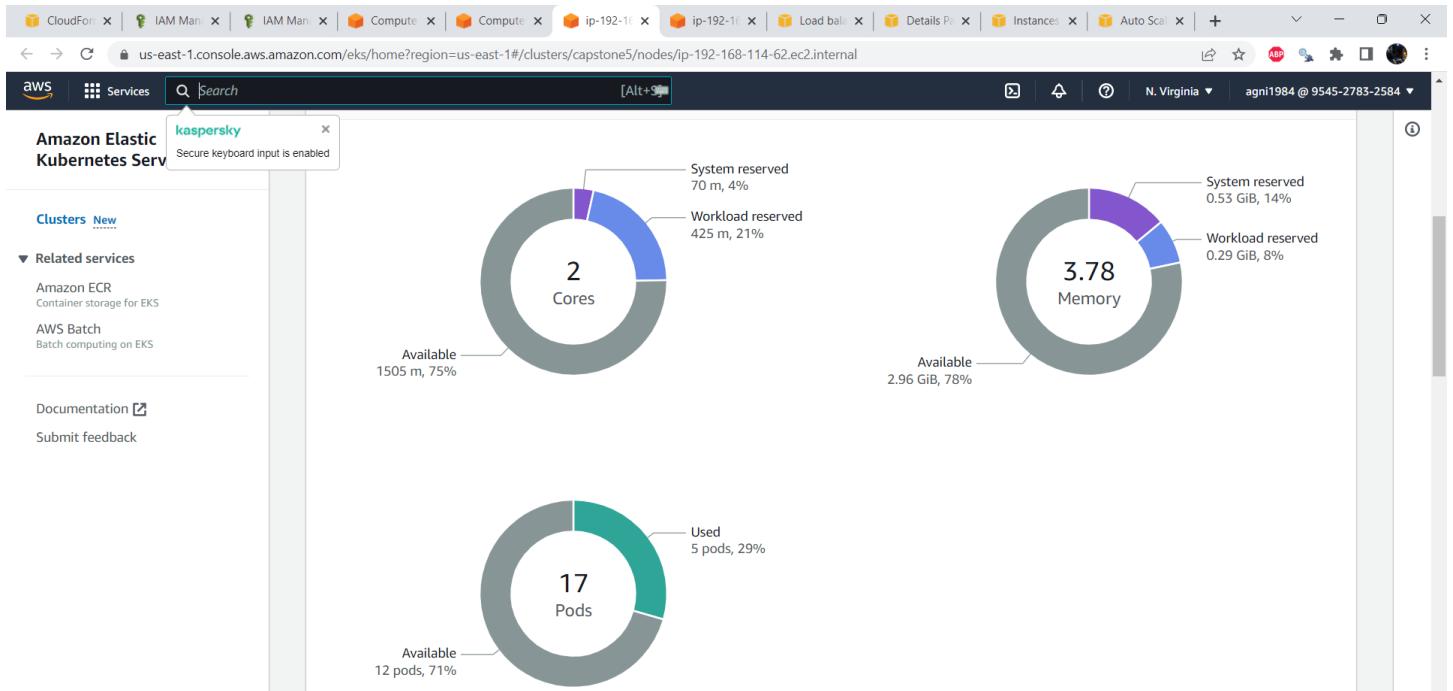
Node name	Instance type	Node group	Created	Status
ip-192-168-114-62.ec2.internal	t3.medium	capstone5-ng	Created 15 minutes ago	Ready
ip-192-168-135-51.ec2.internal	t3.medium	capstone5-ng	Created 2 hours ago	Ready

Node groups (1) Info

Group name	Desired size	AMI release version	Launch template	Status
capstone5-ng	4	1.23.13-20221112	eksctl-capstone5-nodegroup-capstone5-ng (1)	Active

Fargate profiles (0) Info

No Fargate profiles
This cluster does not have any Fargate profiles.



Feedback Looking for language selection? Find it in the new Unified Settings

© 2022, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

ENG IN 3:41 PM 12/19/2022 2

CloudFront | IAM Manager | IAM Manager | Compute | Compute | ip-192-1 | ip-192-1 | Load bal. | Details P | Instances | Auto Scal. | + | N. Virginia | agni1984 @ 9545-2783-2584 | 3:41 PM | 12/19/2022 | 2

Amazon Elastic Kubernetes Service

Clusters New

Related services

- Amazon ECR Container storage for EKS
- AWS Batch Batch computing on EKS

Documentation **Submit feedback**

Feedback Looking for language selection? Find it in the new Unified Settings

© 2022, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

ENG IN 3:41 PM 12/19/2022 2

Pods (5)

Name	Status	Created	IP
frontend-57df59b89c-ck54l	Running	17 minutes ago	192.168.118.248
frontend-57df59b89c-n2qb6	Running	17 minutes ago	192.168.103.19
frontend-57df59b89c-qmvhg	Running	17 minutes ago	192.168.124.237
aws-node-6zxms	Running	16 minutes ago	192.168.114.62
kube-proxy-r6mgn	Running	16 minutes ago	192.168.114.62

Conditions

Name	Status	Message
MemoryPressure	False	kubelet has sufficient memory available
DiskPressure	False	kubelet has no disk pressure

CloudFor x IAM Man x IAM Man x Compute x Compute x ip-192-1 x ip-192-1 x Load bal x Details Po x Instances x Auto Scal x + N. Virginia agn1984 @ 9545-2783-2584

Amazon Elastic Kubernetes Service

kaspersky Secure keyboard input is enabled

Available 1205 m, 60% Cores Available 2.24 GiB, 59% Memory Available 1.02 GiB, 27% Worker Node

Clusters New

Related services

- Amazon ECR Container storage for EKS
- AWS Batch Batch computing on EKS

Documentation Submit feedback

Pods (8) info

Name	Status	Created	IP
redis-follower-84fcc94dfc-6pn8b	Running	a few seconds ago	192.168.137.218
redis-follower-84fcc94dfc-6zz5s	Running	34 minutes ago	192.168.137.18
redis-leader-766465cd9c-72z6s	Running	a few seconds ago	192.168.141.62
aws-node-m62bs	Running	2 hours ago	192.168.135.51
cluster-autoscaler-5c6d5bdd74-bgj97	Running	40 minutes ago	192.168.150.33
coredns-d5b9bfc4-85vq8	Running	a few seconds ago	192.168.153.171
coredns-d5b9bfc4-mnb99	Running	a few seconds ago	192.168.152.33
kube-proxy-f6w9r	Running	2 hours ago	192.168.135.51

Feedback Looking for language selection? Find it in the new Unified Settings 

© 2022, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

ENG IN 3:41 PM 12/19/2022

CloudFor x IAM Man x IAM Man x Compute x Compute x ip-192-1 x ip-192-1 x Load bal x Details Po x Instances x Auto Scal x + N. Virginia agn1984 @ 9545-2783-2584

Amazon Elastic Kubernetes Service

Clusters New

Related services

- Amazon ECR Container storage for EKS
- AWS Batch Batch computing on EKS

Documentation Submit feedback

Name	Status	Created	IP
redis-follower-84fcc94dfc-6pn8b	Running	a few seconds ago	192.168.137.218
redis-follower-84fcc94dfc-6zz5s	Running	34 minutes ago	192.168.137.18
redis-leader-766465cd9c-72z6s	Running	a few seconds ago	192.168.141.62
aws-node-m62bs	Running	2 hours ago	192.168.135.51
cluster-autoscaler-5c6d5bdd74-bgj97	Running	40 minutes ago	192.168.150.33
coredns-d5b9bfc4-85vq8	Running	a few seconds ago	192.168.153.171
coredns-d5b9bfc4-mnb99	Running	a few seconds ago	192.168.152.33
kube-proxy-f6w9r	Running	2 hours ago	192.168.135.51

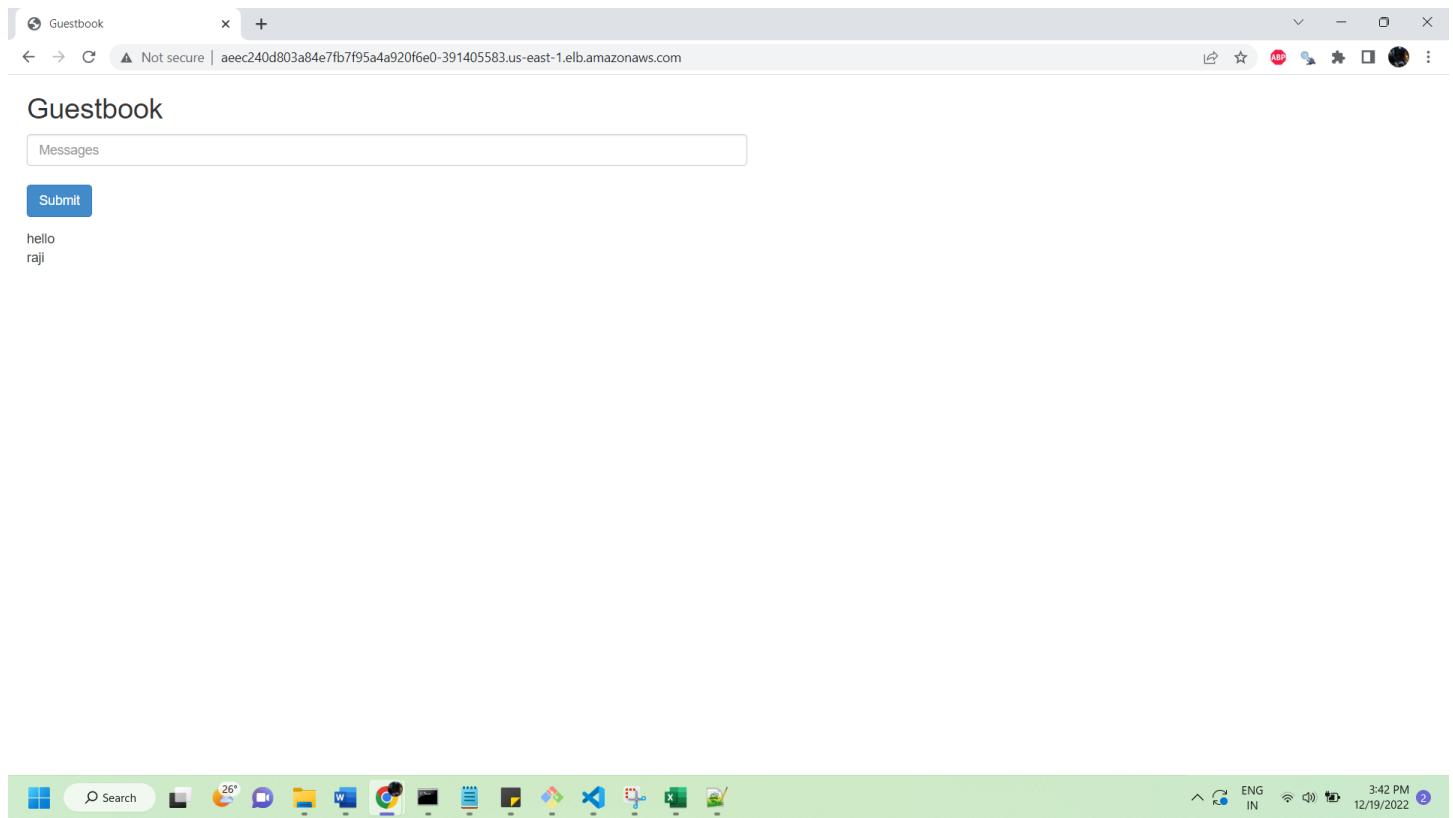
Conditions

Name	Status	Message
MemoryPressure	False	kubelet has sufficient memory available
DiskPressure	False	kubelet has no disk pressure
PIDPressure	False	kubelet has sufficient PID available
Ready	True	kubelet is posting ready status

Feedback Looking for language selection? Find it in the new Unified Settings 

© 2022, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

ENG IN 3:41 PM 12/19/2022



8. Cost Analysis

Problem: Cost Analysis of the implemented Solution. Assume your solution is used by 1000 users for a month, and give monthly billing estimates.

Observations:

Based on observation, t3.medium EC2 instances can run up to 17 pods.

Assumptions:

Worst Case Scenario (Peak time): Assume that 1000 users access 'Guest Book Application' at same time

Normal: Assume 10 users access 'Guest Book Application' at same time

Assume with one backend end pod, 2 users can access site. 30 EC2 instances needed during peak time.

Estimate summary			
Upfront cost	Monthly cost	Total 12 months cost	Currency
0	225.22	2,702.66	USD

* Includes upfront cost

Detailed Estimate						
Region	Service	Upfront	Monthly	First 12 months total	Currency	Configuration summary
US East (N. Virginia)	Amazon EKS	0	73	876	USD	Number of EKS Clusters (1)
US East (N. Virginia)	AWS CloudFormation	0	0	0	USD	Number of third-party extensions managed (0), Average duration per operation (30 seconds), Total number of operations per extension (0 per day)
US East (N. Virginia)	Amazon EC2	0	152.2216	1826.66	USD	Tenancy (Shared Instances), Operating system (Linux), DT Inbound: Not selected (0 TB per month), DT Outbound: Not selected (0 TB per month), DT Intra-Region: (0 TB per month), Workload (Monthly, Baseline: 2, Peak: 30, Duration of peak: 3 Day 0 Hr 0 Min), Snapshot Frequency (No snapshot storage), Enable disabled, Advance EC2 instance (t3.medium), Pricing strategy (On-Demand), Storage amount (20 GB)

Acknowledgement

* AWS Pricing Calculator provides only an estimate of your AWS fees and doesn't include any taxes that might apply. Your actual fees depend on a variety of factors, including your actual usage of AWS services.

<https://calculator.aws/#/estimate?id=4f0be107e823df5d2aa9894585c9008e8d238452>

9. Lessons & Observations

- Learnt to create EKS cluster in AWS using AWS CLI
- Learnt to create node group for cluster
- Learnt to use Kubernetes to deploy application in EKS cluster in AWS
- Learnt to configure Load balancer and auto scaling for EKS cluster