Capstone Summary

Project Title: Cloud-Based Serverless Security Orchestration, Automation, and Response (SOAR) System on AWS

1. Introduction

This capstone project focused on designing and deploying a cloud-native, serverless SOAR system on AWS. The initiative was driven by the need to provide small and medium-sized enterprises (SMEs) with an affordable yet effective solution for incident detection and response. Many SMEs either lack the capital to invest in complex automated SOAR/SIEM platforms or cannot dedicate resources for continuous monitoring and analysis. This project demonstrates that AWS-native services can be orchestrated into a modular, auditable, and cost-efficient SOAR framework accessible to organizations with limited budgets.

2. Objectives

The objectives of the project were to:

1. Implement a serverless SOAR architecture using AWS-native services.
2. Automate incident response playbooks for common cloud threats.
3. Ensure auditable logs of detection and remediation in DynamoDB and S3.
4. Deliver real-time alerts via SNS.
5. Provide an interactive dashboard for visualizing detections and responses.

3. System Design and Architecture

The system leverages Infrastructure as Code (CloudFormation) for modular deployment:

- Detection: GuardDuty continuously analyzes AWS resources for malicious activity.
- Orchestration: EventBridge routes findings to the relevant Lambda playbook.
- Response: Lambda applies containment, such as IP blocking, IAM key revocation, or S3 policy removal.
- Logging: DynamoDB stores detailed remediation metadata; S3 archives logs and outputs.
- Alerting: SNS sends real-time notifications to security personnel.

- Visibility: A GitHub-hosted dashboard integrates via API Gateway to present metrics and remediation outcomes.

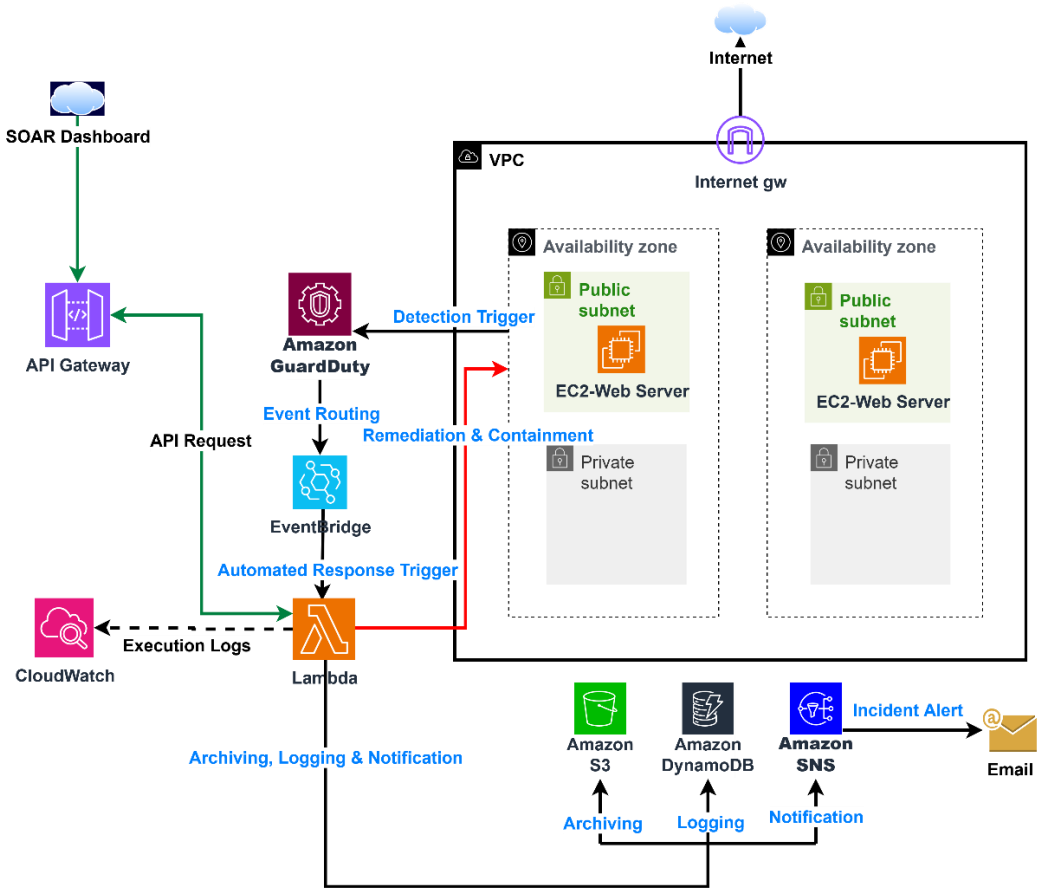

Fig 1: SOAR Architecture Diagram

AWS Service	Role in SOAR Platform
Amazon VPC	Provides the secure network foundation with subnets, security groups, and NACLs; supports isolation and dynamic IP blocking.
Amazon EC2	Acts as the attack surface for testing; intentionally exposes SSH and HTTP services within a public subnet.
Amazon GuardDuty	Core detection engine; analyzes logs and network activity to detect brute force, port scans, credential misuse, and malicious IPs.
Amazon EventBridge	Orchestration layer; routes GuardDuty findings and custom detections to the appropriate Lambda playbooks.
AWS Lambda	Executes automated responses such as IP blocking, IAM key deactivation, and S3 policy enforcement following NIST guidelines.
Amazon SNS	Sends real-time alerts to stakeholders on remediation events or manual review triggers.
Amazon DynamoDB	Stores structured logs (RemediationLog, ThreatMetadata, NACLBlockList) for auditing, reporting, and visualization.
Amazon S3	Archives forensic data, logs, and GuardDuty findings for traceability and compliance.
Amazon API Gateway	Powers the SOAR dashboard by securely exposing endpoints to retrieve and display real-time data from DynamoDB and S3.

Table 1: Components of AWS Cloud-Based SOAR

4. Playbooks Implemented

Eight automated responders were implemented:

1. SSH Brute Force → Block source IP and tag affected EC2.
2. Port Scanning → Apply NACL deny rules.
3. IAM Anomalous Behavior → Restrict console access.
4. IAM Exfiltration → Revoke IAM access keys.
5. Web Login Abuse (custom) → Block IP after repeated failed login attempts.
6. Tor Access → Block Tor exit node IPs.
7. GeoIP Threats → Block regions flagged as high risk.
8. S3 Unauthorized Access → Remove public bucket policies and enforce blocks.

The framework was designed for flexibility. While eight playbooks were implemented, the system can easily be expanded to include additional playbooks tailored to an organization's unique requirements.

Remediation Playbook Name	GuardDuty Finding Type	Containment Action	Review Classification	Severity
SSHBruteForce	UnauthorizedAccess:EC2/SSHBruteForce	Close port 22, tag EC2 for quarantine.	Automated	Low
Port Scanning	Recon:EC2/PortProbeUnprotectedPort Recon:EC2/Portscan Impact:EC2/PortSweep	Remove access to all ports except 80/443 (web-only)	Manual Review Required	Low
IAM User Anomaly Behavior	CredentialAccess:IAMUser/AnomalousBehavior Persistence:IAMUser/AnomalousBehavior	Disable IAM user's console login if anomaly behavior is detected. Demo Users created for this.	Manual Review Required	Medium
IAM User Credential Exfiltration	Exfiltration:IAMUser/AnomalousBehavior	Automatically deactivate the affected IAM user's access key. Demo Users created for this.	Manual Review Required	High
Web Login Abuse	Custom.web.logs UnauthorizedAccess:IAMUser/ConsoleLoginSuccess.B	Block offender's IP after 5 consecutive failed login attempts using custom logs, and update the NACL table	Automated	High
Tor Access	UnauthorizedAccess:EC2/TorClient UnauthorizedAccess:Runtime/TorClient	Block Tor client IP and update the NACL table	Automated	Medium
Geo IP Threat	UnauthorizedAccess:EC2/MaliciousIPCaller.Custom	Block IPs found in threat IP list or Regions highlighted as high-risk	Automated	Critical
S3 Anonymous Unauthorized Access	Policy:S3/BucketAnonymousAccessGranted	Automatically remove public access and apply strict bucket policies.	Automated	Critical

Table 2: SOAR Playbook & Containment Actions

Playbook	AWS Resource	Pre-test setup	How to simulate attack
SSH Brute Force	EC2	Make SSH open to all IP	Hydra brute force from Kali/external
Port Scanning	EC2	Enable multiple open ports	Nmap scan from outside AWS
IAM Anomaly	IAM User	Create demo user with console login	Impossible travel/mass API calls/VPN
IAM Key Exfiltration	IAM User	Create Demo user with access keys	Key use from suspicious region/mass S3 download
Web Login Abuse	EC2/Web App/IAM	Web login page & monitor	Attempt 5 consequetive login fails on web app
Tor Access	EC2	Open port, attacker with Tor	Proxychains or Tor browser scan
GeoIP Threat	EC2	Open port, add threat ip to GuardDuty threat list	Access/scan from Threap IP using VPN
S3 Unauthorized	S3	Attach public policy, and set block public off	Attach public policy, then access file from Internet

Table 3: SOAR Playbook Simulation Summary

5. Testing and Validation

The framework was validated using manual EventBridge test events and controlled attack simulations from Kali Linux. Results confirmed that:

- GuardDuty findings invoked the correct Lambda responders.
- Containment actions were successfully applied.
- Logs were consistently recorded in DynamoDB and archived in S3.
- The dashboard accurately visualized MTTR, remediation rates, and automated vs manual classifications.

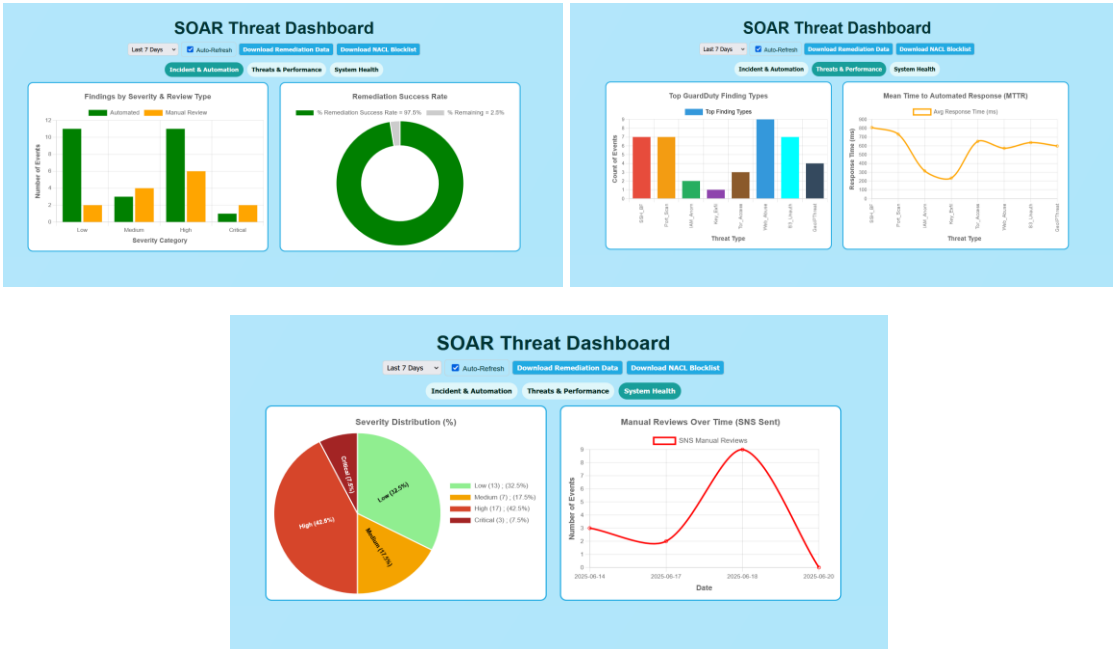

Fig 2: SOAR Dashboard Overview

6. Results

- Achieved real-time containment across diverse threat scenarios.
- Delivered a 40% reduction in incident handling time compared to manual workflows.
- Validated that the system can run cost-effectively within AWS Free Tier constraints, making it feasible for SMEs.
- Proved that serverless automation can bridge the resource gap faced by smaller firms without a dedicated SOC team.

7. Conclusion

The project successfully demonstrated a scalable and cost-effective SOAR system for organizations that cannot afford commercial-grade platforms or maintain dedicated monitoring staff. By combining AWS-native services, the system achieves continuous detection, automated remediation, structured logging, and dashboard visualization in a unified framework.

The solution is especially suited for small and medium enterprises, enabling them to benefit from modern security automation without significant financial investment. Furthermore, the playbook library is extensible, allowing firms to add new detection and response logic based on evolving security needs.

Future work could incorporate threat intelligence feeds, machine learning-based anomaly detection, and multi-cloud support to further enhance adaptability and coverage.