**Deploying a Java Spring Boot application on AWS Elastic Beanstalk involves several steps.**

**Step 1: Prepare the Spring Boot Application**

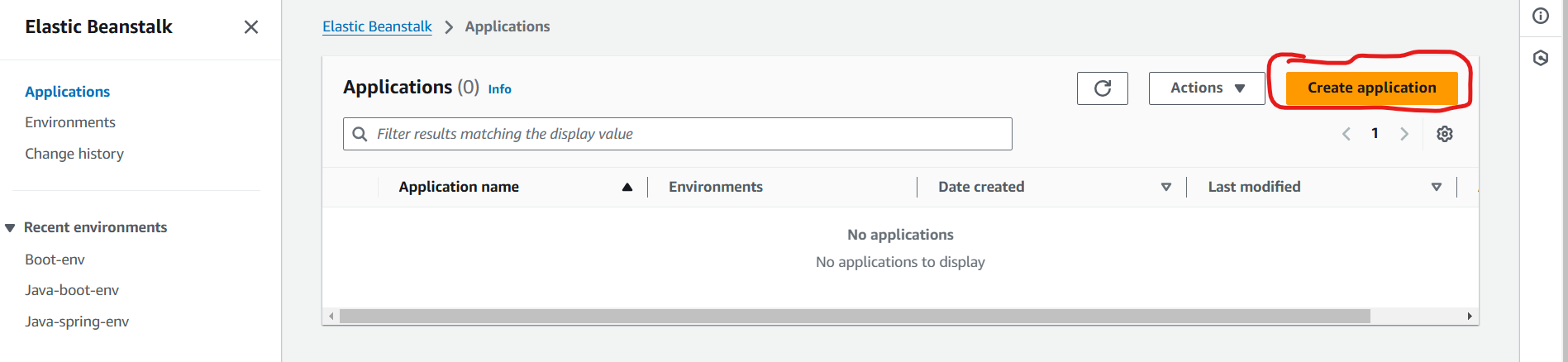
1. **Build the Application: Ensure your Spring Boot application is packaged as a .jar file. You can do this by running:**

**mvn clean package**

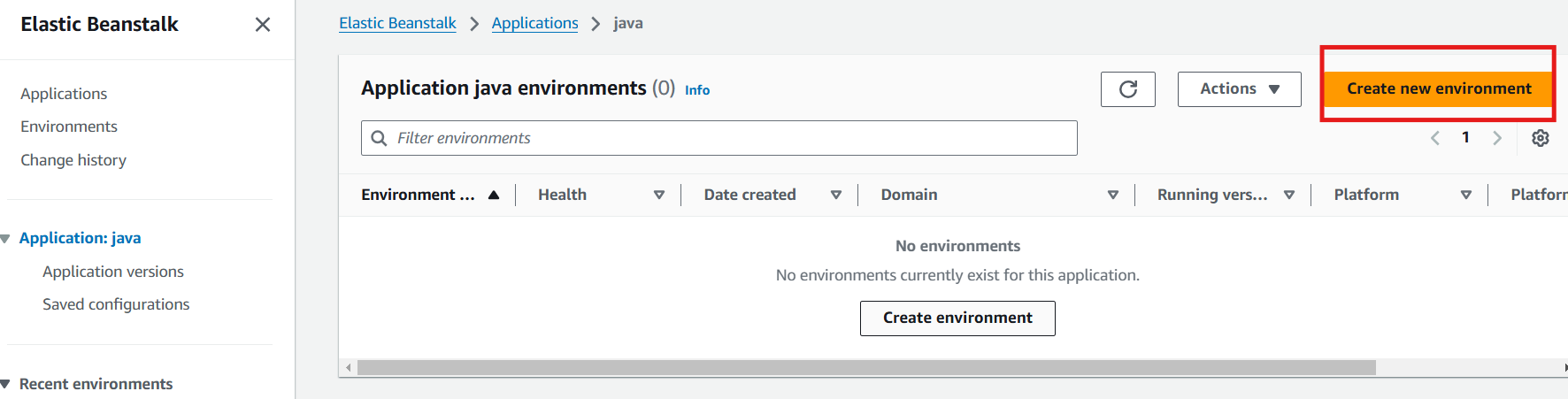
**This command will create a JAR file in the target directory (e.g., target/yourapp.jar).**

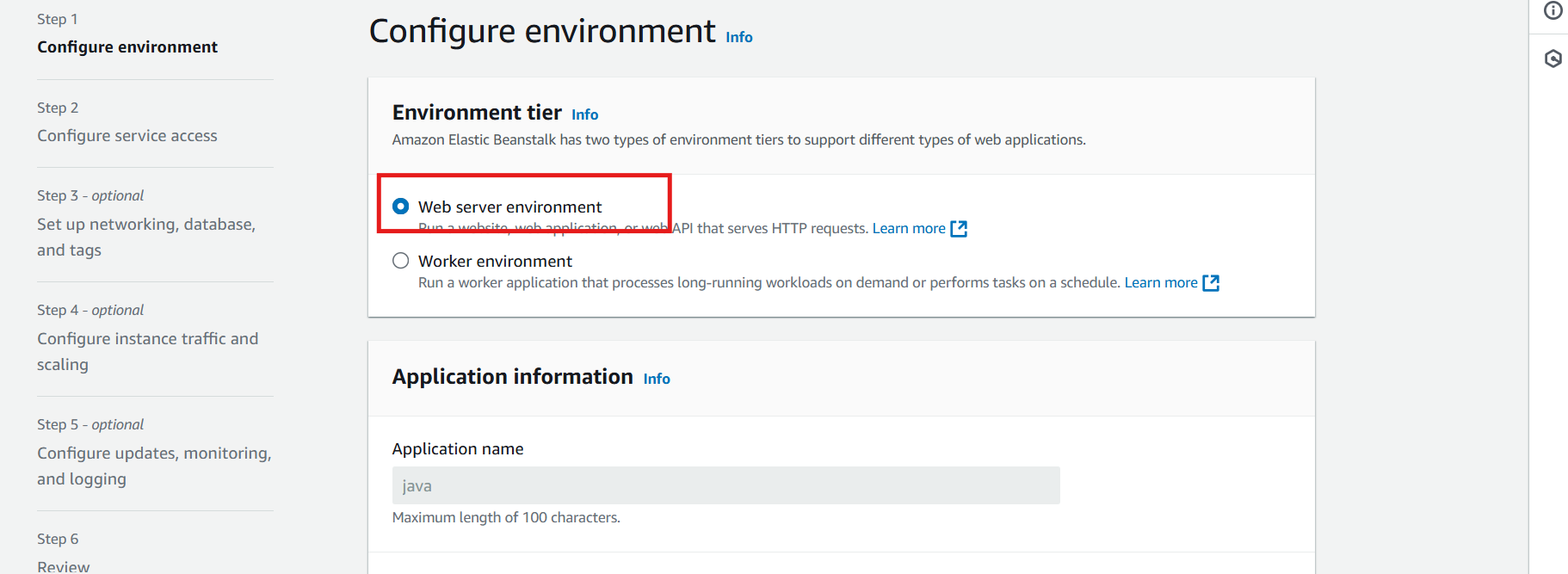
**Step 2: Set Up Elastic Beanstalk Environment**

1. **Login to AWS Console: Open the** [**AWS Management Console**](https://aws.amazon.com/console/) **and navigate to the Elastic Beanstalk service.**
2. **Create a New Application:**
   * **Click on Create Application.**
   * **Give your application a name and select Java as the platform.**

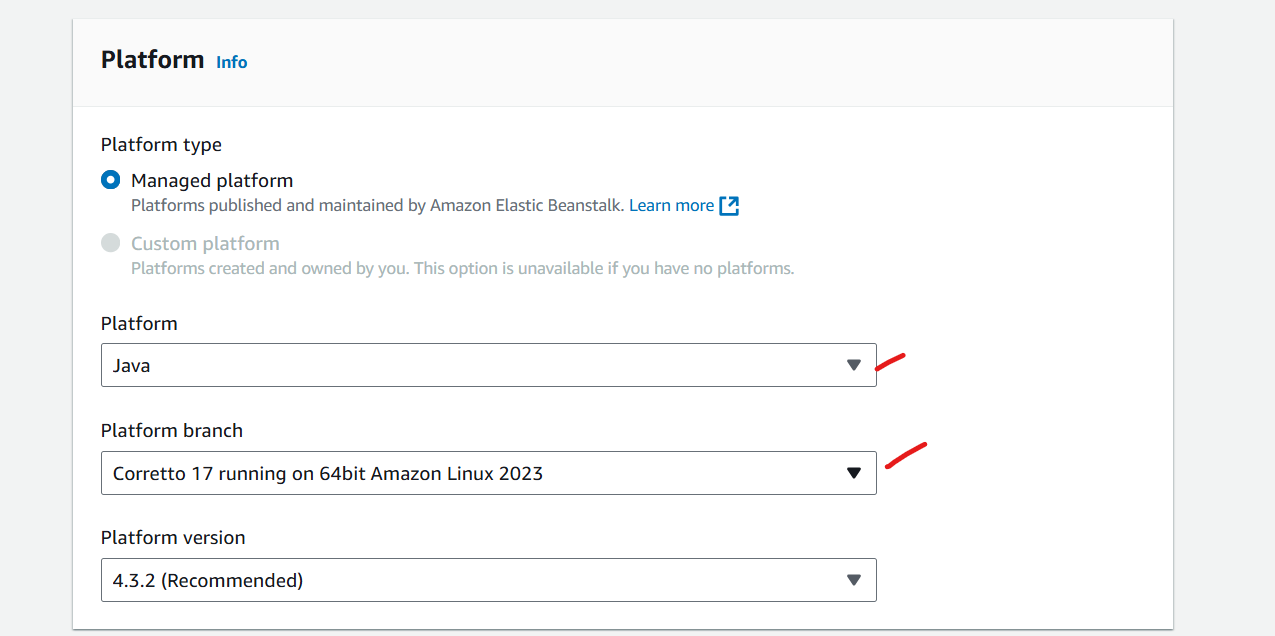
****

1. **Configure Environment:**

****

****

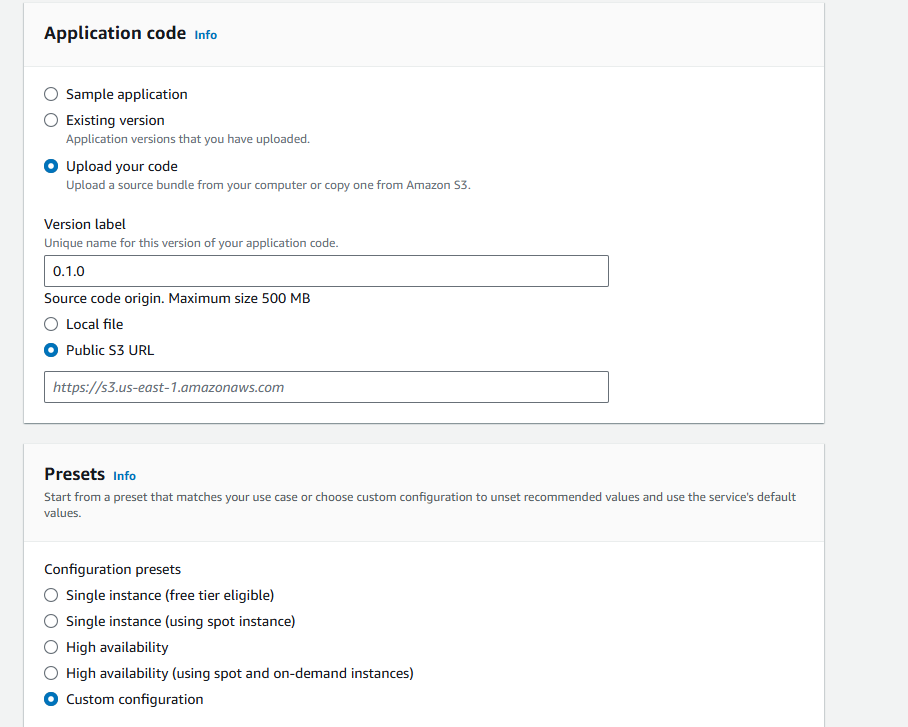
* + **In the Platform Branch, select a version compatible with your Java application (e.g., Java 17 running on 64bit Amazon Linux 2).**

****

* + **Configure additional settings like instance type, environment type (Load balancing or single instance), and key pair if you need SSH access to the instance.**

**Step 3: Deploy the Application**

1. **Upload and Deploy:**
   * **In the Application code section, choose Upload your code.**
   * **Upload the .jar file from the target directory.**
   * **Click on Deploy.**

****

**Configure service access**

**Set Up the Service Role**

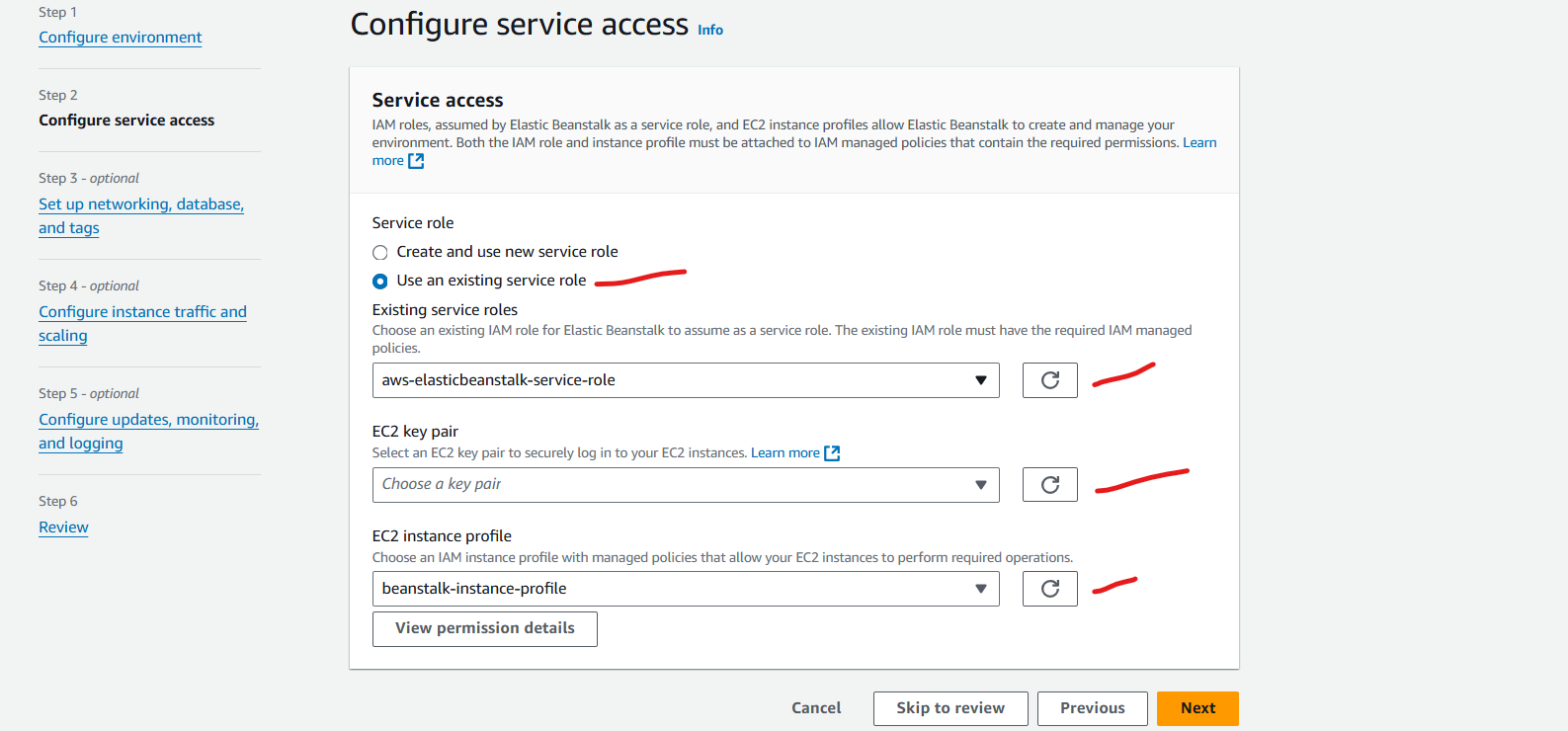
1. **In the Elastic Beanstalk Console:**
   * **Go to Configuration > Security.**
2. **Service Role:**
   * **Choose Create and use a new service role or Use an existing service role if you already have one set up.**
   * **If using an existing role, select aws-elasticbeanstalk-service-role from the Existing service roles list. Make sure it has the necessary permissions attached (e.g., AWSElasticBeanstalkService policy).**
3. **View Permissions:**
   * **You can click on View permission details to see the exact policies attached to this role, which typically includes permissions to manage and monitor Elastic Beanstalk environments and related resources.**

**Step 2: Select an EC2 Key Pair**

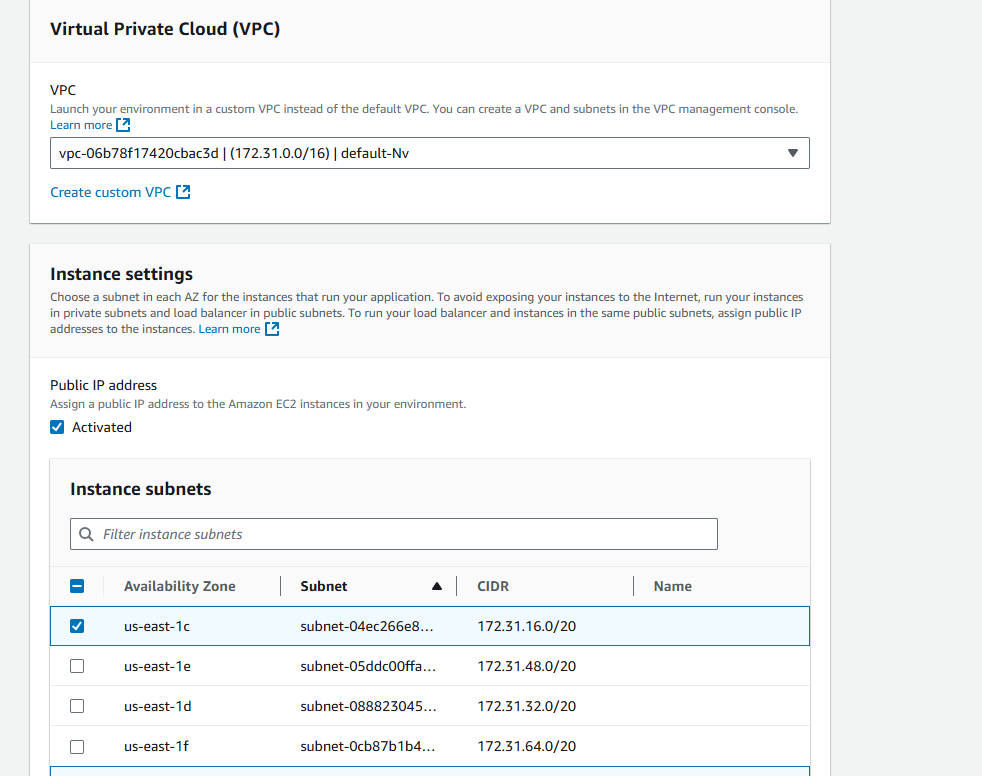
1. **EC2 Key Pair:**
   * **In the EC2 key pair section, select an existing key pair or create a new one if you don’t have one. This key pair is used to securely SSH into your EC2 instances.**
   * **To create a new key pair:**
     + **Go to the** [**EC2 Dashboard**](https://console.aws.amazon.com/ec2/)**.**
     + **Navigate to Key Pairs > Create Key Pair.**
     + **Name your key pair and download it to your system.**
   * **Back in Elastic Beanstalk, select the newly created key pair.**

**Step 3: Configure the EC2 Instance Profile**

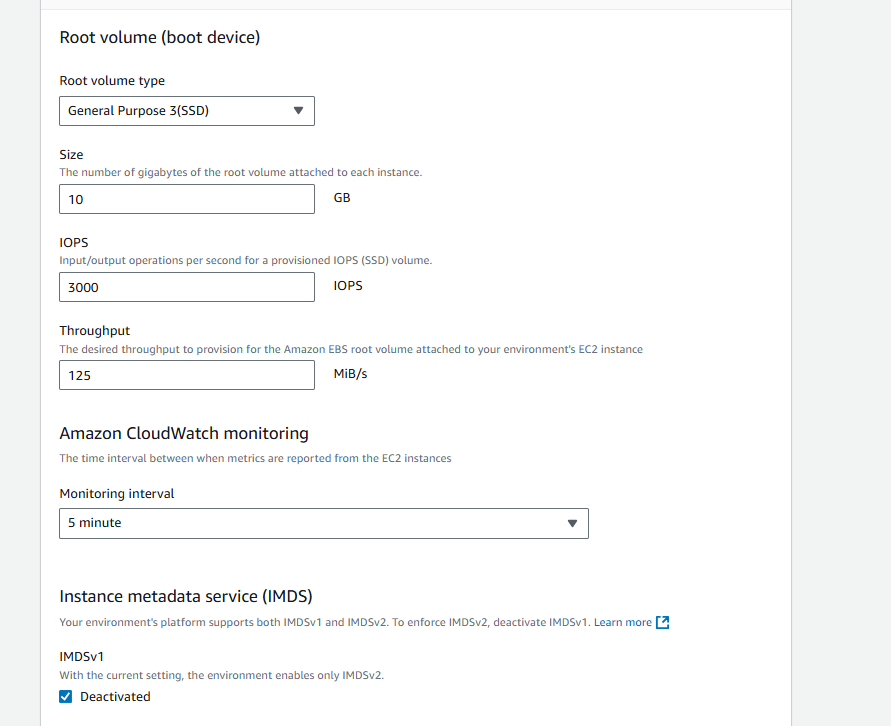
1. **EC2 Instance Profile:**
   * **Choose beanstalk-instance-profile as the EC2 instance profile. This profile should have managed policies allowing EC2 instances to perform required operations, such as AWSElasticBeanstalkWebTier and AWSElasticBeanstalkWorkerTier.**
2. **View Permissions:**
   * **Click View permission details to see attached policies and ensure they have the necessary permissions, allowing the instance to interact with AWS services such as S3, CloudWatch, and EC2.**

****

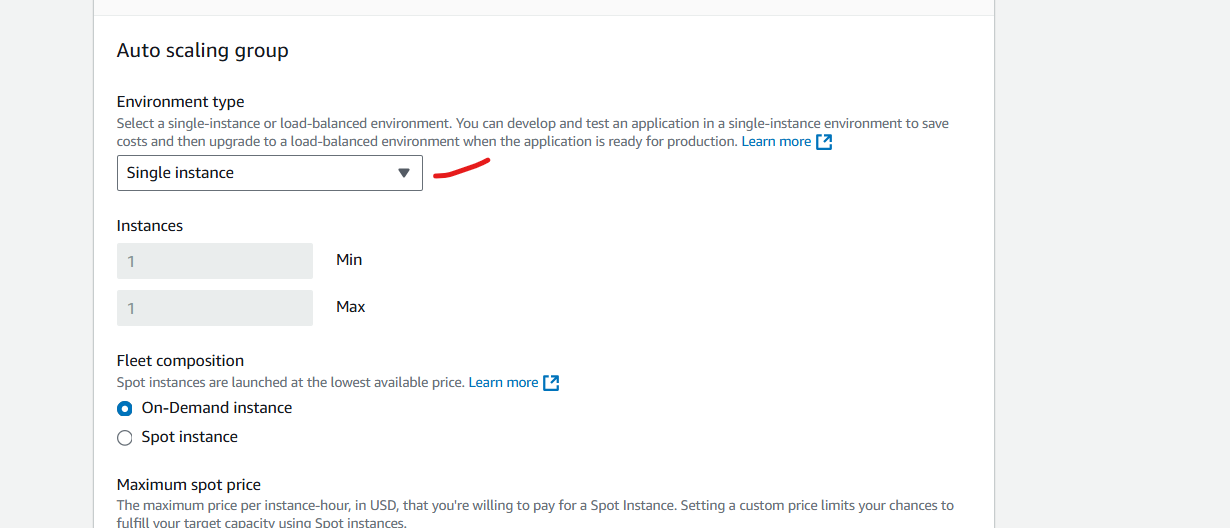
**Step 4: Configure Virtual Private Cloud (VPC)**

****

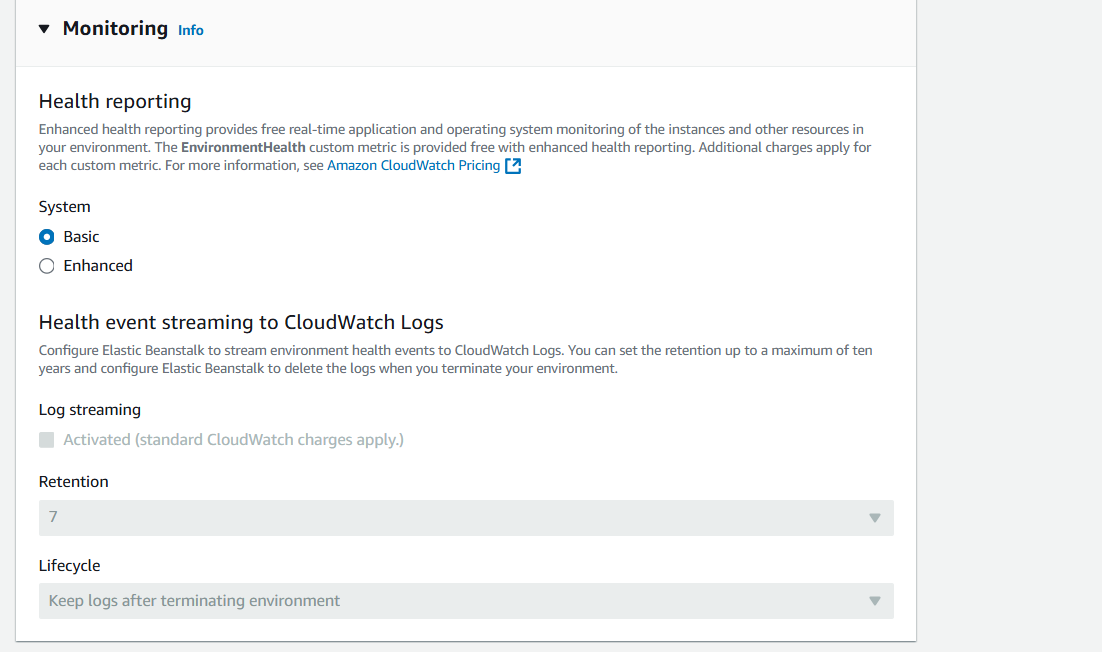
**Steps: Configure Instance Settings**

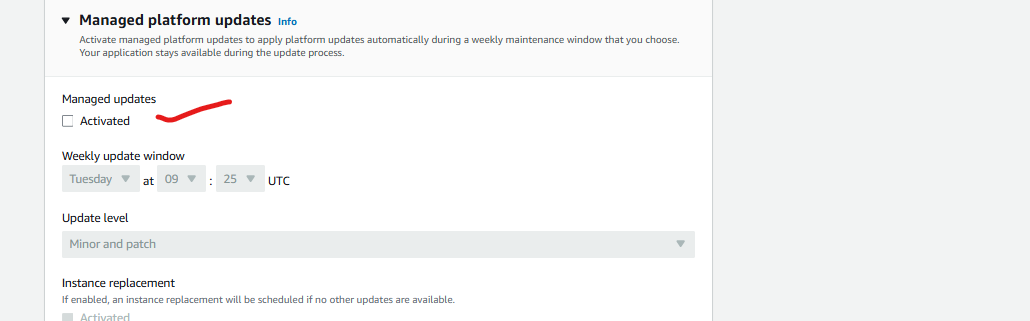
****

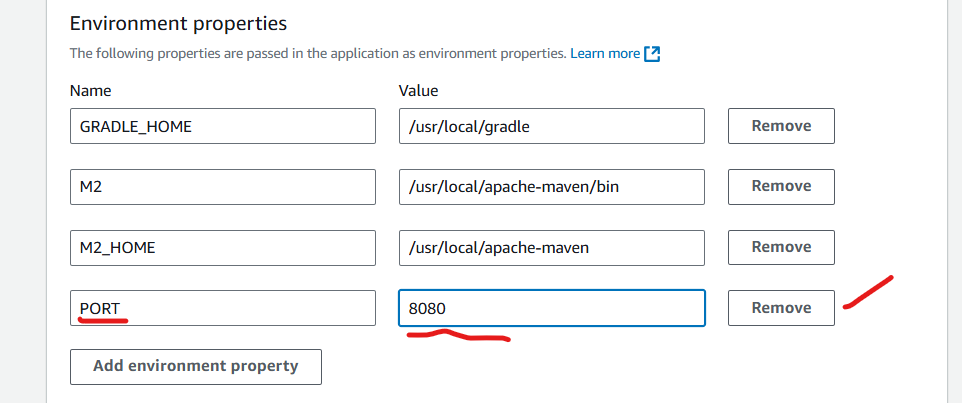
**Monitor Deployment: Elastic Beanstalk will start provisioning resources like EC2, Auto Scaling, Load Balancer, and other necessary AWS services. This process may take a few minutes.**

****

**Configure updates, monitoring, and logging**

****

****

****