**Steps for Lambda pulling data from AWS ElasticSearch and Uploading Data to S3**

1. **Account B (Bucket Owner):**
   1. **Create an S3 Bucket**: If you haven't already, create an S3 bucket in Account B where you want to allow uploads from Account A.
   2. **Define Bucket Policy**: Create a bucket policy for the S3 bucket in Account B that allows the Lambda function from Account A to upload objects. The policy should include a **Principal** element referencing the IAM role in Account A.

**Example**: {

"Version": "2012-10-17",

"Statement": [

{

"Effect": "Allow",

"Principal": {

"AWS": "arn:aws:iam::ACCOUNT-A-ID:role/lambda-role-in-account-a"

},

"Action": "s3:PutObject",

"Resource": "arn:aws:s3:::bucket-in-account-b/\*"

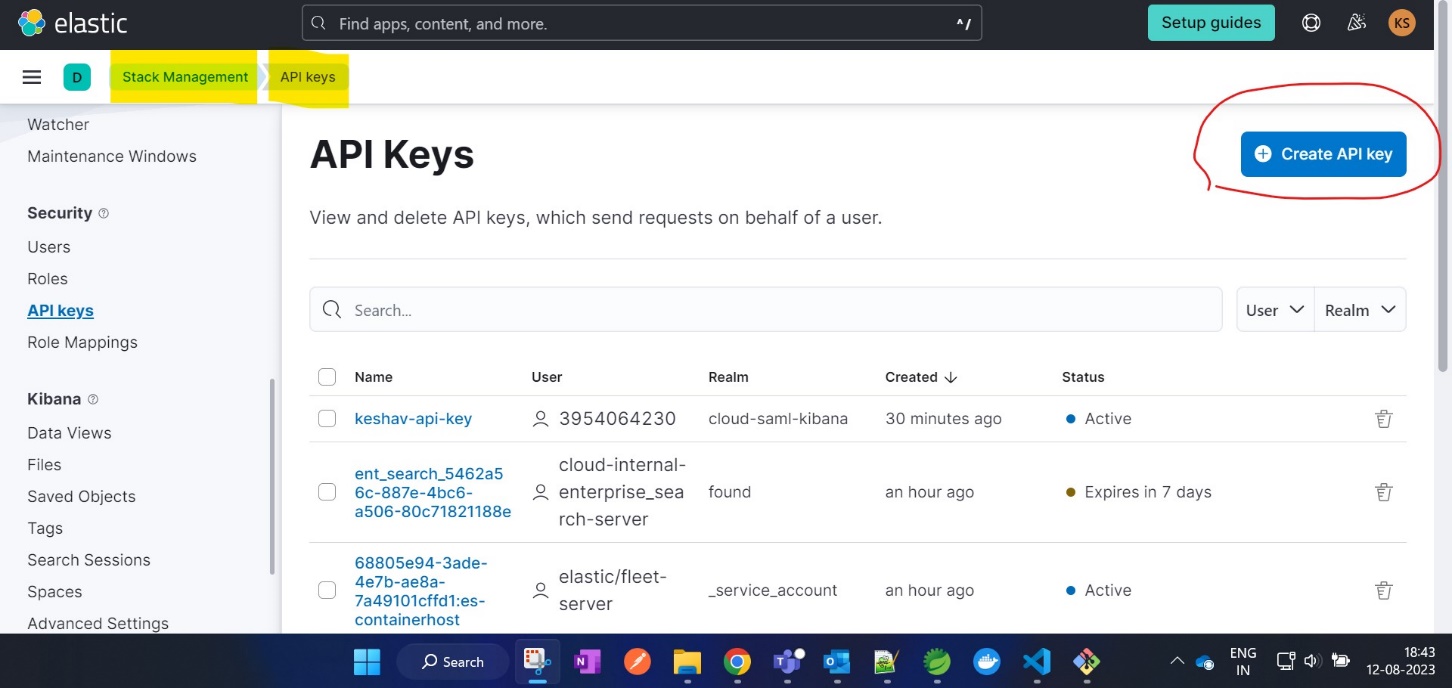
}

]

}

**Note:** Replace ACCOUNT-A-ID with the AWS account ID of Account A and lambda-role-in-account-a with the IAM role name in Account A.

1. **Account C (ElasticSearch):**
   1. **ElasticSearch API Key:** Create API Key to be used in Lambda for pulling data from ElasticSearch.



1. **Account A (Lambda Executor):**
   1. **Create IAM Role**: Create an IAM role in Account A that the Lambda function will assume to perform the S3 upload. This role should have permissions to assume the cross-account role in Account B.
   2. **Assume Cross-Account Role**: Attach a trust policy to the IAM role in Account A that allows it to assume the cross-account role in Account B.

**Example**: {

"Version": "2012-10-17",

"Statement": [

{

"Effect": "Allow",

"Principal": {

"Service": "lambda.amazonaws.com"

},

"Action": "sts:AssumeRole",

"Condition": {

"StringEquals": {

"sts:ExternalId": "external-id"

}

}

}

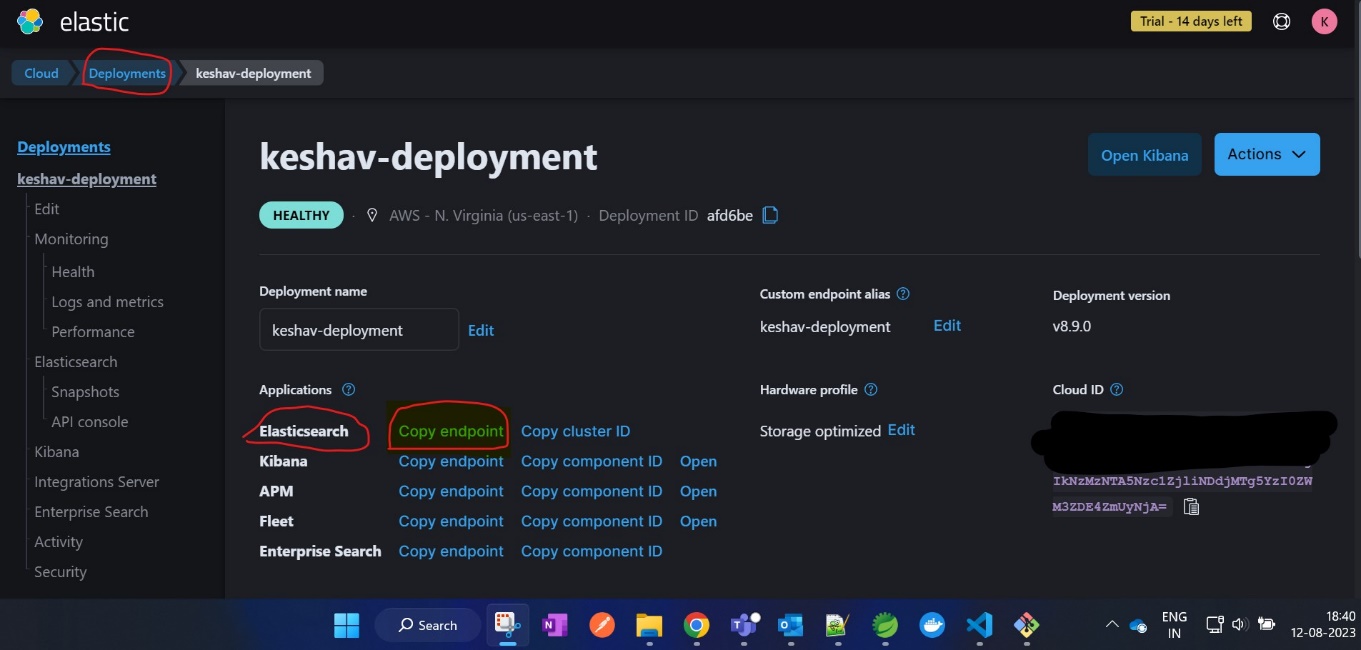
]

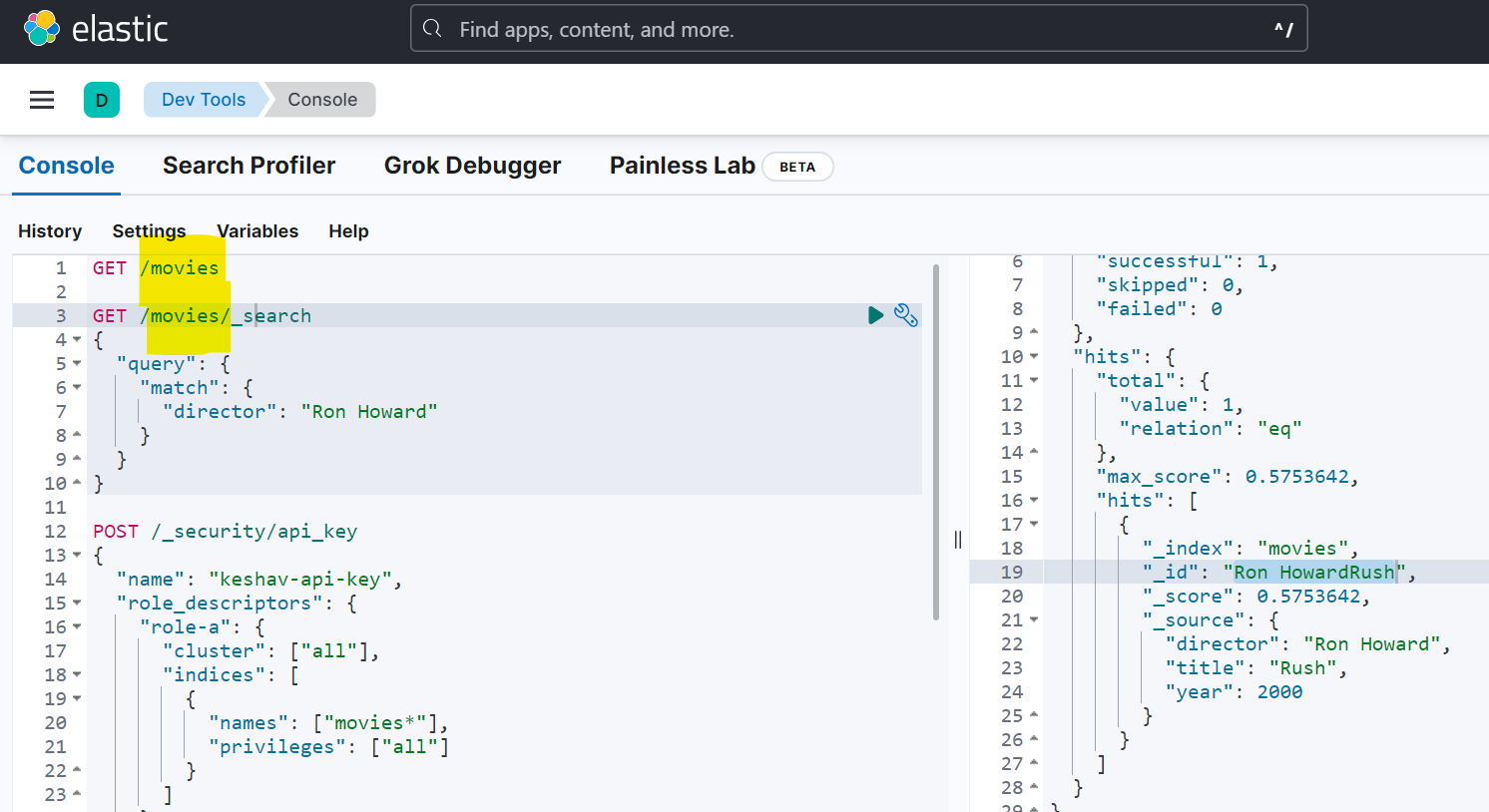
}

**Note:** Replace **"external-id"** with the external ID that you set when you created the cross-account role in Account B.

1. **Lambda Code property changes**
   1. **BeanConfiguration.java:**
      1. aws.elasticsearch.server.url // Copy Endpoint going to the Deployment as shown below from Account C ElasticSearch UI.

**Example:** https://keshav-deployment.es.us-east-1.aws.found.io for below setup.



* + 1. aws.region // Region in which Account B S3 bucket is created. **Example:** us-east-1 for above setup.
  1. **ElasticSearchService.java:**
     1. aws.elasticsearch.index.name 
  2. **S3Service.java:**
     1. aws.s3.elasticsearch.data.file.name // Expected filename in S3.
     2. aws.s3.elasticsearch.data.bucket.name // Expected S3 bucket in which file needs to be uploaded & Lambda is provided access via S3 Bucket Policy in Account B.

1. **Deploy Lambda to AWS Account A**