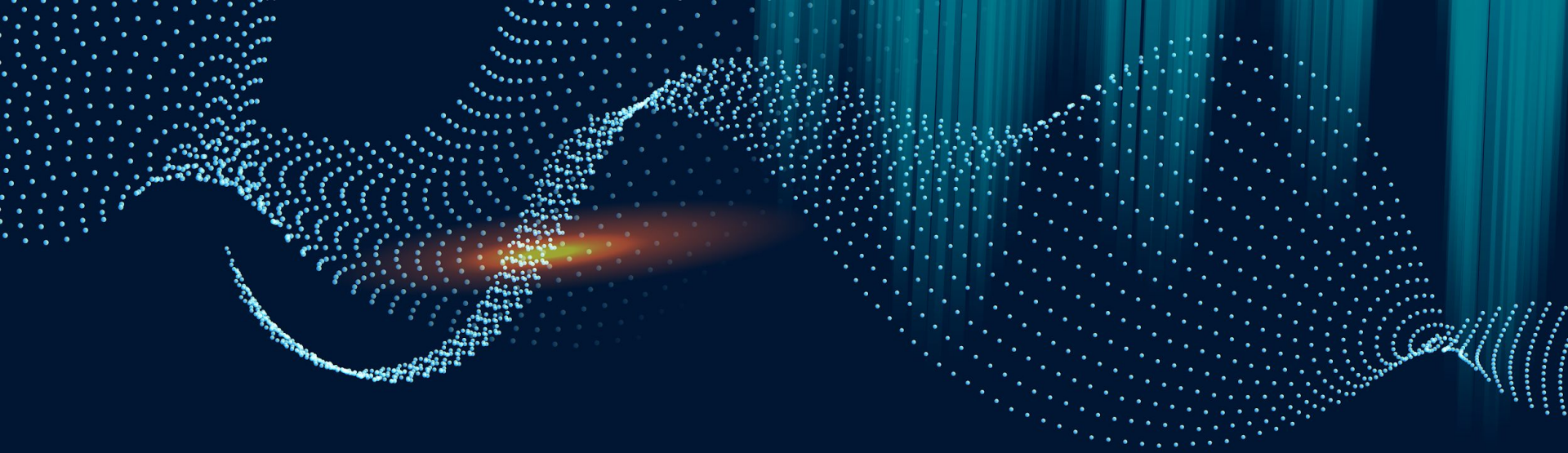
SOC DOCS

Cloud Based Security Operations Center

Team 2: Giovanni Garcia, Mario Register, Pedro Gomez, Matthew Escalera



Introduction

The Who, What, Where, and Why

Introduction

What is a Security Operations Center (SOC)?

A Security Operations Center (SOC) is essentially a dedicated team and a facility responsible for monitoring, detecting, and responding to cybersecurity incidents around the clock. Think of it as the digital equivalent of a security control room that you might find in a large building or campus.

Here are the key points to understand about a SOC:

Constant Vigilance: The SOC operates 24/7, continuously watching over the organization's digital infrastructure, including networks, servers, computers, and data.

Threat Detection: It uses specialized tools and technologies to identify potential security threats, such as hacking attempts, malware, or unauthorized access.

Incident Response: When a threat is detected, the SOC team acts quickly to mitigate the impact. This might involve shutting down parts of the network, removing malicious software, or taking other actions to protect sensitive information.

Proactive Measures: Beyond reacting to immediate threats, the SOC also works on preventing future incidents by identifying vulnerabilities and implementing security improvements.

What is Cloud Computing?

Cloud computing allows us to store and access data and applications over the internet instead of on local servers or computers.

Benefits:

- Scalability: Easily adjust resources based on our needs.
- Cost-Effective: Pay only for what we use, reducing overhead costs.
- Accessibility: Access data and services from anywhere, anytime.

AWS (Amazon Web Services): A leading cloud platform providing reliable and secure infrastructure.

Key Advantages:

- 24/7 Monitoring: Continuous surveillance of our digital environment.
- Advanced Security Tools: Utilizes AWS's cutting-edge security technologies.
- Rapid Response: Quickly detect and respond to threats with AWS's robust infrastructure.
- Scalable Resources: Easily expand our SOC capabilities as our needs grow.
- Cost Efficiency: Optimize costs by using AWS's pay-as-you-go model.

Sprint Schedule

Sprint 1	Sprint 2	Sprint 3	Sprint 4	Sprint 5
Project Planning & Requirements Gathering	Secure Cloud Infrastructure & Access Control	Security Data Collection & Aggregation	Security Automation & Alert Correlation	User Interface & Security Monitoring
April 15 - April 29	April 29 - May 13	May 13 - May 27	May 27 - June 10	June 10 - June 24

Project Goal

The primary goal of the project is to establish a scalable and automated Security Operations Center (SOC) on AWS to centralize security monitoring, incident detection, and response capabilities. Secondary goals include improving security posture and reducing response times to security incidents

Scope

The project will focus on designing, deploying, and configuring the SOC infrastructure on AWS, including the selection and integration of appropriate AWS services and security tools. It will not include extensive customization or integration with on-premises systems.

Target Audience

The SOC platform will primarily serve security operations professionals, IT security teams, and security analysts within an organization.

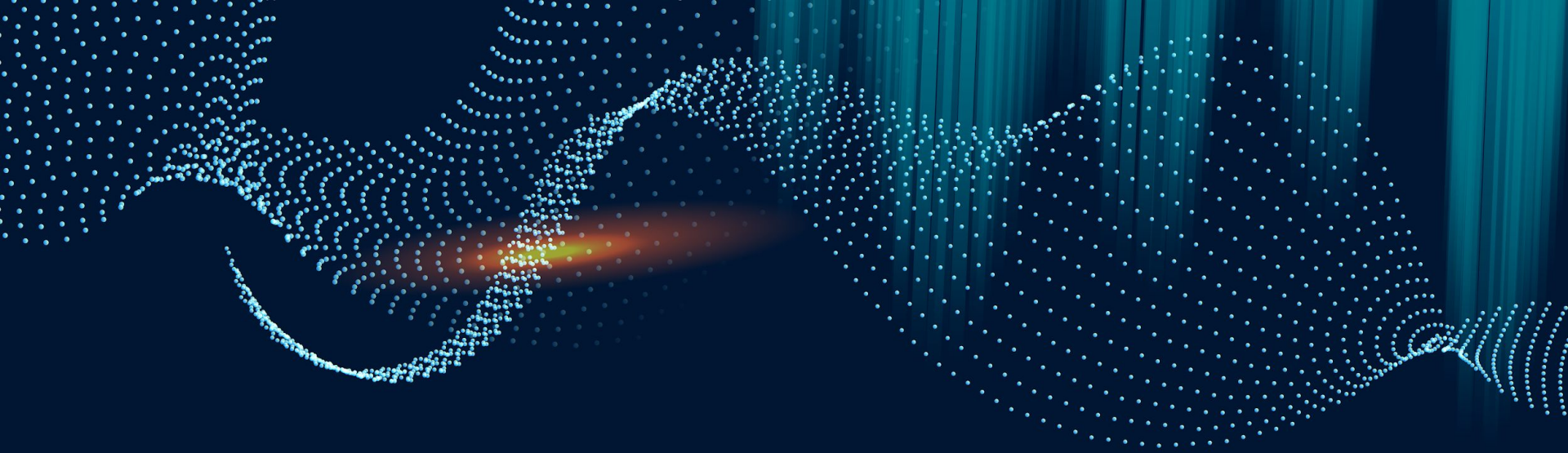
Success Metrics

Reduction in alert fatigue by at least 30% within six months of SOC deployment. Increase incident response times by 20%

Achievement of an average incident response time of less than 15 minutes for critical security events.

Improvement in security visibility with centralized monitoring and reporting capabilities.

Streamline incident response workflows through automation playbooks and improve overall response efficiency

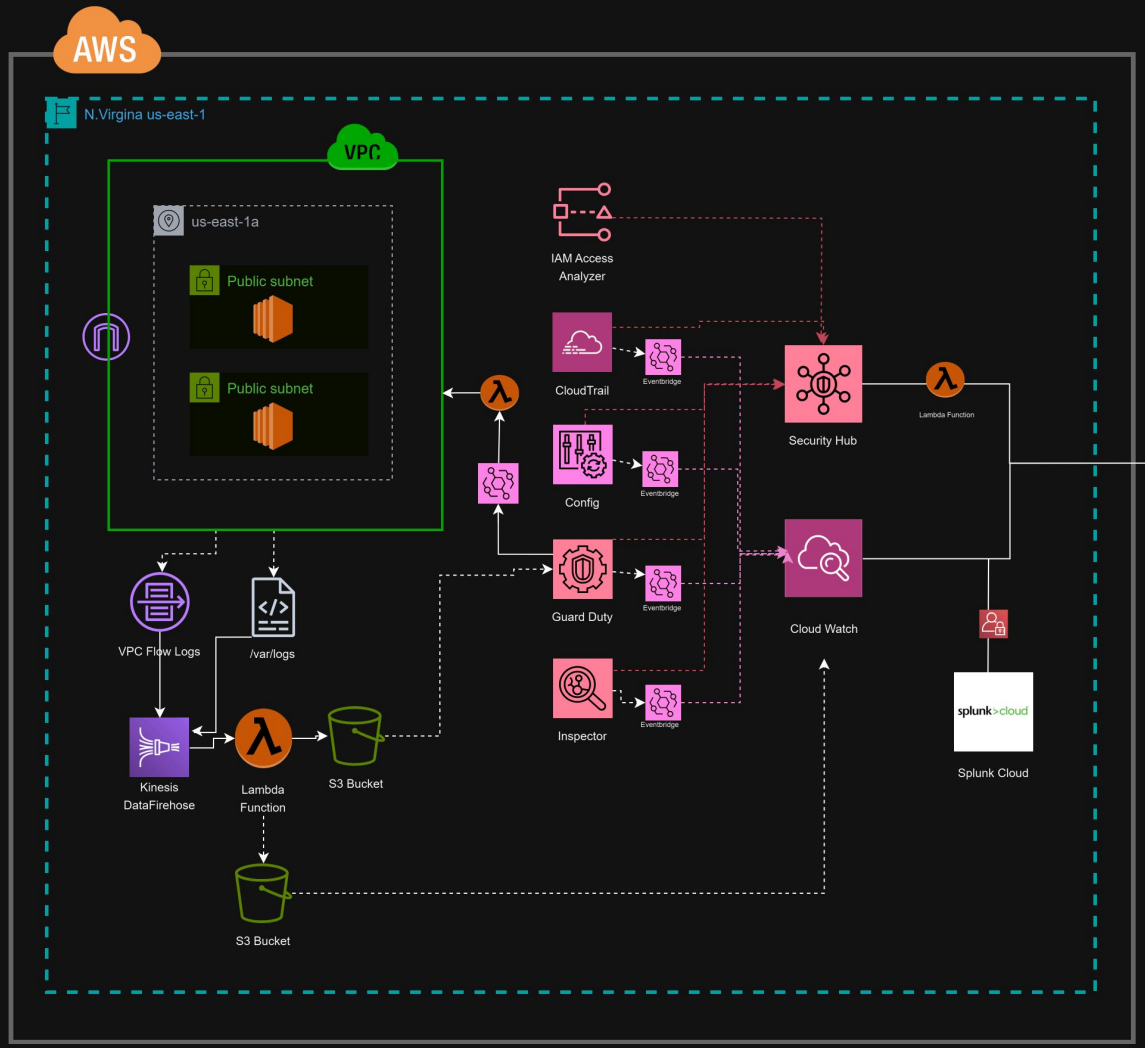


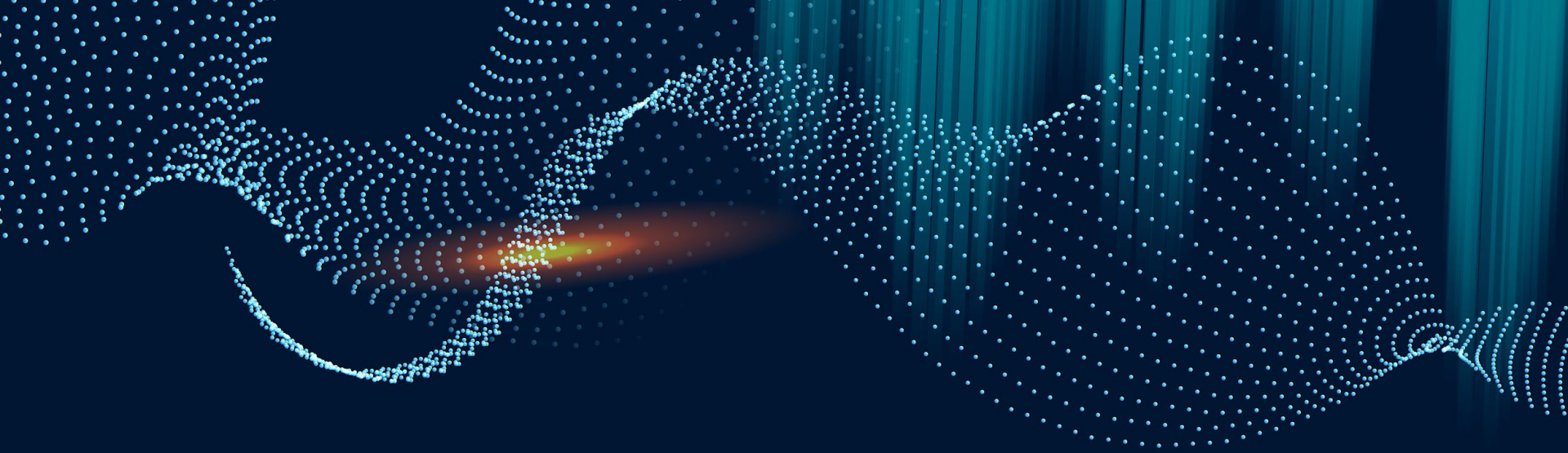
Solution Overview

Cloud-Based Security Operations Center (SOC) on AWS

The Challenge

- Rapidly evolving threat landscape
- Increasing complexity of cloud infrastructures
- Need for real-time threat detection and response
- Managing vast amounts of security data
- Shortage of skilled cybersecurity professionals
- Maintaining compliance across multiple cloud services
- Balancing security with operational efficiency
- Addressing the unique risks of cloud environments
- Coordinating security across diverse AWS services
- Automating security processes to reduce human error


AWS
N.Virgina us-east-1
VPC
us-east-1a
Public subnet
Public subnet
IAM Access Analyzer
CloudTrail
Eventbridge
Security Hub
Lambda Function
Config
Eventbridge
SNS
Guard Duty
Eventbridge
Cloud Watch
Inspector
Eventbridge
splunk>cloud
Splunk Cloud
VPC Flow Logs
/var/logs
Kinesis DataFirehose
Lambda Function
S3 Bucket
S3 Bucket



Technologies

AWS Services and Tools Powering Our SOC

CloudWatch: Monitoring and observability service for AWS resources and applications.

Security Hub: Central hub for viewing and managing security alerts across AWS accounts.

GuardDuty: Intelligent threat detection service that continuously monitors for malicious activity.

IAM (Identity and Access Management): Manages access to AWS services and resources securely.

Config: Assesses, audits, and evaluates the configurations of your AWS resources.

Amazon Inspector

Inspector: Automated security assessment service to help improve the security and compliance of applications.

CloudTrail:Tracks user activity and API usage across your AWS infrastructure.

EventBridge: Serverless event bus that connects application data from your own apps, SaaS, and AWS services.

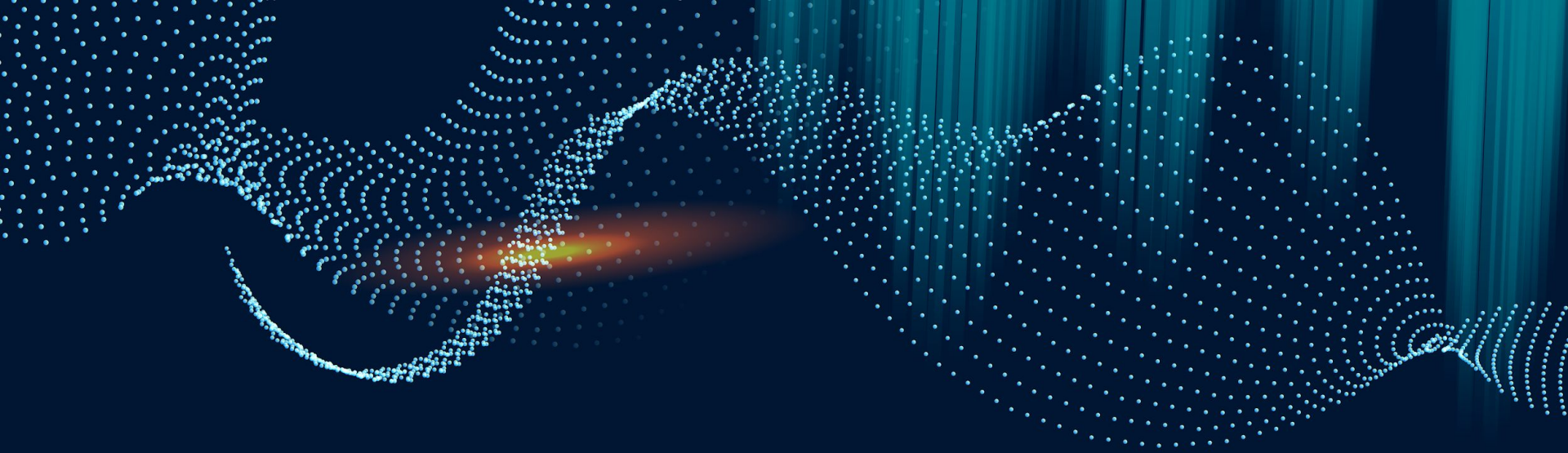
S3 (Simple Storage Service): Object storage service offering industry-leading scalability, data availability, and performance.

Lambda: Serverless compute service that runs code in response to events without provisioning servers.

EC2 (Elastic Compute Cloud): Scalable virtual servers for running applications in the cloud.

VPC Flow Logs: Captures information about IP traffic going to and from network interfaces in your VPC.

Splunk Cloud: Cloud-based service for real-time operational intelligence, log management, and advanced analytics. A Security information and event management tool for our lab.



Implementation

Key Components of Our AWS SOC Solution

Secure Infrastructure

Centralized Log Management

CloudWatch

Favorites and recents

Dashboards

Alarms 0 0 0

Logs

Log groups

Log Anomalies

Live Tail

Logs Insights

Contributor Insights

Metrics

All metrics

Explorer

Streams

X-Ray traces

Events

Rules

Log groups (10)

Actions | View in Logs Insights | Start tailing | Create log group

By default, we only load up to 10000 log groups.

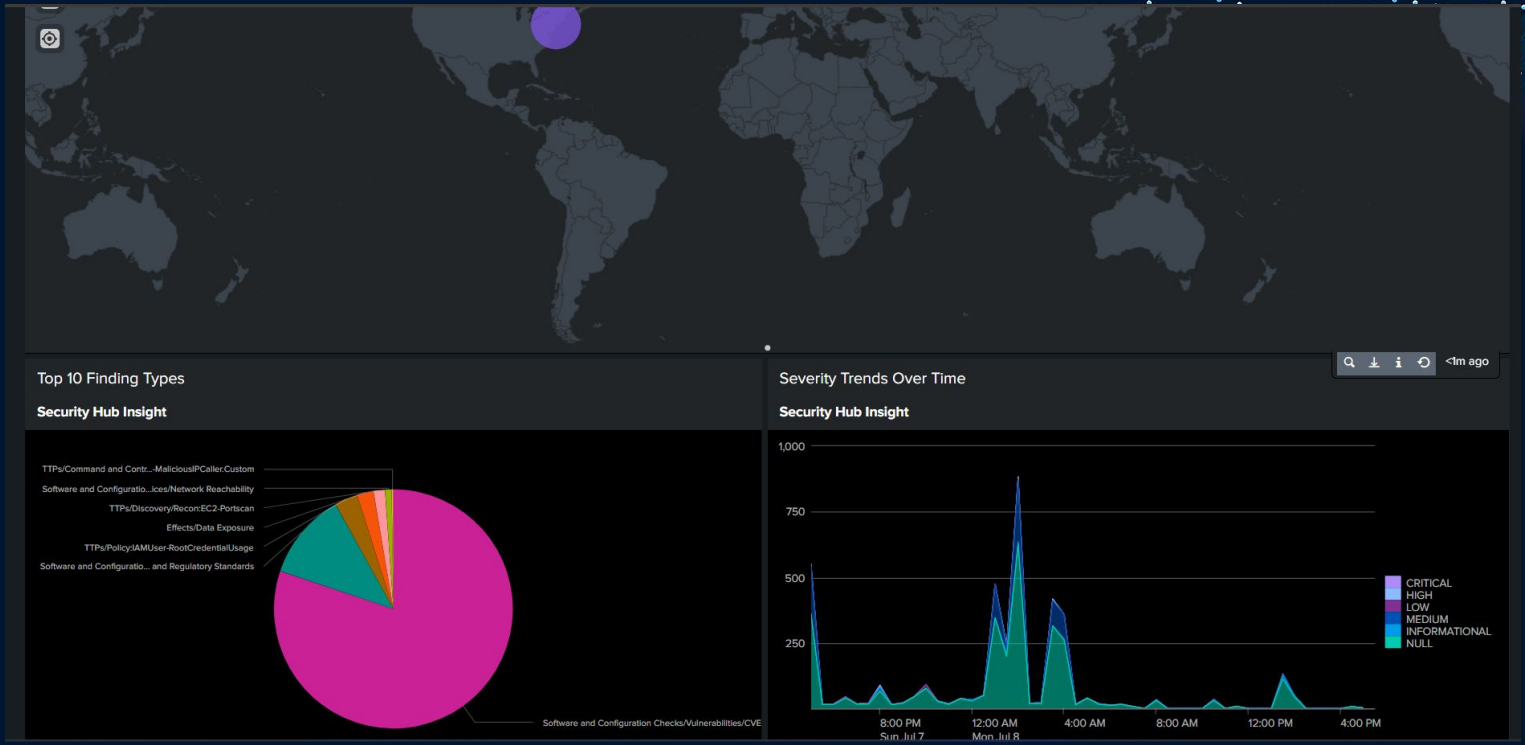
Filter log groups or try prefix search | Exact match | 1

Log group	Log class	Anomaly d...	Data p...	Sensit...	Retenti...	Metr
/aws/events/Config	Standard	Configure	-	-	Never expire	-
/aws/events/guardduty	Standard	Configure	-	-	Never expire	-
/aws/events/inspector	Standard	Configure	-	-	Never expire	-
/aws/events/securityhub	Standard	Configure	-	-	Never expire	-
/aws/guardduty/malware-scan-events	Standard	Configure	-	-	3 months	-
/aws/lambda/GuardDutyResponseFunction	Standard	Configure	-	-	Never expire	-
/var/log/auth.log	Standard	Configure	-	-	Never expire	-
/var/log/syslog	Standard	Configure	-	-	Never expire	-
aws-cloudtrail-logs-	Standard	Configure	-	-	Never expire	-
vpc-flowlogs	Standard	Configure	-	-	Never expire	-

CloudShell Feedback

Privacy Terms Cookie preferences

Advanced Analysis



Advanced Analysis

splunk>cloud | Apps | Messages | Settings | Activity | Find | Splunk Cloud Admin | Support & Services

Inputs | Configuration | Search | Health Check

New Search

Save As | Create Table View | Close

```
index="svc-sechub"
| eval
    title=coalesce('detail.findings{}.Title', "N/A"),
    severity=coalesce('detail.findings{}.Severity.Label', "N/A"),
    account='detail.findings{}.AwsAccountId',
    type=mvjoin('detail.findings{}.Types{}', ", "),
    first_observed='detail.findings{}.FirstObservedAt',
    last_observed='detail.findings{}.LastObservedAt'
| where title != "N/A" OR severity != "N/A"
| sort - _time
| table _time, title, severity, account, type, first_observed, last_observed
```

Last 24 hours

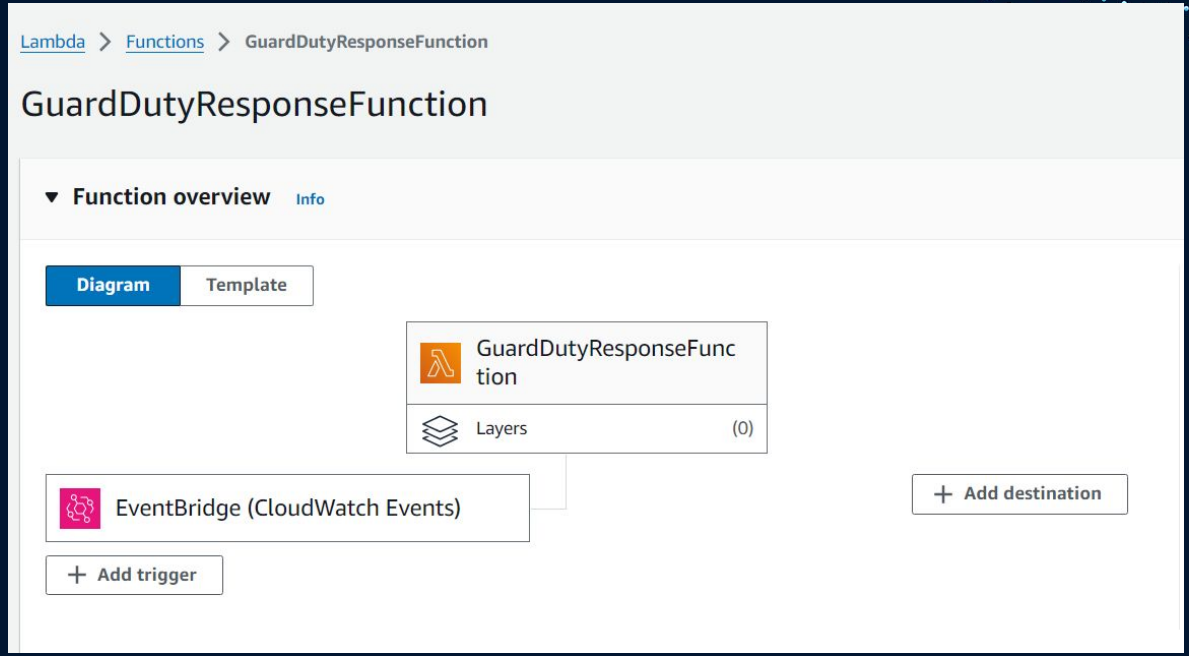
✓ **448 events** (7/7/24 5:00:00.000 PM to 7/8/24 5:14:29.000 PM) | No Event Sampling | Job | Policy-Based Pool | Fast Mode

Events | Patterns | Statistics (448) | Visualization

20 Per Page | Format | Preview | Prev | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | ... | Next

_time	title	severity	account	type	first_observed	last_observed
2024-07-08 14:18:58	S3 general purpose buckets should require requests to use SSL	MEDIUM	637423609702	Software and Configuration Checks/Industry and Regulatory Standards	2024-07-05T21:14:08.679Z	2024-07-08T14:18:52.902Z
2024-07-08 14:18:48	S3 general purpose buckets should block public access	INFORMATIONAL	637423609702	Software and Configuration Checks/Industry and Regulatory Standards	2024-07-05T21:14:08.554Z	2024-07-08T14:18:39.505Z
2024-07-08 14:18:41	S3 general purpose buckets should have server access logging enabled	MEDIUM	637423609702	Software and Configuration Checks/Industry and Regulatory Standards	2024-07-05T21:13:16.786Z	2024-07-08T14:18:36.157Z
2024-07-08 13:22:08	S3 general purpose buckets should block public write	INFORMATIONAL	637423609702	Effects/Data Exposure	2024-07-	2024-07-

SOAR Automated Response



SOAR Automated Response

aws | Services | Search [Alt+S] | N. Virginia | GG1211

EC2 | VPC | Lambda | IAM | Security Hub | CloudWatch | AWS Config | GuardDuty | CloudTrail | Amazon Data Firehose | S3 | Simple Notification Service | Amazon Inspector | Amazi

Code source Info

Upload from

File | Edit | Find | View | Go | Tools | Window | Test | Deploy

Go to Anything (Ctrl-P)

Environment

GuardDutyResponse

lambda_function.py

lambda_function × | Environment Var ×

```python
import json
import boto3

def lambda_handler(event, context):
    print(f"Received event: {json.dumps(event)}")

    # Initialize AWS clients
    ec2 = boto3.client('ec2')
    sns = boto3.client('sns')

    try:
        # Extract details from the event
        detail = event['detail']
        finding_type = detail.get('type', '')
        severity = detail.get('severity', 0)

        # Check if it's a threat list finding and severity is 4 or above
        if 'UnauthorizedAccess:EC2/MaliciousIPCaller.Custom' in finding_type and severity >= 4:
            # Get the instance ID of the target instance
            instance_id = detail.get('resource', {}).get('instanceDetails', {}).get('instanceId')

            if not instance_id:
                print("Unable to find target instance ID in the event")
                return {
                    'statusCode': 400,
                    'body': json.dumps('No target instance ID found in the event')
                }

            print(f"Attempting to stop instance {instance_id}")

            # Stop the instance
            try:
                ec2.stop_instances(InstanceIds=[instance_id])
                action = "stopped"
```

CloudShell | Feedback

 Privacy | Terms | Cookie preferences

SOAR Automated Response

aws | Services | Search [Alt+S] | N. Virginia | GG1211

EC2 | VPC | Lambda | IAM | Security Hub | CloudWatch | AWS Config | GuardDuty | CloudTrail | Amazon Data Firehose | S3 | Simple Notification Service | Amazon Inspector | Amazon

GuardDuty

- Summary
- Findings
- Usage
- EC2 malware scans

▼ **Protection plans**

- S3 Protection
- EKS Protection
- Runtime Monitoring
- Malware Protection for EC2
- Malware Protection for S3 New
- RDS Protection
- Lambda Protection

- Accounts
- Settings
  - Lists

GuardDuty > Settings > Lists

List management Info

GuardDuty does not generate findings for IP addresses that are included in trusted IP lists. GuardDuty generates findings for IP addresses that are included in threat lists.

Trusted IP lists

IP addresses that are trusted for secure communication with your AWS environment. You can only have one trusted IP list per Region.

List name	List file URL	Format	Tags	Status

No trusted IP list

No trusted IP list to display

Add a trusted IP list

Threat IP lists

Actions | Add a threat IP list

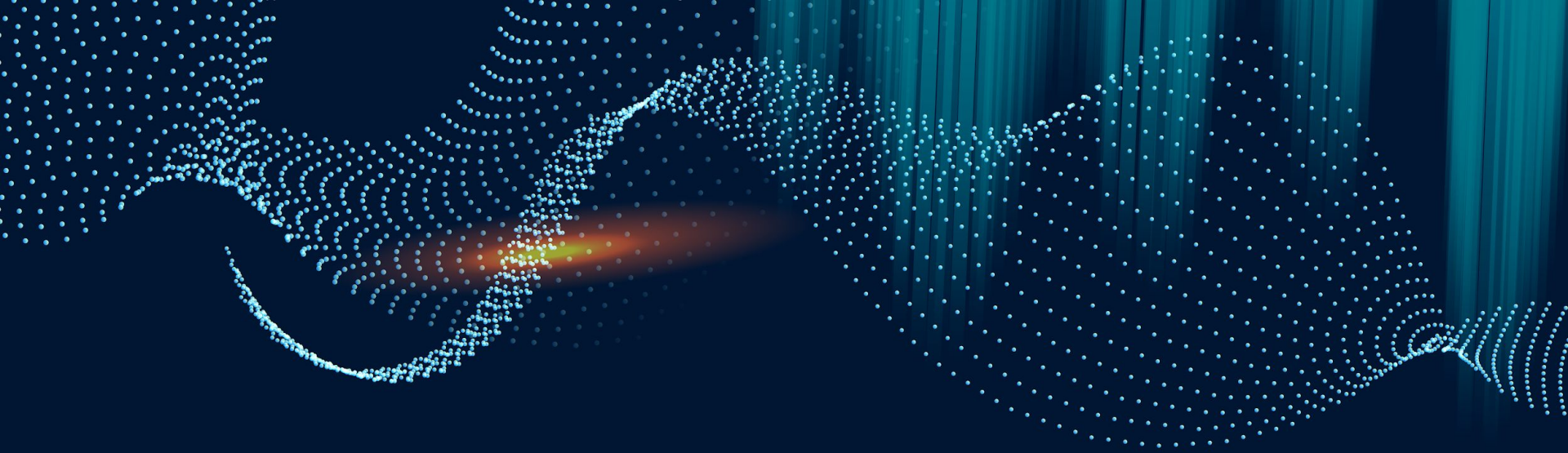
Manage known malicious IP addresses for your account. You can have up to six uploaded threat lists per AWS account per Region.

	List name	List file URL	Format	Tags	Status
○	AlienVault Malicious Addresses	s3://alienvault-maliciousip-threatlist/reputation (2).generic	ALIEN_VAULT	0	Active
○	Custom Threatlist1211	s3://threat-list1211/threat_list1.txt	TXT	0	Active

CloudShell | Feedback

 Privacy | Terms | Cookie preferences

DEMO



Outcomes & Future

Challenges, Results, and Future Enhancements

Overall Challenges

Splunk Cloud Implementation

Going through documentation and figuring out how to configure Splunk to accept AWS metrics and logs. Then learn how to use its dashboards.

Multiple Services and Logging Efficiency

Finding a way to properly collect data from across the whole infrastructure and learning new tools along the way.

SOAR Workflow

Figuring out which workflow would work best, many options for automated playbooks. Using Lambda to write a python script that worked through trial and error.

The Outcomes

A Working Security Operations Center

We created a scalable foundation for future enhancements and additional use cases.

Improved Visibility

Using AWS-native services to have better security understanding and observability.

SOAR Workflow

A demonstration of an automated workflow which was a great learning experience.

AWS Platform Navigation

Thanks to this project, we learned a lot about navigating and understanding how AWS works and added valuable insights.

Future Enhancements

There are some things we would like to look into to take this project further like:

1.) Expanding to a multi-account, multi-region, and multi-user environment

2.) Integrating DevOps practices and implementing honeypots for advanced threat detection

3.) Incorporating AI capabilities and enhancing our threat intelligence

4.) Experimenting with more advanced dashboard visualizations

5.) Implementing a ticketing system for streamlined incident management

THANK YOU

Q&A