**Objective:** Enhance EKS cluster security with AWS native tools.

**Tasks:**

1. Enable AWS GuardDuty for EKS.

2. Set up AWS WAF for EKS exposed services.

3. Monitor and respond to security threats.

**Before Perform this lab:**

**Require tools:**

1. Eksctl
2. Kubectl
3. Aws cli
4. Docker
5. Basic knowledge for Kubernetes

**Documentation:**

- EKS security best practices.

- Introduction to GuardDuty and WAF.

- Proactive threat detection and response.

Task

**Task 1: Enable AWS GuardDuty for EKS**

| # Enable GuardDuty. This creates a GuardDuty detector for the current region. aws guardduty create-detector --enable  # If you need to specify a particular region, use the --region flag # aws guardduty create-detector --enable --region us-east-1  # List all detectors (to get the DetectorId for the next steps) aws guardduty list-detectors  # Replace 'DetectorId' with the actual ID from the previous output # Enable Kubernetes protection aws guardduty update-detector --detector-id DetectorId --data-sources '{"Kubernetes": {"AuditLogs": {"Enable":true}}}' |
| --- |

**Task 2. Set up AWS WAF for EKS exposed services.**

| aws elbv2 create-load-balancer --name my-load-balancer --subnets subnet-1234567890abcdef0 subnet-0987654321abcdef0 --security-groups sg-0123456789abcdef0 |
| --- |

**Check Load Balancer Status:**

| aws elbv2 describe-load-balancers --load-balancer-arns arn:aws:elasticloadbalancing:us-east-1:587027614473:loadbalancer/app/my-load-balancer/1463bb0c091ad97a |
| --- |

**Check Web ACL Status:**

| aws wafv2 list-web-acls --scope REGIONAL --region us-east-1 |
| --- |

| aws wafv2 create-web-acl --name my-web-acl --scope REGIONAL --region us-east-1 --default-action Allow={} --visibility-config SampledRequestsEnabled=true,CloudWatchMetricsEnabled=true,MetricName=my-web-acl-metric  aws elbv2 describe-load-balancers  aws wafv2 associate-web-acl \ --web-acl-arn arn:aws:wafv2:us-east-1:587027614473:regional/webacl/my-web-acl/72863abc-4d78-49b8-a33c-e5a75aee316a \ --resource-arn arn:aws:elasticloadbalancing:us-east-1:587027614473:loadbalancer/app/my-load-balancer/1463bb0c091ad97a |
| --- |

**Task 3. Monitor and respond to security threats.**

| ##Check if the Log Group Exists: aws logs describe-log-groups  aws guardduty list-detectors ##Create the Log Group: aws logs create-log-group --log-group-name /aws/guardduty/DetectorId  # Create a metric filter that matches GuardDuty findings aws logs put-metric-filter \ --log-group-name /aws/guardduty/DetectorId \ --filter-name GuardDutyFilter \ --filter-pattern '{ $.detail.type = "Recon:EC2/PortProbeUnprotectedPort" }' \ --metric-transformations metricName=PortProbeMetric,metricNamespace='GuardDuty',metricValue=1  # Create an SNS topic to receive notifications aws sns create-topic --name guardduty-alerts  # Subscribe to the SNS topic **(replace EMAIL with your email)** aws sns list-topics  aws sns subscribe --topic-arn SnsTopicArn --protocol email --notification-endpoint EMAIL@example.com  aws sns list-topics  # Create a CloudWatch alarm based on the metric filter aws cloudwatch put-metric-alarm \  --alarm-name GuardDutyAlarm \  --metric-name PortProbeMetric \  --namespace 'GuardDuty' \  --statistic Sum \  --period 300 \  --evaluation-periods 1 \  --threshold 1 \  --comparison-operator GreaterThanOrEqualToThreshold \  --alarm-actions arn:aws:sns:us-east-1:587027614473:guardduty-alerts |
| --- |

Be sure to replace **SnsTopicArn** with the actual ARN of your SNS topic where you want to send the notifications.

Using the AWS Management Console

Sign in to the AWS Management Console and open the Amazon SNS console at <https://console.aws.amazon.com/sns/v2/home>.

In the navigation pane, choose Topics.

In the list of topics, you can see the ARNs listed under the ARN column next to the corresponding topic name.

**Documentation**

**EKS Security Best Practices**

* Ensure all Kubernetes workloads are running with the least necessary privileges.
* Regularly update your EKS clusters and worker nodes to the latest stable versions.
* Implement network policies to control the communication between pods.

**Introduction to GuardDuty and WAF**

**GuardDuty:** A threat detection service that provides detailed security findings.

**WAF:** A web application firewall that helps protect your web applications or APIs.

**Proactive Threat Detection and Response**

* Implementing GuardDuty and WAF helps in proactive threat detection.
* Ensure that you have response mechanisms in place like AWS Lambda functions to automatically remediate specific security findings.