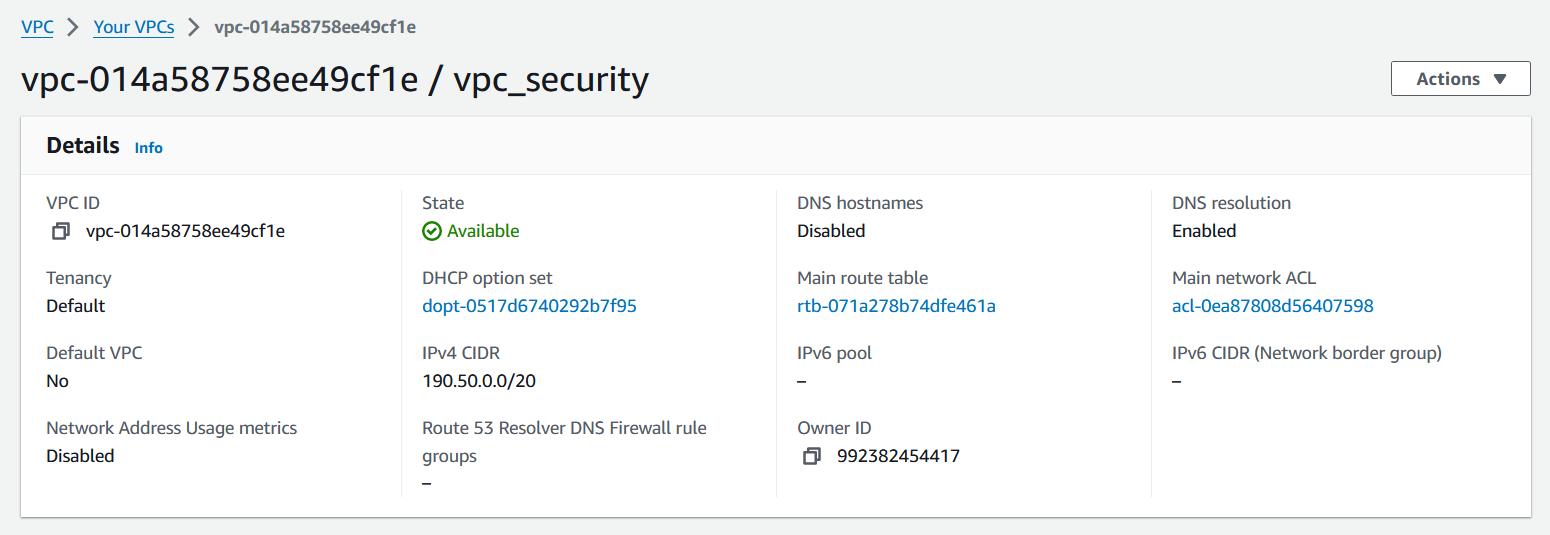
**Implement highly secured infrastructure and configure monitoring with alerts - Assignment**

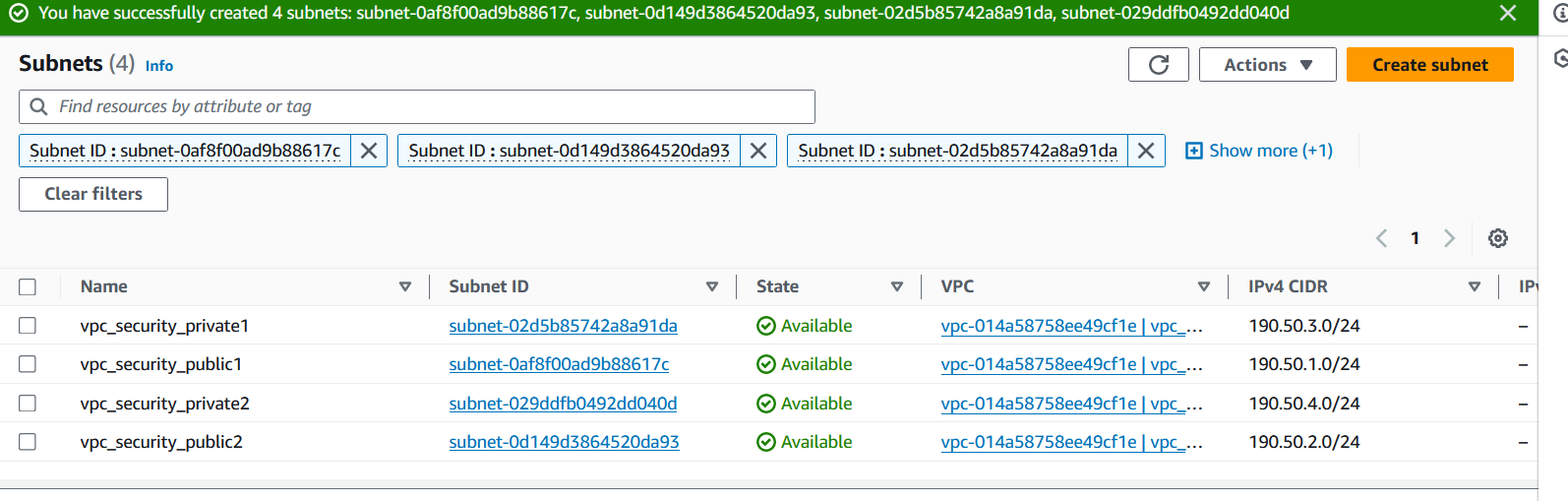
Below are step-by-step instructions for implementing a highly secured infrastructure on AWS, including EC2 instances, EBS volumes, Autoscaling, AWS WAF (Web Application Firewall), AWS Key Management Service (KMS), and AWS Security Hub for monitoring with alerts. This documentation assumes you have an AWS account and basic familiarity with AWS services.

**Step 1: Set Up AWS VPC (Virtual Private Cloud)**

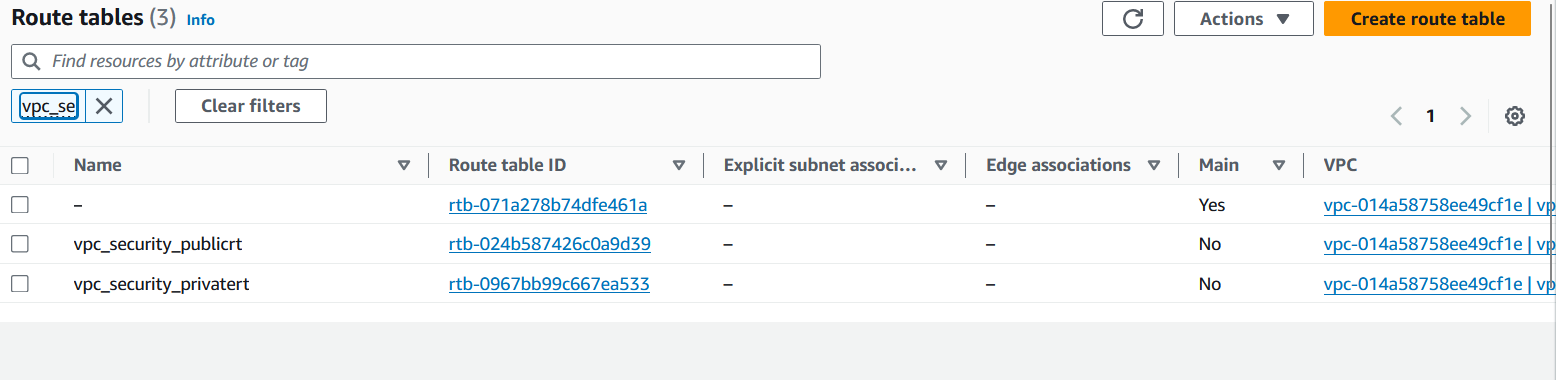
1. Navigate to the AWS Management Console and open the VPC dashboard.
2. Create a new VPC with appropriate CIDR block, ensuring that it's appropriately segmented.



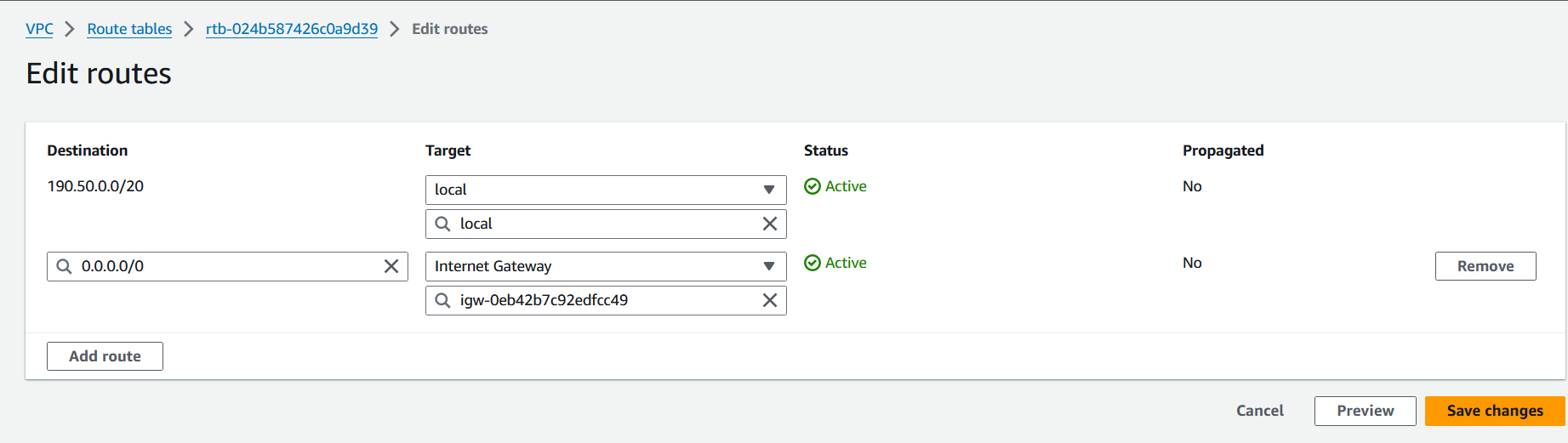
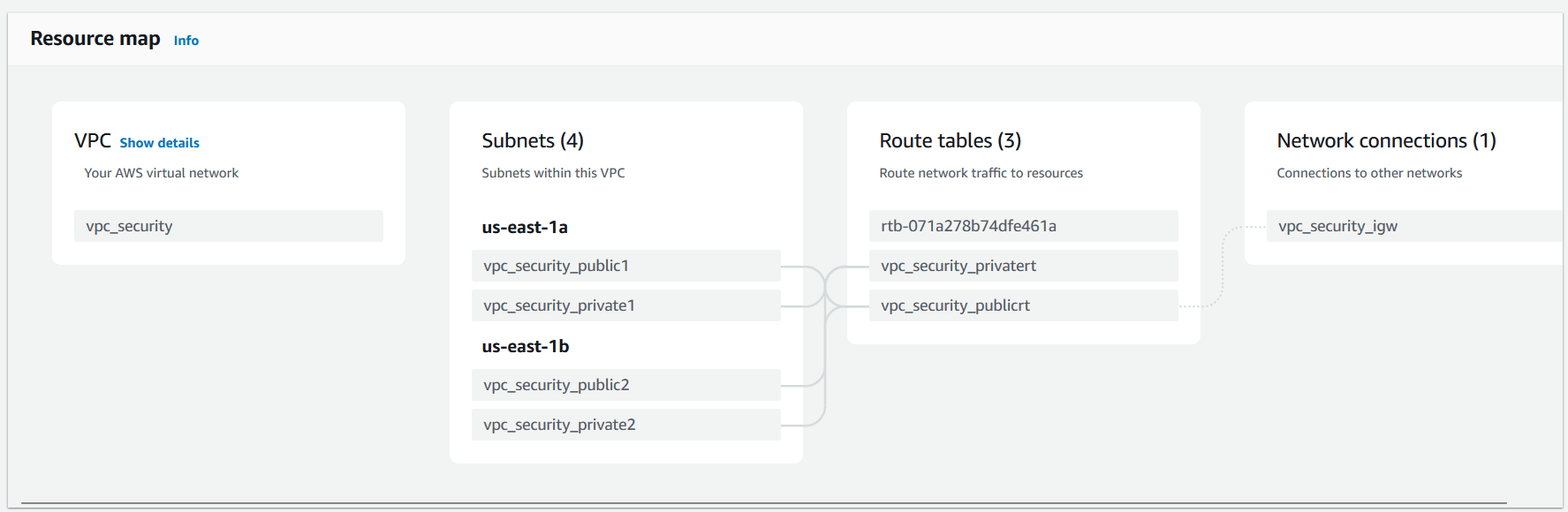
1. Create public and private subnets within the VPC.



1. Set up two Route tables and associate with public and private subnets

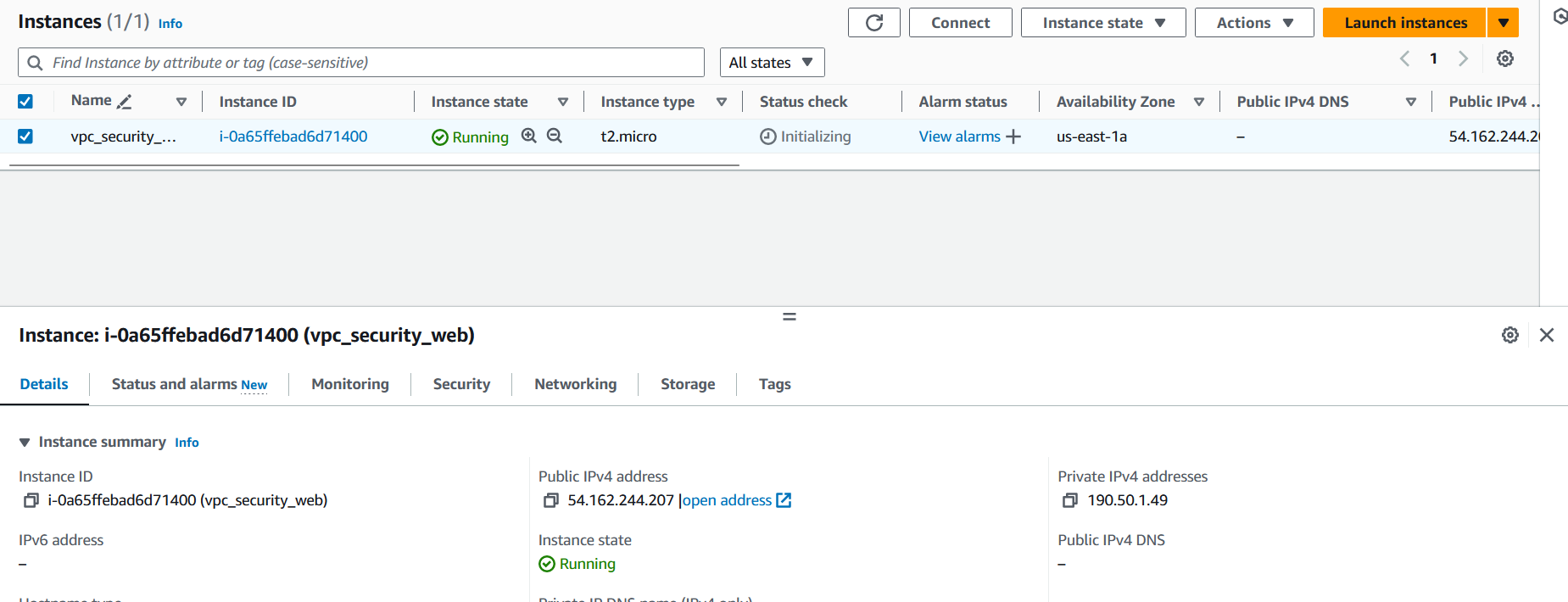


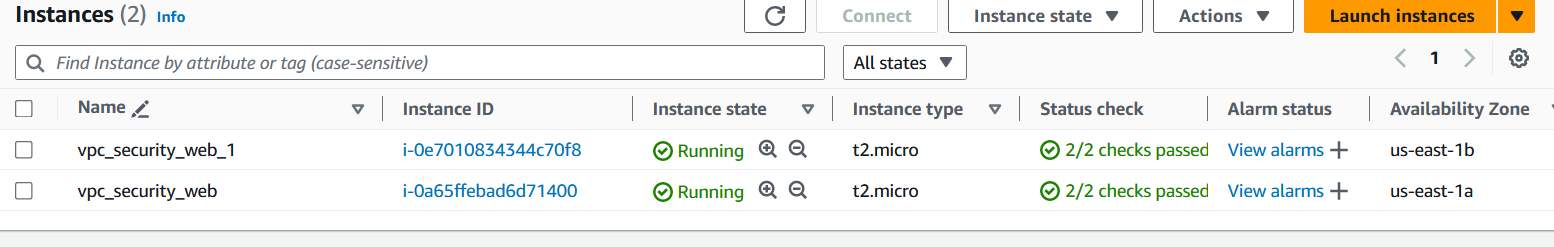
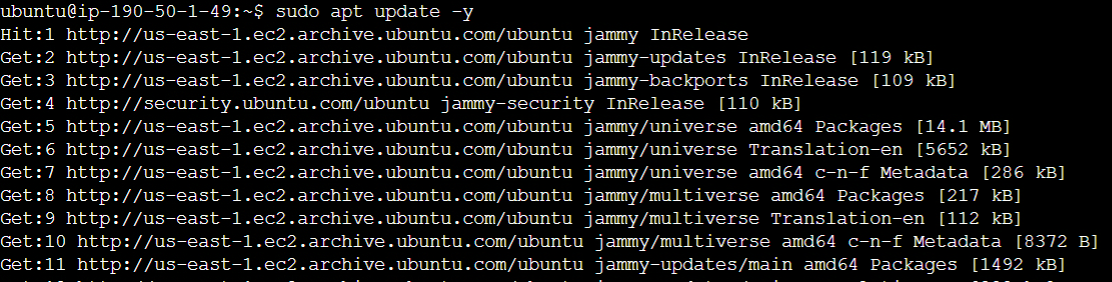
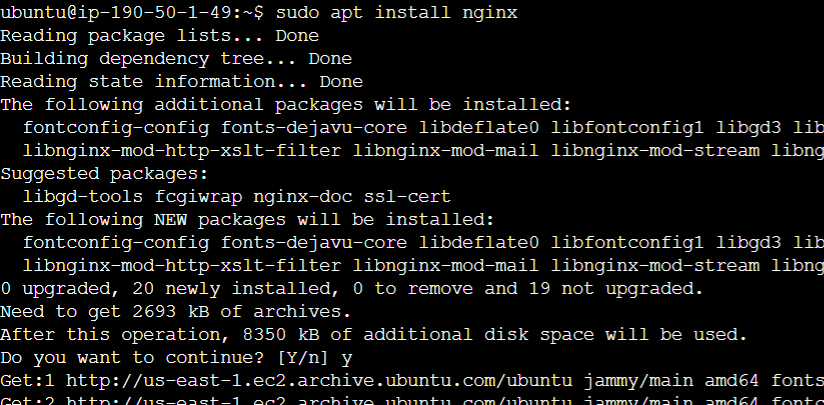
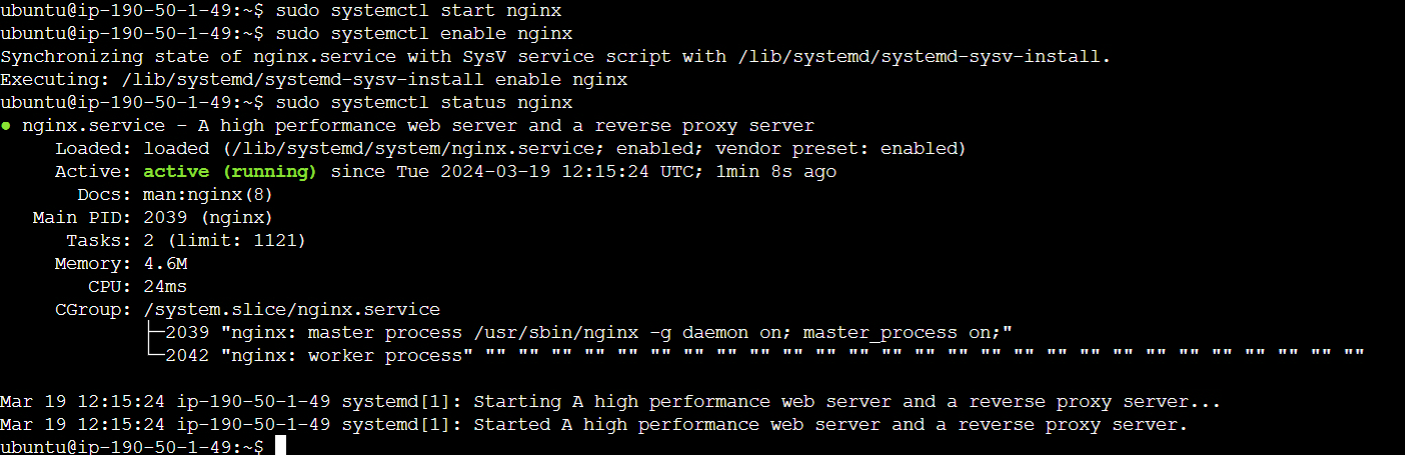
1. Set up Internet Gateway (IGW) and attach it to the VPC for public subnet internet access.

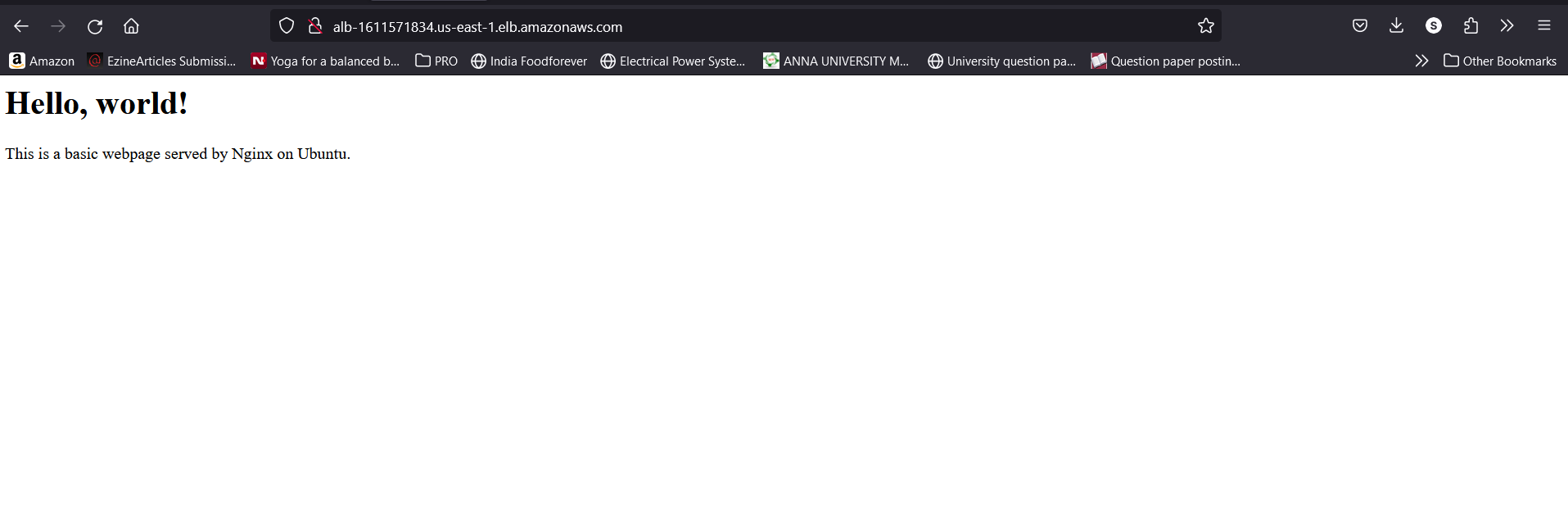
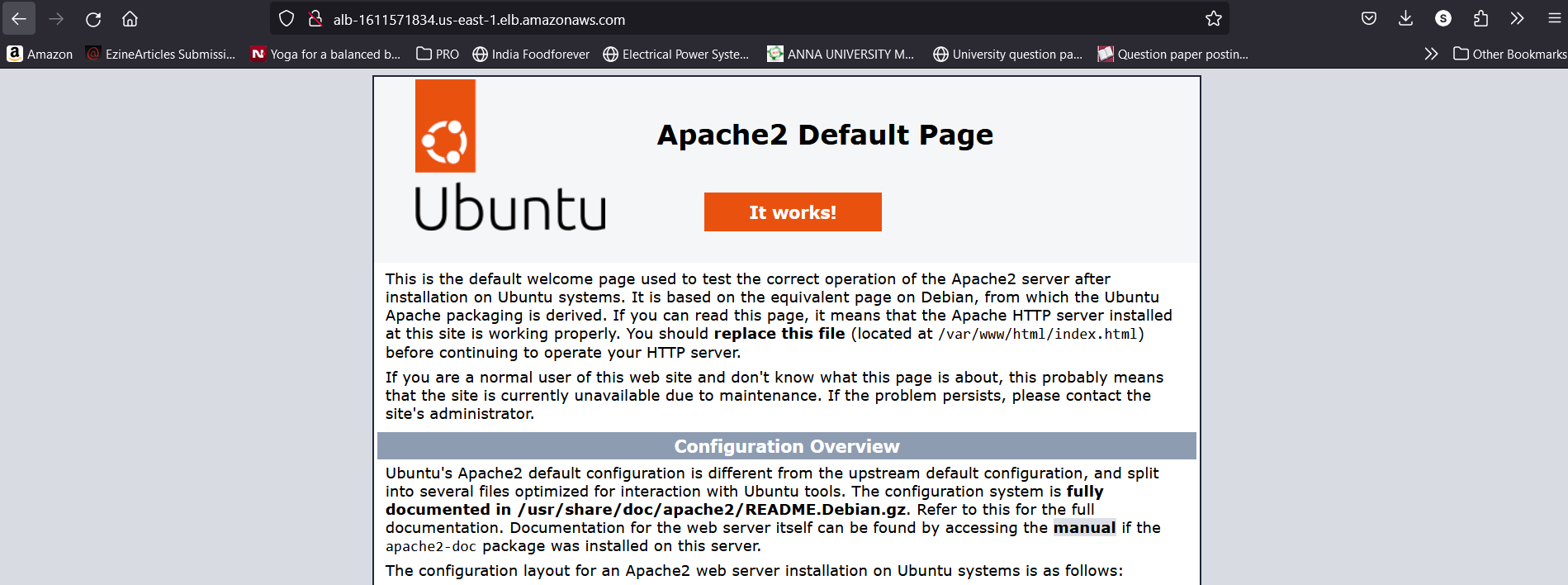
**Step 2: Set Up EC2 Instances**

1. Launch EC2 instances within the public subnet.



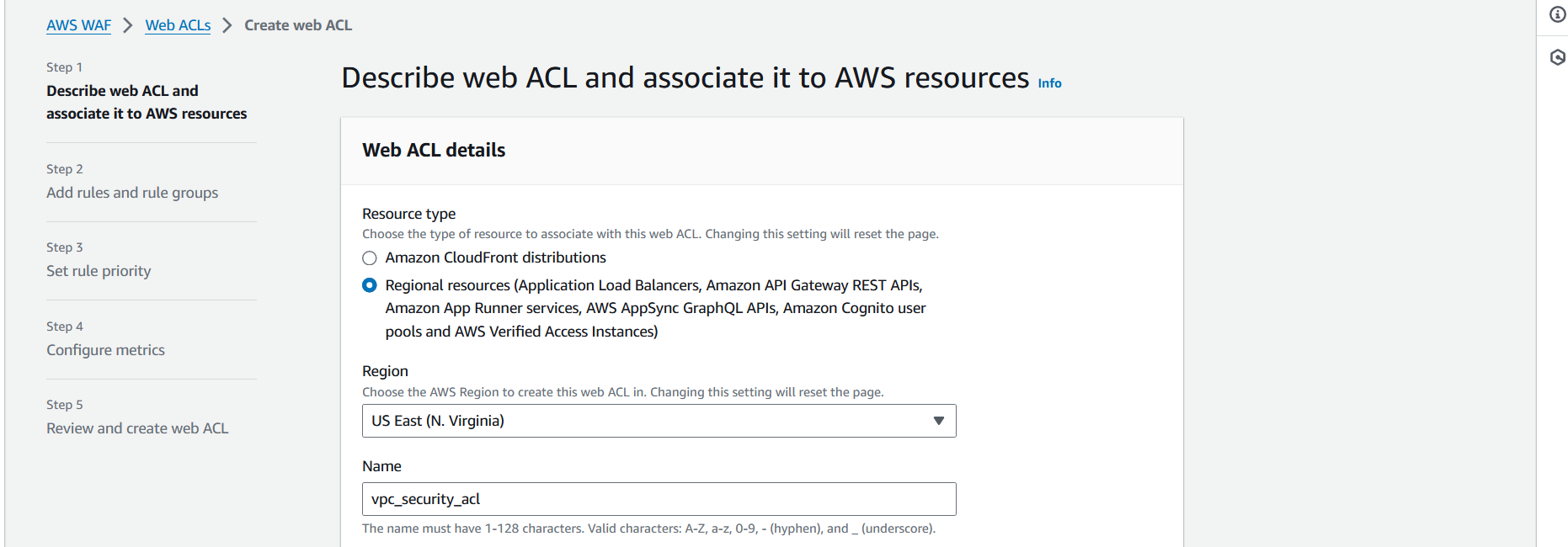
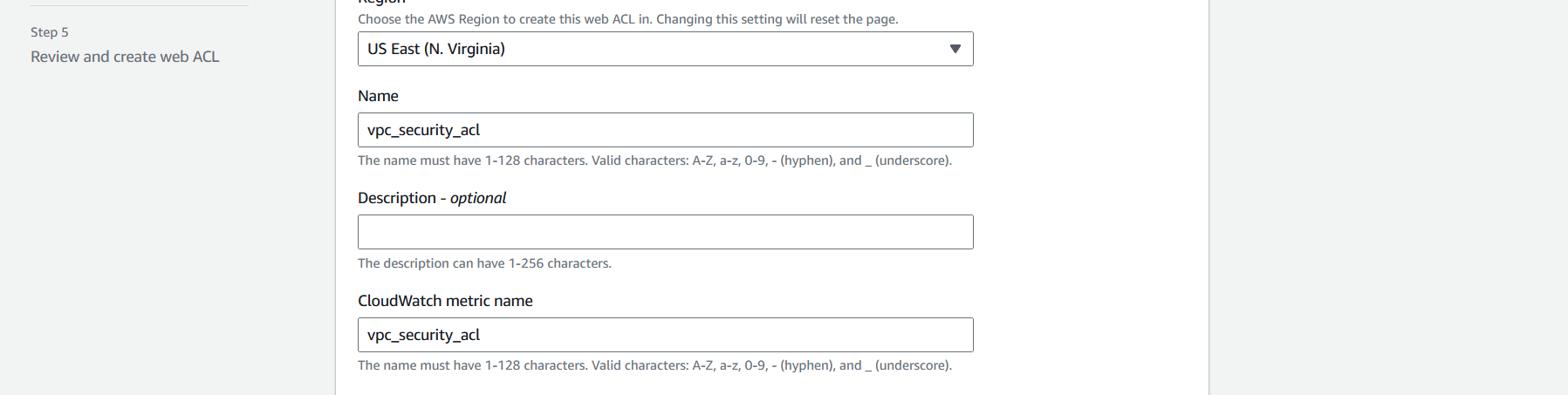
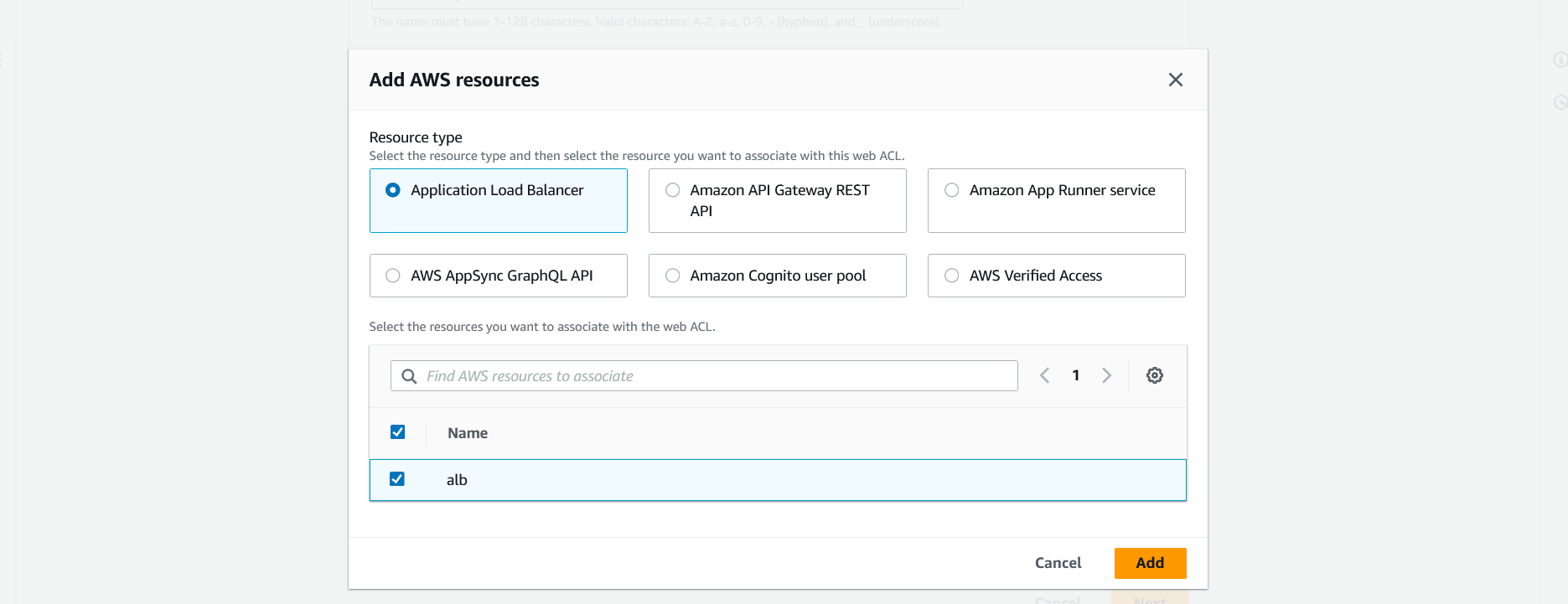
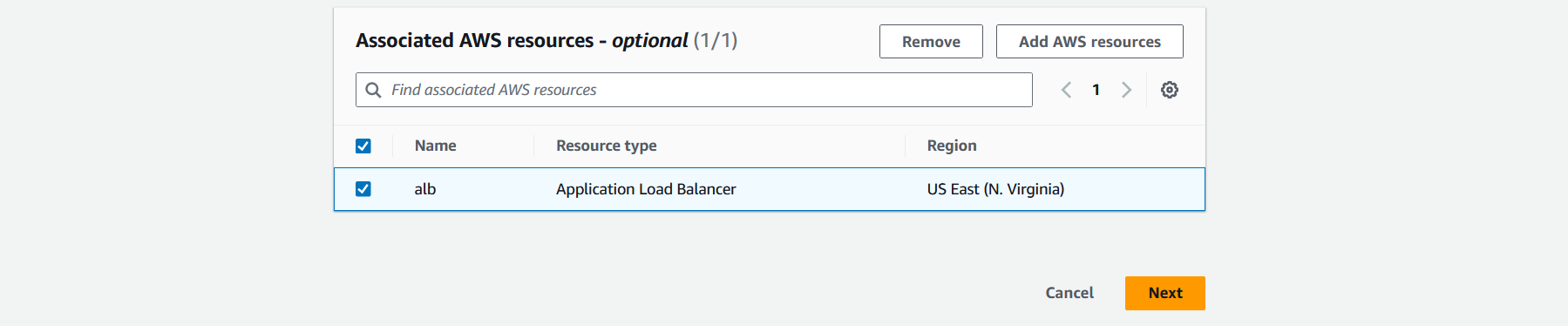
   

1. Configure Elastic Load Balancer (ELB) for distributing traffic across EC2 instances.

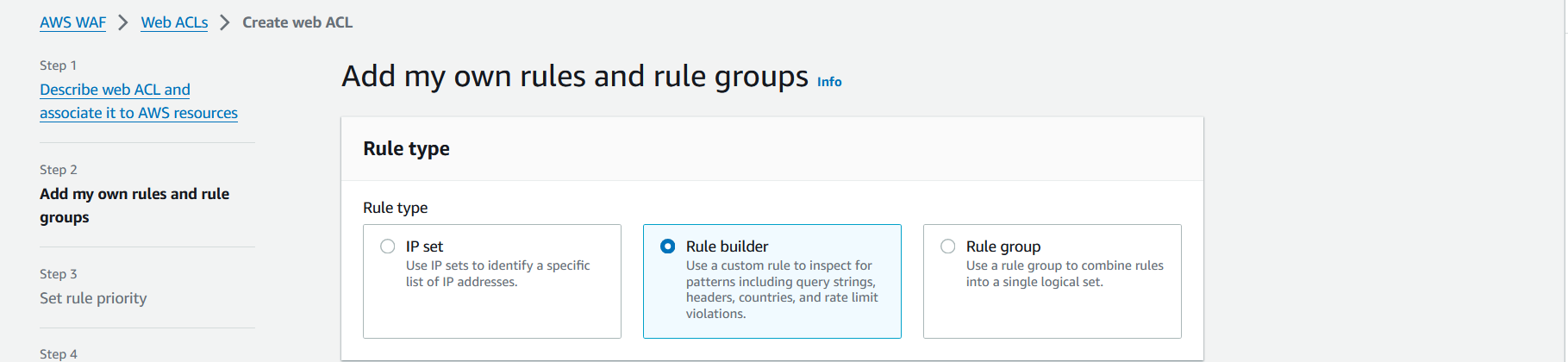
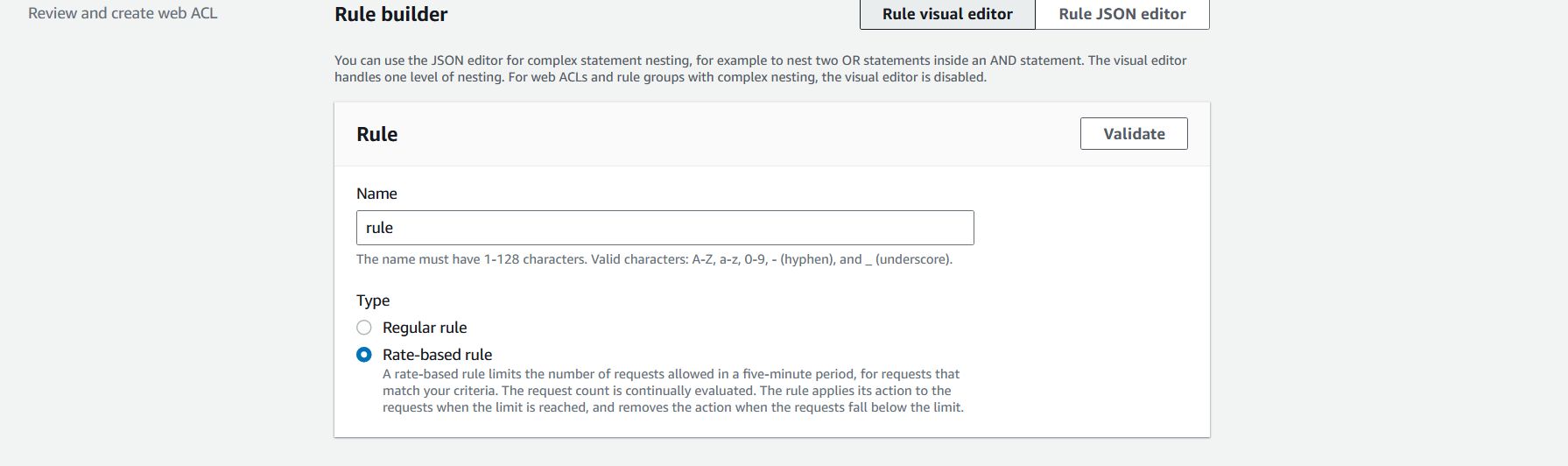
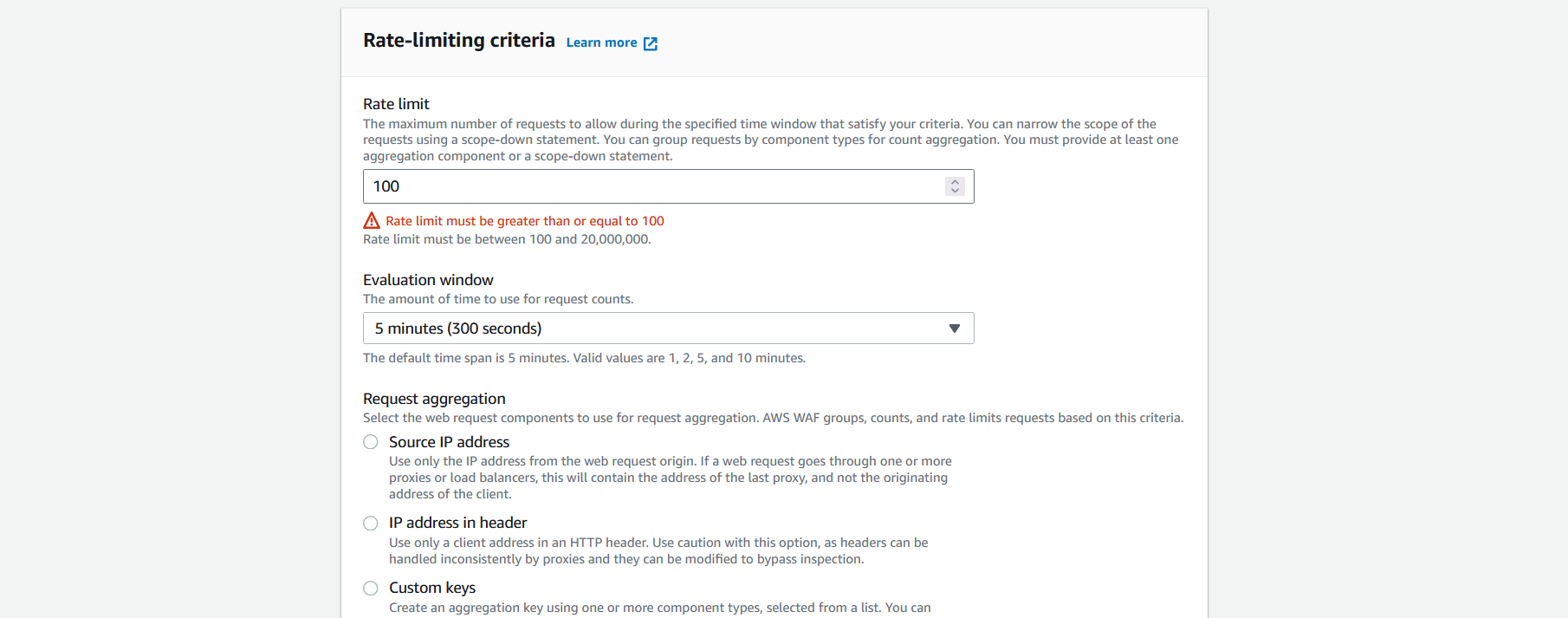
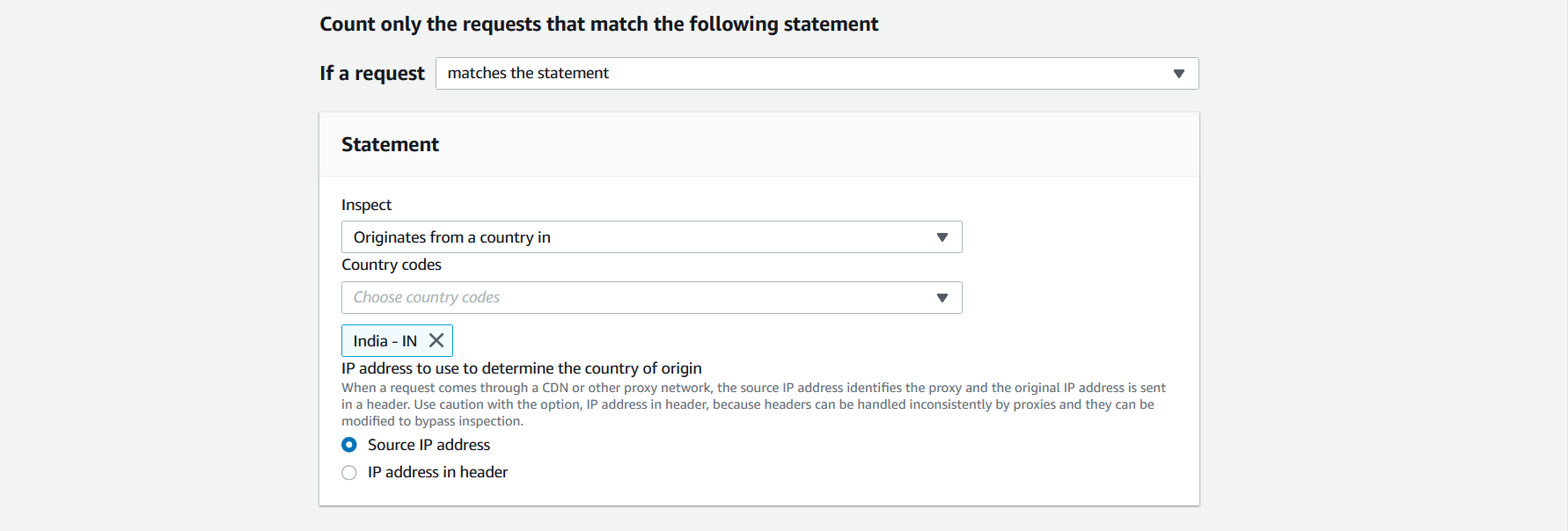
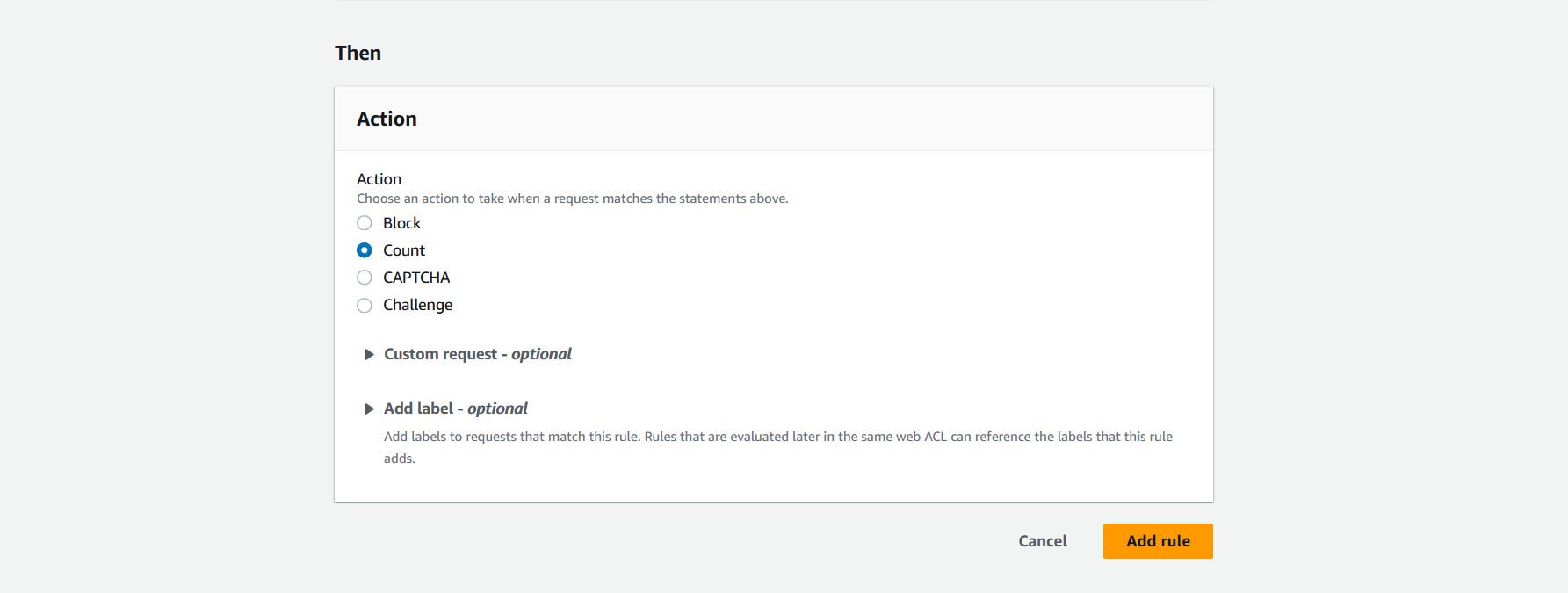
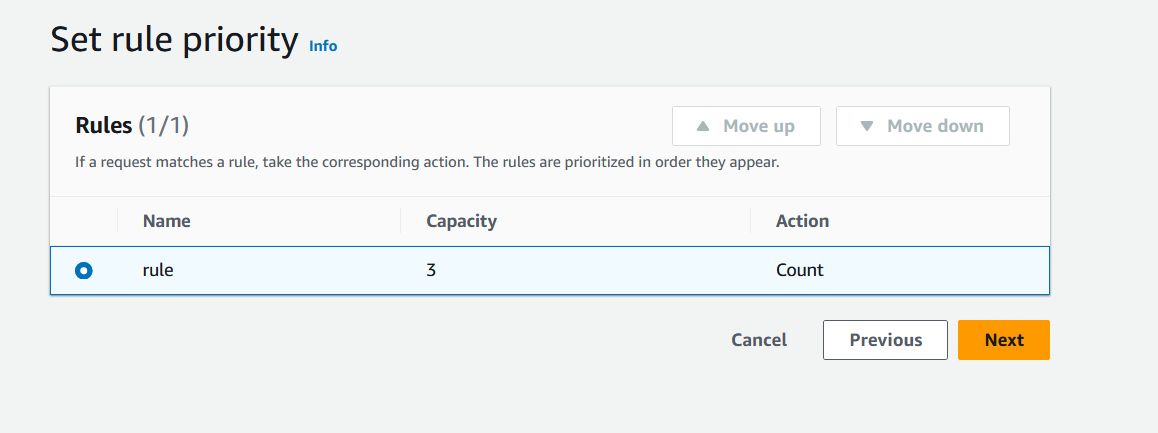
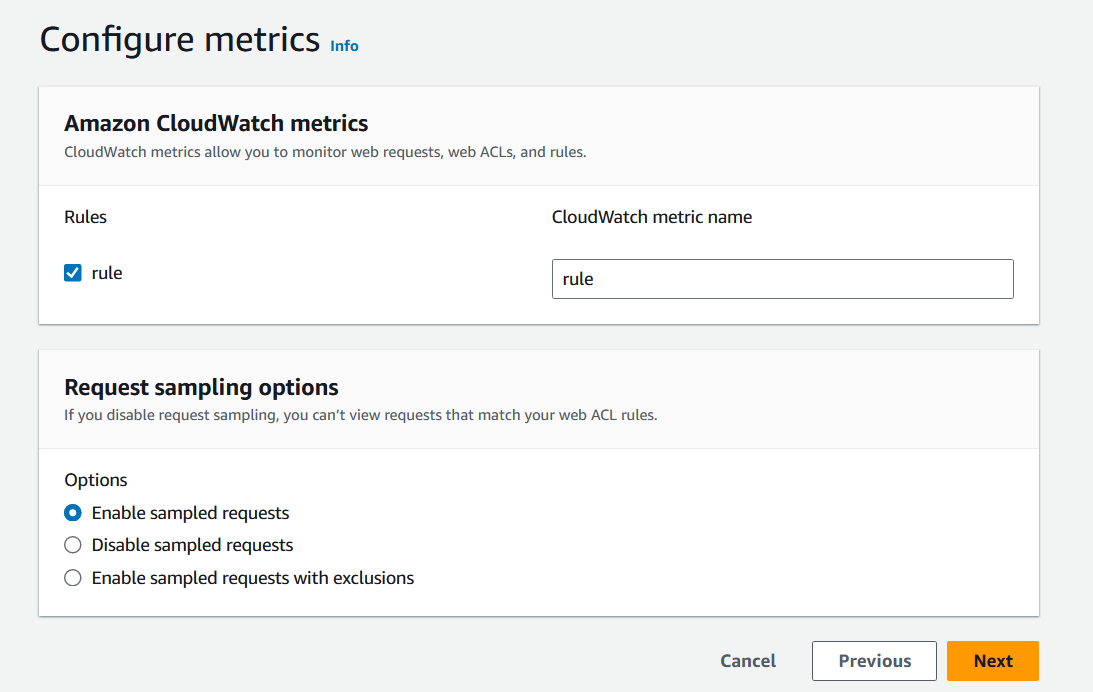
 

**Step 3: Implement AWS WAF**

1. Navigate to the AWS WAF & Shield dashboard.
2. Create a new Web ACL (Web Access Control List) and associate it with your CloudFront distribution or Application Load Balancer.

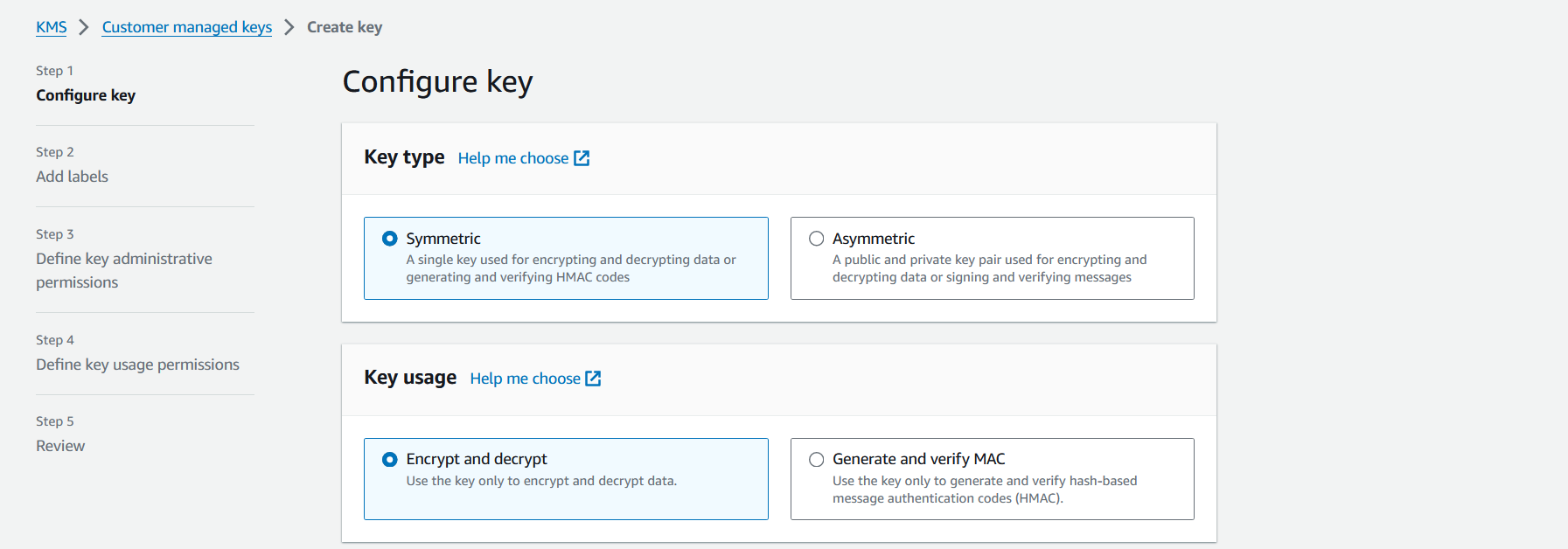
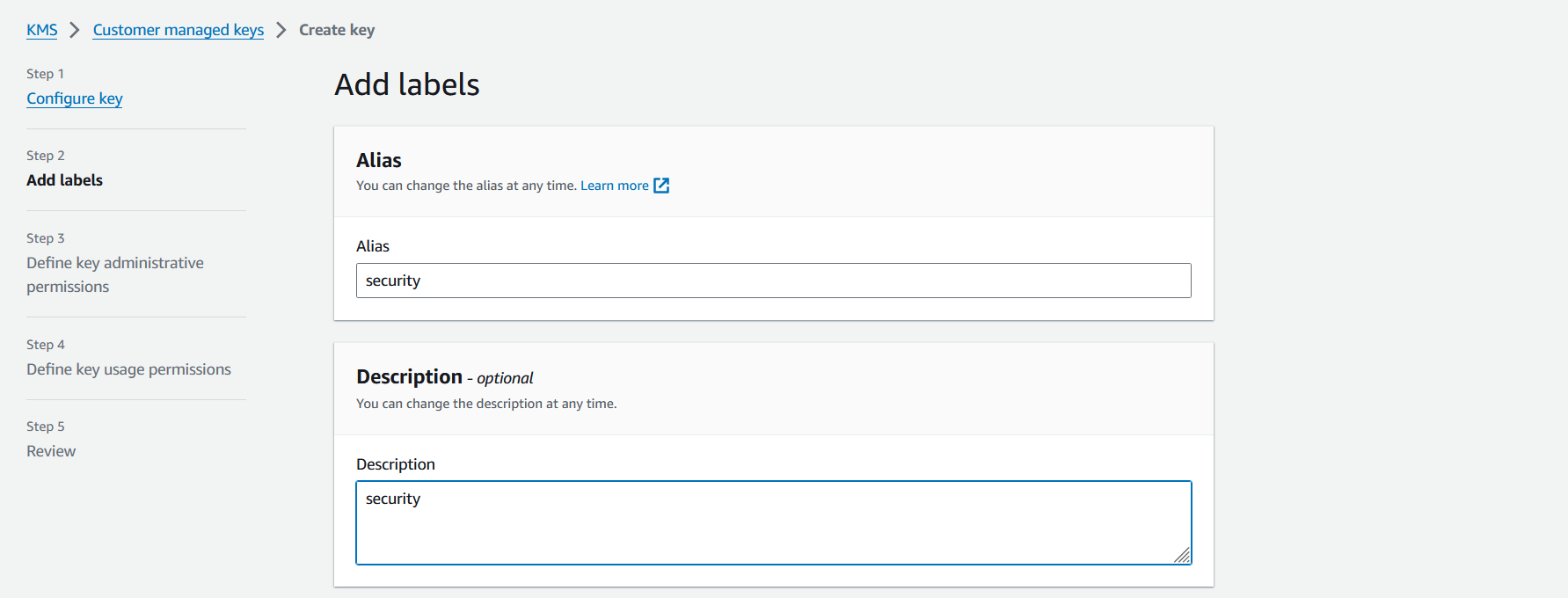
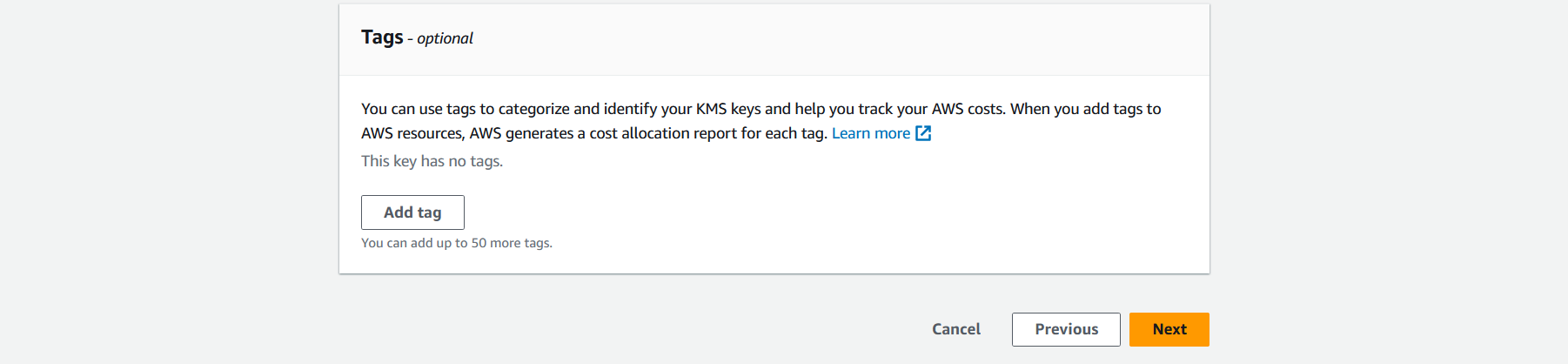
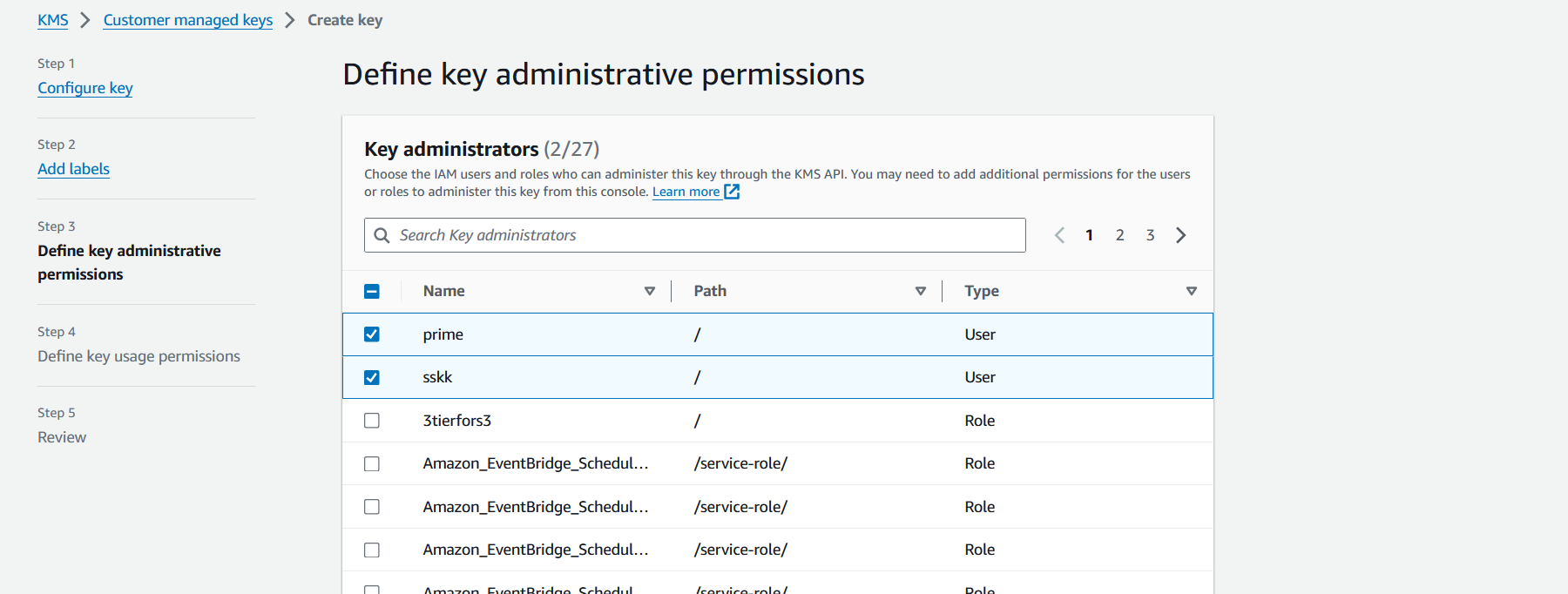
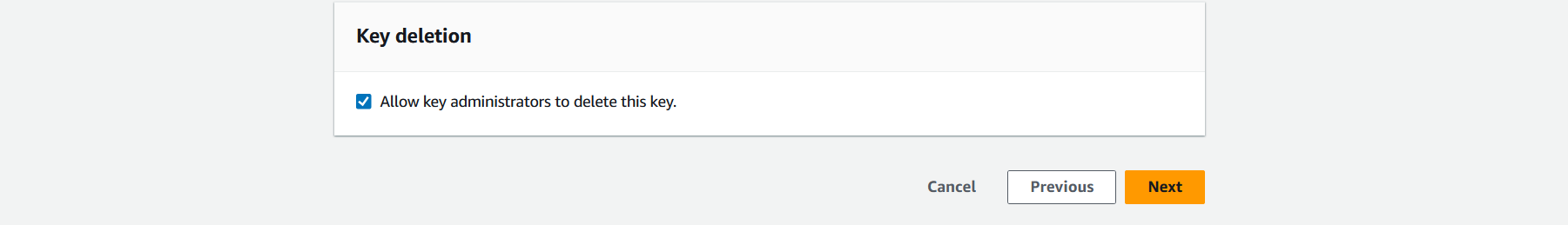
1. Configure rules to block or allow traffic based on conditions such as IP addresses, SQL injection, cross-site scripting (XSS), etc.

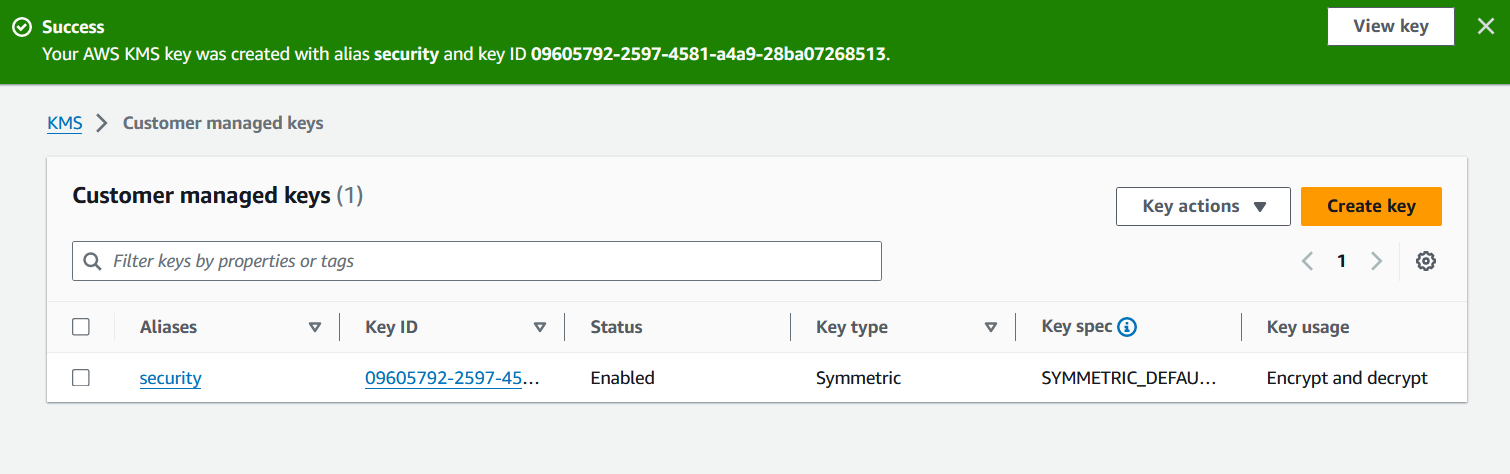
1. Regularly monitor and update your WAF rules to adapt to new threats.

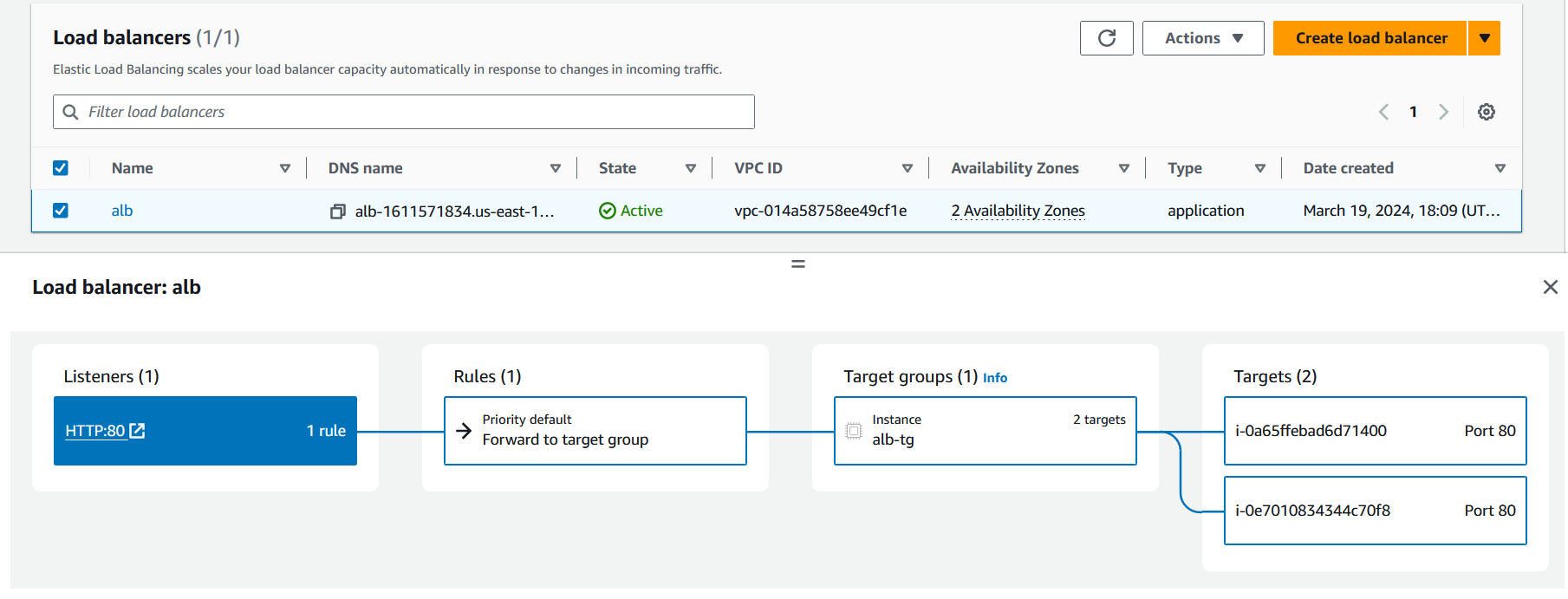
**Step 4: Set Up AWS KMS for Encryption**

1. Navigate to the AWS KMS dashboard.

1. Create a new Customer Master Key (CMK) or use the default key.
2. Define key policies to control access to the CMK.
3. Use the CMK to encrypt sensitive data stored in EBS volumes or other AWS services.





**Step 5: Configure AWS Security Hub for Monitoring**

1. Navigate to the AWS Security Hub dashboard.
2. Enable Security Hub for your AWS account region.
3. Integrate Security Hub with AWS Config, AWS CloudTrail, and Amazon GuardDuty for comprehensive security monitoring.
4. Set up automated security checks and custom actions based on findings.
5. Configure CloudWatch Events to trigger SNS (Simple Notification Service) topics for sending alerts.

**Step 6: Regular Maintenance and Monitoring**

1. Regularly review AWS Security Hub findings and take necessary actions to remediate security issues.
2. Monitor CloudWatch metrics for EC2 instances, ELB, and other AWS resources.
3. Set up CloudWatch Alarms to trigger notifications for CPU utilization, network traffic, etc.
4. Perform regular security audits, penetration testing, and vulnerability assessments.
5. Keep abreast of AWS security best practices and new features to continuously improve your infrastructure's security posture.

**Conclusion**

By following these steps, you can implement a highly secured infrastructure on AWS while configuring robust monitoring with alerts. Make sure to document each step thoroughly for reference and future maintenance purposes. Regularly review and update your security measures to stay ahead of emerging threats and ensure ongoing protection for your AWS environment.

We can consider few additional points to remember and best practices to consider when implementing a highly secured infrastructure on AWS:

**Points to Remember:**

1. **Least Privilege Principle**: Grant only the minimum level of access necessary for each user or service.
2. **Regular Backups**: Implement automated backups for critical data and systems.
3. **Audit Logging**: Enable detailed logging for all AWS services and regularly review logs for suspicious activities.
4. **Network Segmentation**: Use separate VPCs and subnets for different tiers of your application to limit the blast radius of potential breaches.
5. **Data Encryption**: Encrypt data at rest and in transit using AWS KMS and SSL/TLS encryption.
6. **Secure Key Management**: Follow best practices for managing encryption keys, including key rotation and restricted access.
7. **Patch Management**: Keep EC2 instances and other AWS services up to date with the latest security patches.
8. **Multi-Factor Authentication (MFA)**: Enable MFA for AWS IAM users to add an extra layer of security.
9. **Incident Response Plan**: Develop and regularly test an incident response plan to effectively handle security incidents.
10. **Compliance and Regulatory Requirements**: Ensure that your infrastructure complies with relevant industry standards and regulations.

**Best Practices:**

1. **Automate Security**: Use AWS services such as AWS Config, AWS Systems Manager, and AWS Security Hub to automate security tasks.
2. **Immutable Infrastructure**: Deploy infrastructure using templates (e.g., AWS CloudFormation) and immutable images to reduce configuration drift and enhance security.
3. **Continuous Monitoring**: Implement continuous monitoring and alerting using AWS CloudWatch, AWS CloudTrail, and AWS GuardDuty.
4. **Implement Security as Code**: Define security configurations as code using tools like AWS CloudFormation or AWS CDK to ensure consistency and repeatability.
5. **Regular Security Assessments**: Conduct regular security assessments, including vulnerability scanning and penetration testing, to identify and remediate potential weaknesses.
6. **Third-party Security Tools**: Consider using third-party security tools and services that integrate with AWS for additional protection and visibility.
7. **Documentation and Training**: Document security configurations, procedures, and incident response plans, and provide regular security training for personnel.
8. **Keep Abreast of AWS Security Updates**: Stay informed about AWS security best practices, new features, and updates to ensure your infrastructure remains secure against evolving threats.
9. **Secure Development Practices**: Implement secure coding practices and perform code reviews to prevent common security vulnerabilities in your applications.
10. **Regular Reviews and Updates**: Review and update security configurations, policies, and procedures regularly to adapt to changes in your environment and emerging threats.

By incorporating these points to remember and best practices into your AWS infrastructure, you can enhance its security posture and better protect your applications and data from potential threats and vulnerabilities.