**Capstone Project**

**Super Mario Game Deployment​​​**

Steps to be followed:

1. Launch an EC2 Instance
2. Install and configure Docker, Terraform, AWS CLI, and kubectl
3. Create an IAM role
4. Assign the IAM role and set up the project
5. Set up an S3 Bucket
6. Configure Terraform and deploy the application

**Step 1:** **Launch an EC2 Instance**

1. Log in to the AWS Management ConsoleA screenshot of a computer

   Description automatically generated
2. Navigate to **EC2** by searching for it in the services search bar and selecting the service

A screenshot of a computer

Description automatically generated

* 1. Click **Launch instances**   
       
     A screenshot of a computer

     Description automatically generated
  2. Provide a name for your instance  
       
     A screenshot of a computer

     Description automatically generated
  3. Select **Ubuntu** as the AMI (Amazon Machine Image)  
       
     A screenshot of a computer

     Description automatically generated
  4. Choose the instance type **t3.medium**  
       
     A screenshot of a computer program

     Description automatically generated
  5. Click **Create new key pair**, enter a name for the key pair, and click **Create key pair**  
       
     A screenshot of a computer

     Description automatically generated  
     A screenshot of a computer

     Description automatically generated
  6. Select all three security group rules  
       
     A screenshot of a computer

     Description automatically generated
  7. Click **Launch instance**  
       
     A screenshot of a computer

     Description automatically generated

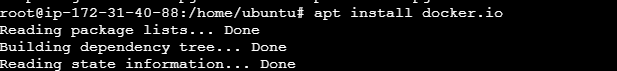
**Step 2: Install and configure Docker, Terraform, AWS CLI, and kubectl**

* 1. Select the instance you created, then click **Connect**  
     A screenshot of a computer

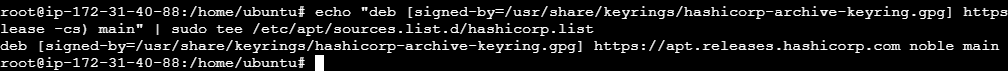
     Description automatically generated
  2. Click on **Connect**  
       
     A screenshot of a chat

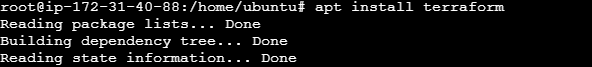
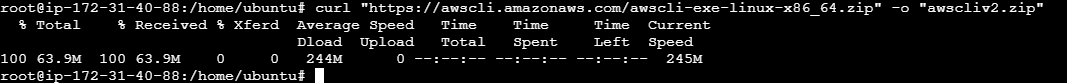
     Description automatically generated
  3. Switch to the root user using the following command:  
     **sudo su**  
     A screenshot of a computer program

     Description automatically generated
  4. Update the package manager using the following command:  
     **apt update**A screen shot of a computer

     Description automatically generated
  5. Install Docker using the following command: **apt install docker.io**
  6. Add the current user to the Docker group using the following command:  
     **usermod -aG docker $ubuntu**A screen shot of a computer

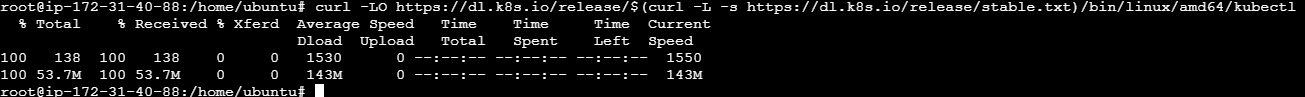
     Description automatically generated
  7. Apply group changes using the following command:  
     **newgrp docker**
  8. Add the HashiCorp GPG key using the following command:  
      **wget -O- https://apt.releases.hashicorp.com/gpg | sudo gpg --dearmor -o /usr/share/keyrings/hashicorp-archive-keyring.gpg**  
     A screenshot of a computer screen

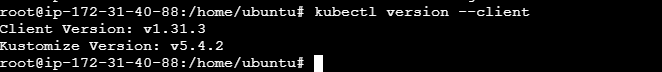
     Description automatically generated
  9. Configure the Terraform repository using the following command: **echo "deb [signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] https://apt.releases.hashicorp.com $(lsb\_release -cs) main" | sudo tee /etc/apt/sources.list.d/hashicorp.list**
  10. Update the package manager again using the following command:  
      **apt update**A screen shot of a computer

      Description automatically generated
  11. Install Terraform using the following command: **apt install terraform**  
      
  12. Download the AWS CLI installer using the following command: **curl "https://awscli.amazonaws.com/awscli-exe-linux-x86\_64.zip" -o "awscliv2.zip"**
  13. Install the unzip utility using the following command:  
      **apt install unzip -y**A black screen with white text

      Description automatically generated
  14. Unzip the AWS CLI installer using the following command:  
      **unzip awscliv2.zip**A computer screen shot of a black screen

      Description automatically generated
  15. Install the AWS CLI using the following command:  
      **./aws/install**  
        
      
  16. Install curl using the following command:  
      **apt install curl -y**  
      A screen shot of a computer

      Description automatically generated
  17. Download kubectl using the following command:  
      **curl -LO https://dl.k8s.io/release/$(curl -L -s https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl**  
      
  18. Install kubectl using the following command:  
      **install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl**  
      
  19. Verify the kubectl version using the following command:  
      **version --client**



**Step 3: Create an IAM role**

1. Navigate to **IAM** by searching for it in the AWS services search bar

A screenshot of a computer

Description automatically generated

1. In the left navigation pane, click **Roles**

A screenshot of a computer

Description automatically generated

1. Click **Create role**

A screenshot of a computer

Description automatically generated

1. Select **AWS service** as the trusted entity  
   A screenshot of a computer

   Description automatically generated
2. Choose **EC2** as the use caseA screenshot of a computer

   Description automatically generated
3. Click **Next**A screenshot of a computer

   Description automatically generated
4. Attach the **AdministratorAccess** policy  
     
   A screenshot of a computer

   Description automatically generated
5. Click **Next**A screenshot of a computer

   Description automatically generated
6. Enter a name for the role  
   A screenshot of a computer

   Description automatically generated
7. Click **Create role**  
     
   A screenshot of a computer

   Description automatically generated  
     
   The role is now successfully created.  
     
   A screenshot of a computer

   Description automatically generated

**Step 4: Assign the IAM role and set up the project**

1. Select your instance, click **Actions**, navigate to **Security**, and choose **Modify IAM role**  
     
   A screenshot of a computer

   Description automatically generated
2. Assign the role you created earlier, then click **Update IAM role**   
     
   A screenshot of a computer

   Description automatically generated
3. Create a directory for the project using the following command:  
   **mkdir supermario**   
   
4. Change into the new directory using the following command:  
   **cd supermario**
5. Clone the repository using the following command: **git clone https://github.com/simplilearn10/supermario-game.git**A black screen with white text

   Description automatically generated
6. List the files in the directory using the following command:   
   **ls**A black background with white text

   Description automatically generated
7. Navigate to the Terraform folder using the following command:   
   **cd EKS-TF/**A black background with white text

   Description automatically generated

**Step 5: Set up an S3 bucket**

1. Navigate to **S3** by searching for it in the AWS services search bar  
     
   A screenshot of a computer

   Description automatically generated
2. Click **Create bucket**  
     
   A screenshot of a computer

   Description automatically generated
3. Enter a name for your bucket  
     
   A screenshot of a computer

   Description automatically generated
4. Scroll down and click **Create bucket**  
     
   A screenshot of a computer

   Description automatically generated

The bucket is successfully created.  
A screenshot of a computer

Description automatically generated

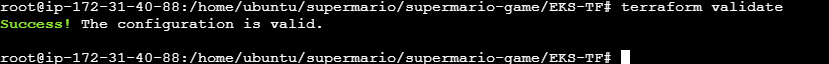
**Step 6: Configure Terraform and deploy the application**

1. Return to the terminal and list the files using the following command:  
   **ls**  
     
   A black background with white text

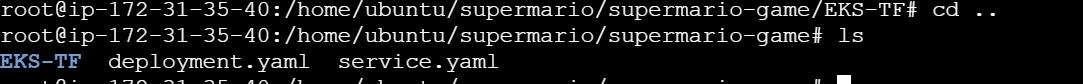
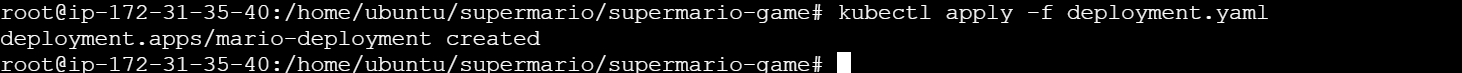
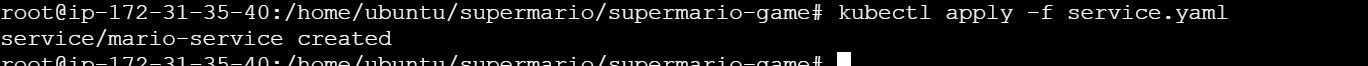
   Description automatically generated
2. Edit the backend.tf file using the following command:  
   **nano backend.tf**   
     
   A black background with white text

   Description automatically generated
3. Replace the bucket name with your bucket name using the following command:  
   A black screen with white text

   Description automatically generated
4. Initialize Terraform using the following command:  
   **terraform init**  
   A screenshot of a computer program

   Description automatically generated
5. Validate the Terraform configuration using the following command:  
   **terraform validate**  
     
   
6. Create a Terraform execution plan using the following command:  
   **terraform plan**  
   A screen shot of a computer

   Description automatically generated
7. Apply the configuration using the following command:  
   **terraform apply –auto-approve**  
     
   A screenshot of a computer

   Description automatically generated
8. Update the kubeconfig file for your cluster using the following command:  
   **aws eks update-kubeconfig --name EKS\_CLOUD --region ap-south-1**  
   
9. Exit the Terraform folder using the following command:  
   **cd ..**  
     
   
10. Apply the Kubernetes deployment using the following command:  
    **kubectl apply -f deployment.yaml**  
      
    
11. Apply the Kubernetes service using the following command:  
    **kubectl apply -f service.yaml**  
    
12. Verify the resources using the following command:  
    **kubectl get all**A screen shot of a computer

    Description automatically generated
13. Describe the service using the following command:  
    **kubectl describe service mario-service**A black screen with white text

    Description automatically generated
14. Copy the LoadBalancer Ingress URL from the output   
      
    A screenshot of a computer

    Description automatically generated
15. Paste the URL into a browser to access the Super Mario game interface  
      
    A screenshot of a video game

    Description automatically generated  
    By following these steps, you have successfully deployed the Super Mario game on an AWS EKS cluster, making it accessible via a LoadBalancer URL for users to enjoy the game interface.