

Attack, Detect & Secure the Environment (AWS)

Student Name: Sanjana Thawait

College: Rungta College of Engineering and Technology

B.Tech CSE Cyber Security

ERP: 6604710

5TH SEM MAJOR PROJECT

ABSTRACT

Cloud computing environments are increasingly targeted by cyber attackers due to their public accessibility. This project implements a cloud-based security monitoring solution using Amazon Web Services (AWS). Unauthorized SSH login attempts are simulated on an EC2 instance, and authentication logs are collected using AWS CloudWatch Agent. These logs are analyzed using AWS CloudWatch Logs acting as a SIEM platform. The project demonstrates attack detection, centralized logging, and practical SIEM implementation.

1. INTRODUCTION

Cloud computing has become the backbone of modern IT infrastructure. Organizations rely on platforms such as Amazon Web Services (AWS) for scalable and cost-effective solutions. However, publicly accessible cloud servers are frequent targets of cyber-attacks. One of the most commonly targeted services is Secure Shell (SSH), which attackers exploit using brute-force techniques. Security Information and Event Management (SIEM) systems help detect such threats by collecting and analyzing logs. AWS CloudWatch provides native monitoring capabilities that can be used as a lightweight SIEM solution. This project demonstrates real-world attack simulation and detection using AWS CloudWatch.

2. OBJECTIVES OF THE PROJECT

The primary objective of this project is to design and implement a cloud-based monitoring solution capable of detecting unauthorized access attempts. The project simulates SSH attacks, configures CloudWatch Agent

for log collection, and analyzes logs using AWS CloudWatch. The project bridges theoretical security concepts with practical cloud implementation.

3. TOOLS AND TECHNOLOGIES USED

1. Amazon Web Services (AWS):

AWS provides the cloud infrastructure used in this project. It enables the deployment of virtual machines, monitoring services, and secure access control. AWS allows simulation of real-world attack scenarios in a controlled environment.

2. Amazon EC2:

EC2 is used to deploy attacker and victim virtual machines. The victim instance runs the SSH service and generates authentication logs, while the attacker instance simulates unauthorized access attempts.

3. AWS IAM:

IAM is used to manage permissions securely. An IAM role is attached to the EC2 instance to allow CloudWatch Agent to send logs to CloudWatch without exposing credentials.

4. AWS CloudWatch:

CloudWatch acts as the SIEM platform. It collects, stores, and displays authentication logs in log groups and log streams, enabling centralized monitoring and attack detection.

5. CloudWatch Agent:

The agent runs on the victim EC2 instance and forwards system logs such as `/var/log/auth.log` to CloudWatch in near real time.

6. Ubuntu Linux:

Ubuntu is used as the victim operating system. It generates SSH authentication logs whenever login attempts occur.

7. SSH:

SSH is the target service. Failed login attempts are used as indicators of attack activity.

4. Screenshot-wise Explanation

The screenshot shows the AWS EC2 Instances page with the following details:

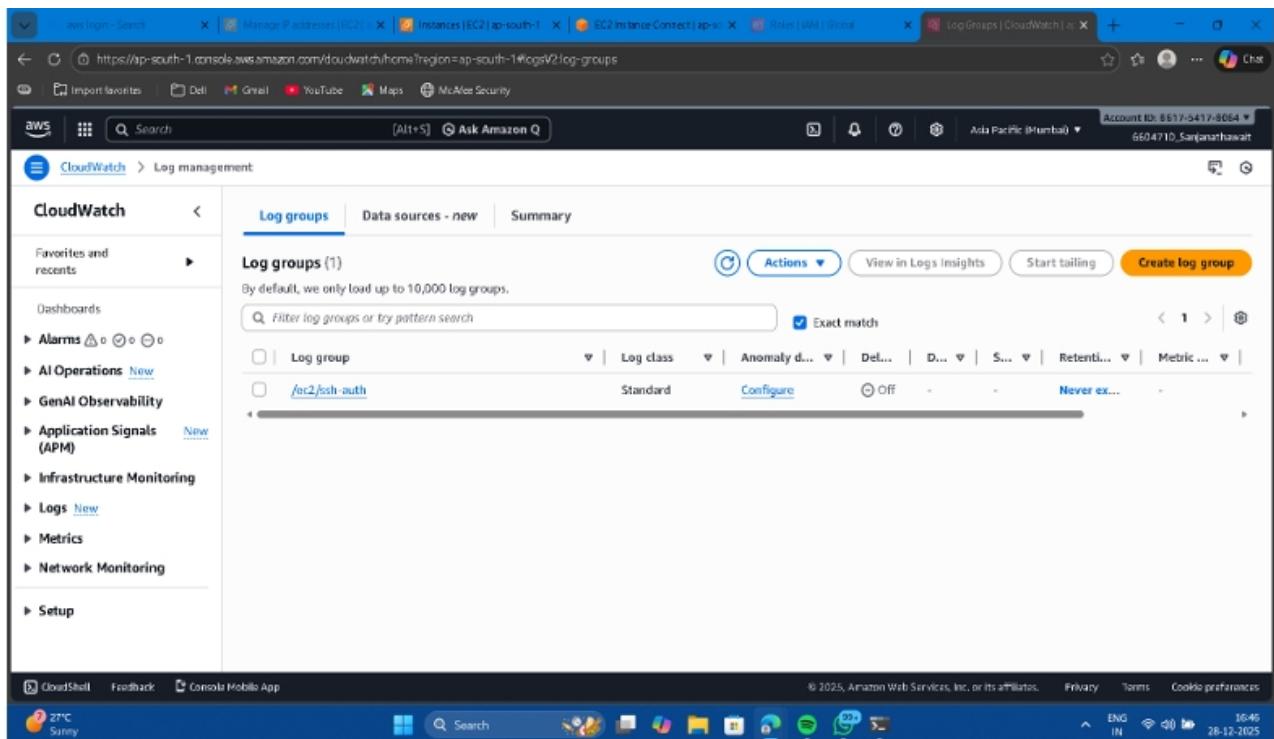
- Region:** ap-south-1
- Instances:** 6
- Instance Types:** t3.micro
- Status:** All are Running
- Health:** All 3/3 checks pass
- Availability Zone:** ap-south-1a
- Public IPv4:** Available for all instances

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4
Hexarootsystem	i-0198b3c34699a283a	Running	t3.micro	3/3 checks pass	View alarms +	ap-south-1a	-
Hexarootsystem	i-058ae8bdb40a5807	Running	t3.micro	3/3 checks pass	View alarms +	ap-south-1a	-
Hexarootsystem	i-0ccf70b402b0b931d	Running	t3.micro	3/3 checks pass	View alarms +	ap-south-1a	-
Attacker-Ubu...	i-076b0088d2c3766e5	Running	t3.micro	3/3 checks pass	View alarms +	ap-south-1a	-
VM1-Public-Test	i-0e91a6d69639b1004	Running	t3.micro	3/3 checks pass	View alarms +	ap-south-1a	-
Wazuh-Manager	i-0941e0856b15c92ee	Running	t3.micro	3/3 checks pass	View alarms +	ap-south-1a	-

EC2 Instances Overview

This screenshot shows multiple EC2 instances running in the AWS Mumbai region. It includes the attacker machine, victim machine, and Wazuh manager instance. This setup forms the foundation of the cloud security lab environment

Attack Simulation from Attacker VM



This screenshot demonstrates SSH attack attempts initiated from the attacker virtual machine. Multiple invalid login attempts are generated to simulate a brute-force attack scenario.

CloudWatch Log Group Creation

The screenshot shows the AWS CloudWatch Log Management interface. On the left, there's a sidebar with various navigation options like Alarms, AI Operations, GenAI Observability, Application Signals (APM), Infrastructure Monitoring, Logs, Metrics, Network Monitoring, and Setup. The main area is titled '/ec2/ssh-auth' under 'Log management'. It displays 'Log group details' with fields for Log class (Standard), ARN (arn:aws:logs:ap-south-1:061754178064:log-group:/ec2/ssh-auth), Creation time (26 minutes ago), Retention (Never expire), and Stored bytes (~). To the right, there are sections for Metric filters (0), Subscription filters (0), Contributor Insights rules (none), KMS key ID (none), and Deletion protection (Off). Further right are sections for Data protection (none), Sensitive data count (none), Custom field indexes (Configure), Transformer (Configure), and Anomaly detection (Configure). At the bottom, there are tabs for Log streams (selected), Tags, Anomaly detection, Metric filters, Subscription filters, Contributor Insights, Data protection, and Field in. Below these tabs, it says 'Log streams (1)' and shows a single stream with a delete button and a 'Create log stream' button. The status bar at the bottom indicates it's 27°C and sunny.

This screenshot shows the creation of the CloudWatch log group '/ec2/ssh-auth'. The log group stores authentication logs collected from the victim EC2 instance

SSH Authentication Logs in CloudWatch

The screenshot shows the AWS CloudWatch Log Events interface. The sidebar is identical to the previous screenshot. The main area is titled 'Log events' under 'Log management' for the '/ec2/ssh-auth' log group, with the identifier 'i-0e91a6d69639b1004'. It features a search bar with 'Filter events - press enter to search' and a time range selector with 'Custom' (selected) and 'UTC timezone'. Below this is a 'Display' dropdown. The log events table has columns for 'Timestamp' and 'Message'. The messages listed are:

- Dec 28 11:00:35 ip-10-0-2-225 sshd[3176]: Connection closed by invalid user wronguser@127.0.0.1 port 47150 [preauth]
- Dec 28 11:00:46 ip-10-0-2-225 sshd[3180]: Invalid user test from 127.0.0.1 port 35936
- Dec 28 11:00:46 ip-10-0-2-225 sshd[3180]: Connection closed by invalid user test 127.0.0.1 port 35936 [preauth]
- Dec 28 11:01:00 ip-10-0-2-225 sudo: ubuntu : TTY=pts/0 ; PWD=/home/ubuntu ; USER=root ; COMMAND=/usr/bin/tail -n 20 /var/log/auth.log
- Dec 28 11:01:00 ip-10-0-2-225 sudo pam_unix(sudo:session): session opened for user root (uid=0) by ubuntu(uid=1000)
- Dec 28 11:01:00 ip-10-0-2-225 sudo pam_unix(sudo:session): session closed for user root
- Dec 28 11:03:05 ip-10-0-2-225 sshd[3187]: error: key_exchange_identification: banner line contains invalid characters
- Dec 28 11:03:05 ip-10-0-2-225 sshd[3187]: banner exchange: Connection from 41.231.36.197 port 35078: invalid format
- Dec 28 11:03:20 ip-10-0-2-225 sshd[3188]: Invalid user wqmerldulqjgms from 41.231.36.197 port 49988
- Dec 28 11:03:20 ip-10-0-2-225 sshd[3188]: fatal: userauth_pubkey: parse publickey packet: incomplete message [preauth]

At the bottom, it says 'No newer events at this moment. Auto retry paused. Resume' and has a 'Back to top' button. The status bar at the bottom indicates it's 27°C and sunny.

This screenshot displays SSH authentication logs in CloudWatch. Events such as invalid users, failed login attempts, and connection resets are visible, confirming successful attack detection.

5. Conclusion

This project successfully demonstrates the implementation of a cloud-based SIEM solution using AWS CloudWatch. By simulating real-world SSH attacks and analyzing authentication logs, the project highlights the importance of centralized logging and continuous monitoring in cloud environments. The results confirm that AWS CloudWatch can effectively detect unauthorized access attempts and provide valuable security insights. This project enhances practical understanding of cloud security concepts and prepares students for real-world cybersecurity challenges

6. FUTURE SCOPE

Future enhancements of this project include configuring automated alerts using CloudWatch Alarms, integrating AWS GuardDuty for advanced threat detection, and implementing automated incident response using AWS Lambda. The project can also be extended by integrating enterprise-grade SIEM tools such as Wazuh or ELK Stack for large-scale environments.