Audiobook S3 Bucket and API Gateway Integration

Masuod

April 4, 2024

1 Introduction

In this document, we will explore the setup of an Amazon S3 bucket designed specifically for storing and sharing audiobook files. The S3 bucket acts as a server where these files are stored securely and can be accessed programmatically through API Gateway.

2 S3 Bucket Setup

The S3 bucket named s3://audio-book-bucket was created to store audiobook files. The bucket is configured with appropriate permissions to ensure data security and integrity. Files uploaded to this bucket are organized and accessible based on their unique keys.

3 API Gateway Integration

To interact with the S3 bucket programmatically, an API Gateway was set up. The API Gateway serves as an interface for external applications and services to interact with the audiobook files stored in the S3 bucket. Two main methods were created:

1. **POST Method**: This method allows users to upload audiobook files to the S3 bucket. The API Gateway handles the authentication and forwards the file to the appropriate endpoint in the S3 bucket.

2. **GET Method**: Users can retrieve audiobook files from the S3 bucket using the GET method. The API Gateway validates the request and fetches the requested file from the S3 bucket, providing a secure and controlled access mechanism.

4 Postman Requests

Postman, a popular API testing tool, was used to test the functionality of the API Gateway and S3 bucket integration. The following requests were made:

1. **POST Request**: An audiobook file was uploaded to the S3 bucket using the POST method. The request included the necessary authentication credentials and file data.

2. **GET Request**: A GET request was made to retrieve an audiobook file from the S3 bucket. The request specified the file's unique identifier, and upon successful authentication, the file was returned by the API Gateway.

5 Request Response Template

Below is a template illustrating the structure of request and response for both POST and GET methods:

5.1 POST Request Template

Request:

```
POST /upload-audiobook HTTP/1.1
Host: api-gateway-url.com
Authorization: Bearer your-auth-token
Content-Type: multipart/form-data
```

Response:

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "message": "File uploaded successfully."
}
```

5.2 GET Request Template

Request:

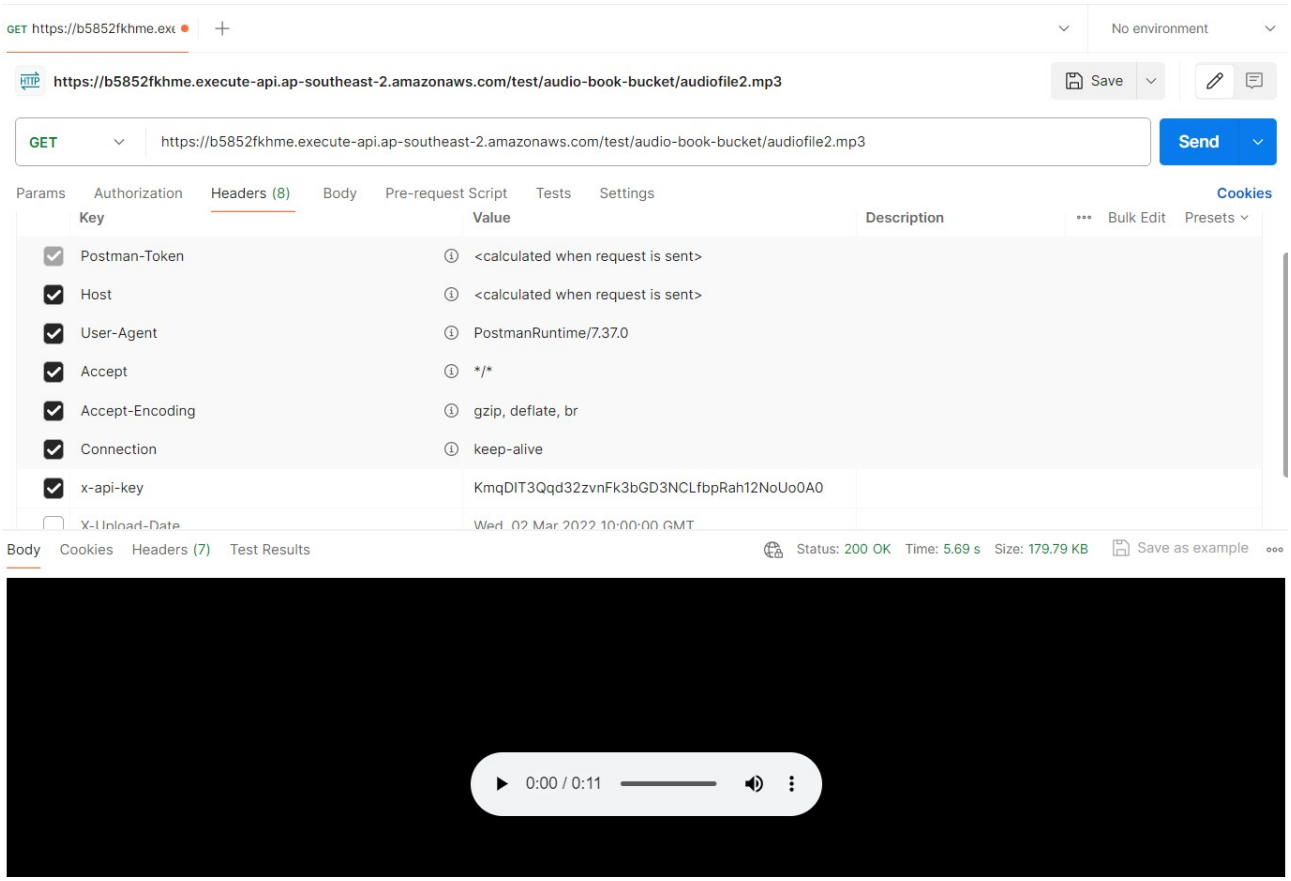
```
GET /download-audiobook?fileId=your-file-id HTTP/1.1
Host: api-gateway-url.com
Authorization: Bearer your-auth-token
```

Response:

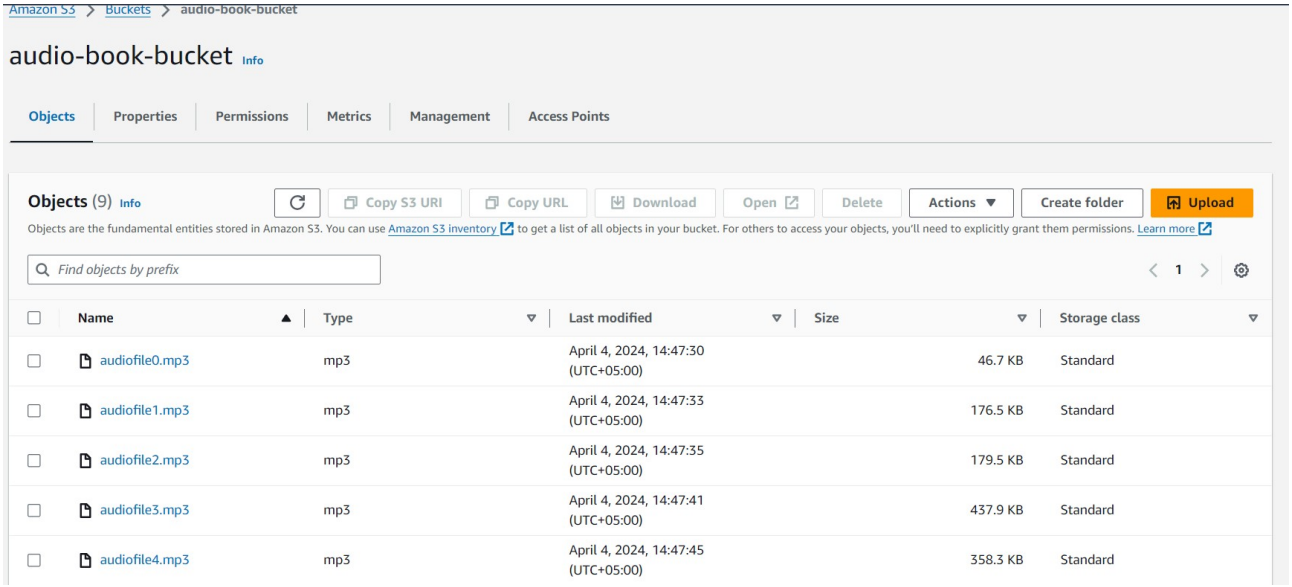
```
HTTP/1.1 200 OK
Content-Type: application/octet-stream
```

6 Bucket Contents and Request Responses

6.1 Postman Request Example



6.2 S3 Bucket Contents



7 Conclusion

The combination of an Amazon S3 bucket and API Gateway provides a robust and scalable solution for managing audiobook files. The secure storage and controlled access mechanisms

ensure data confidentiality and integrity, making it suitable for various audiobook sharing and distribution applications.