AWS S3 Setup Guide (ADVANN Project)

Prepared for: Sandip Mondal
Date: 15 Feb 2026

1. Objective

The objective of this setup is to store product images securely in AWS S3 and access them from a backend service (Spring Boot) for the ADVANN e-commerce project. This document also includes common AWS S3 interview questions with professional answers.

2. Steps Followed: Create AWS S3 Bucket

1. **Open AWS Console** and search for **S3**.
2. Go to **Buckets** section and click **Create bucket**.
3. Set the **AWS Region** as **ap-south-1 (Mumbai)** (recommended for India-based application).
4. Select **Bucket type** as **General purpose**.
5. Enter bucket name as **advann-product-images** (or a unique variant such as **advann-product-images-2026**).
6. Keep **Object Ownership** as **ACLs disabled (recommended)**.
7. If public image access is required, disable **Block all public access** and confirm acknowledgement.
8. Keep **Bucket Versioning** disabled to reduce cost (unless version history is required).
9. Keep **Default encryption** enabled using **SSE-S3 (Amazon S3 managed keys)**.
10. Keep **Object Lock** disabled.
11. Click **Create bucket**.

3. Steps Followed: Create IAM User for S3 Access

1. Go to AWS Console and search for **IAM**.
2. Navigate to **Users** and click **Create user**.
3. Enter user name as **advann-s3-user**.
4. Do not enable AWS Management Console access (not required for backend integration).
5. Proceed to **Set permissions**.
6. Select **Attach policies directly**.
7. Search and select **AmazonS3FullAccess** policy (AWS managed policy).
8. Click **Next** and proceed to **Review and create**.
9. Click **Create user**.

4. Steps Followed: Generate Access Key & Secret Key

1. Open the created user **advann-s3-user**.
2. Go to **Security credentials** tab.

3. Under **Access keys**, click **Create access key**.
4. Select use case as **Local code** (because Spring Boot app runs on local machine during development).
5. Optional: Provide description tag such as **advann-product-service**.
6. Click **Create access key**.
7. Copy **Access key ID** and **Secret access key**. Download the CSV file for backup.
8. **Important:** Secret access key is shown only once.

5. Spring Boot Configuration (application.yml)

In the Spring Boot project, store bucket details and AWS credentials inside **application.yml**. Example configuration:

```
aws:
s3:
bucket-name: advann-product-images
region: ap-south-1
access-key: AKIAxxxxxxxxxxxxxxxx
secret-key: xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

6. Interview Questions and Answers (AWS S3 + IAM)

Q: What is Amazon S3?

A: Amazon S3 (Simple Storage Service) is an object storage service in AWS used for storing and retrieving files such as images, documents, backups, and videos. It is scalable, durable, and highly available.

Q: What is an S3 bucket?

A: A bucket is a container in Amazon S3 that stores objects. Every object must belong to a bucket.

Q: What is an object in S3?

A: An object is a file stored in a bucket along with metadata. Example: product-image.jpg.

Q: Why did you use S3 in your project?

A: I used S3 to store product images because it is cost-effective, scalable, highly durable, and provides direct URL access.

Q: What is the meaning of Region in S3?

A: Region specifies the AWS location where the bucket is created. Choosing a nearby region reduces latency and improves performance.

Q: What is Block Public Access in S3?

A: Block Public Access is a security feature that prevents buckets and objects from becoming publicly accessible accidentally.

Q: How did you make S3 objects accessible publicly?

A: By disabling Block Public Access and attaching a bucket policy that allows public read access (s3:GetObject).

Q: What is an IAM user?

A: IAM user is an identity in AWS with specific permissions. It is used for programmatic access to AWS services using access keys.

Q: Why did you create an IAM user for your Spring Boot project?

A: To provide controlled programmatic access to S3 without using root credentials. IAM user allows applying least-privilege permissions.

Q: What is an Access Key and Secret Key?

A: Access Key ID and Secret Access Key are credentials used for programmatic authentication when calling AWS APIs.

Q: How do you secure AWS credentials?

A: AWS credentials should never be committed to GitHub. They should be stored in environment variables, secrets manager, or external configuration systems.

Q: What is the purpose of AmazonS3FullAccess policy?

A: It is an AWS managed policy that grants full permissions to S3 operations such as upload, download, delete, and list buckets/objects.

Q: What is a bucket policy?

A: Bucket policy is a JSON-based resource policy attached to an S3 bucket. It defines who can access the bucket and what actions are allowed.

Q: IAM Policy vs Bucket Policy?

A: IAM policy is attached to users/roles/groups, while bucket policy is attached directly to the bucket. Both control access permissions.

Q: What is Versioning in S3?

A: Versioning maintains multiple versions of an object. It helps recover from accidental deletes or overwrites.

Q: What is SSE-S3 encryption?

A: SSE-S3 is server-side encryption managed by AWS. It encrypts objects at rest using AWS-managed keys automatically.

Q: What is a pre-signed URL?

A: A pre-signed URL provides temporary access to a private object. It is generated by backend and expires after a defined time.

Q: How can you improve S3 image performance for users?

A: By using AWS CloudFront CDN in front of S3, which caches images and delivers them faster globally.

Q: How do you avoid duplicate file names in S3 uploads?

A: By generating unique names using UUID or timestamp and storing the object key in the database.

7. Conclusion

This setup enables ADVANN backend services to upload and manage product images using AWS S3. For production environments, it is recommended to use least-privilege policies, private buckets, and CloudFront or pre-signed URLs for controlled access.