# Setting Up Django Blog with Docker and S3 Storage

## Prerequisites

1. EC2 instance running
2. blog\_project.pem key file
3. AWS credentials configured locally (use aws configure if not set up)
4. Security group with ports 22 (SSH) and 80 (HTTP) open

## Step 1: Set Key Permissions

chmod 400 blog\_project.pem

## Step 2: Verify AWS Credentials

aws configure list

Expected output should show credentials are configured.

## Step 3: Install Docker Compose

ssh -t -i blog\_project.pem ubuntu@YOUR\_EC2\_IP "sudo apt update && sudo apt install docker-compose -y"

Replace YOUR\_EC2\_IP with your EC2 instance’s public IP.

## Step 4: Deploy Container with AWS Credentials

Copy and paste this entire command block:

ssh -t -i blog\_project.pem ubuntu@YOUR\_EC2\_IP \  
 "AWS\_ACCESS\_KEY=$(aws configure get aws\_access\_key\_id); \  
 AWS\_SECRET\_KEY=$(aws configure get aws\_secret\_access\_key); \  
 sudo docker stop blog\_web || true; \  
 sudo docker rm blog\_web || true; \  
 sudo docker run -d --name blog\_web \  
 -p 80:8000 \  
 -e AWS\_ACCESS\_KEY\_ID=$AWS\_ACCESS\_KEY \  
 -e AWS\_SECRET\_ACCESS\_KEY=$AWS\_SECRET\_KEY \  
 -e AWS\_DEFAULT\_REGION=us-east-1 \  
 119922150720.dkr.ecr.us-east-1.amazonaws.com/blog\_project-web:3f3406d3bc3b5c568188e88769ad341ac74f744d"

## Step 5: Verify Container Status

ssh -t -i blog\_project.pem ubuntu@YOUR\_EC2\_IP "sudo docker ps"

Expected output should show container ‘blog\_web’ running and port 80 mapped to 8000.

## Step 6: Check Container Logs

ssh -t -i blog\_project.pem ubuntu@YOUR\_EC2\_IP "sudo docker logs blog\_web"

Look for “FORCING S3 STORAGE BACKEND” message without errors.

## Accessing Your Blog

1. Open your web browser
2. Navigate to: http://YOUR\_EC2\_IP
3. You should see the blog post listing page

## Troubleshooting

If you see Nginx default page: 1. Clear browser cache 2. Try accessing explicitly with port: http://YOUR\_EC2\_IP:80 3. Check container logs for errors:

ssh -t -i blog\_project.pem ubuntu@YOUR\_EC2\_IP "sudo docker logs blog\_web"

## Expected Results

1. Web interface shows list of blog posts
2. Images from S3 load properly
3. No error messages in container logs

## Configuration Details

* Container port mapping: 80:8000
* S3 bucket: coursera-bucket3
* Region: us-east-1
* Image: Uses Django with Gunicorn
* Storage: AWS S3 for static and media files

## Security Notes

1. AWS credentials are pulled automatically from local configuration
2. No credentials are exposed in command history
3. Container runs with proper port mappings
4. EC2 security group must allow inbound traffic on ports 22 and 80