Project 3: Developing an Automated Security Monitoring and Response System

Detailed Report

Project Objective: Build an automated system for monitoring and responding to security incidents in an AWS environment.

Guide

1. **Set up CloudWatch Monitoring:**
    - Configure CloudWatch metrics, events, and alarms for security monitoring.
2. **Automate Responses with Lambda:**
    - Create Lambda functions to automate responses to specific security incidents.
3. **Integrate Notifications with SNS:**
    - Use SNS to send notifications to the security team for prompt action.
4. **Centralize Security Management:**
    - Enable AWS Security Hub for a centralized view of security data.
5. **Manage Patches and Configurations:**
    - Utilize AWS Systems Manager for patch management and configuration changes.

Key Activities and Implementation:

1. **Monitoring with AWS CloudWatch:**
    - Utilized AWS CloudWatch to monitor metrics and events.
    - Configured alarms for security-related events.

```
bash
aws cloudwatch put-metric-alarm --alarm-name CPUAlarm --metric-name
CPUUtilization --namespace AWS/EC2 --statistic Average --period 300 -
-threshold 70 --comparison-operator GreaterThanOrEqualToThreshold --
dimensions Name=InstanceId,Value=i-1234567890abcdef0 --evaluation-
periods 2 --alarm-actions arn:aws:sns:eu-central-1a
:123456789012:MyTopic
```

2. **Automated Response with AWS Lambda:**
    - Created AWS Lambda functions for automatic response to security incidents.

```
python
import json
import boto3

def lambda_handler(event, context):
    ec2 = boto3.client('ec2')
    instance_id = event['detail']['instance-id']
    ec2.stop_instances(InstanceIds=[instance_id])
    return {
        'statusCode': 200,
        'body': json.dumps('Instance stopped')
    }
```

3. **Notification with AWS SNS:**
    - Integrated AWS SNS to notify the security team of security-related events.

```
bash
aws sns create-topic --name security-alerts
aws sns subscribe --topic-arn arn:aws:sns: eu-central-
1a:123456789012:security-alerts --protocol email --notification-
endpoint sascha.meyer,it@gmail.com
```

4. **Centralized Management with AWS Security Hub:**
    - Utilized AWS Security Hub for centralized management and aggregation of security data.

```
bash
aws securityhub enable-security-hub
```

5. **Patching and Configuration Changes with AWS Systems Manager:**
    - Employed AWS Systems Manager for performing patches and configuration changes.

```
bash
aws ssm create-patch-baseline --name MyPatchBaseline --operating-
system WINDOWS --approved-patches ComplianceLevel=CRITICAL --
approval-rules
PatchRules=[PatchFilterGroup={PatchFilters=[{Key=PRODUCT,Values=[Wind
owsServer2016]}]}]
```

Outcome: An automated and efficient system for monitoring and responding to security incidents, reducing response times and enhancing the overall security posture.