AWS Absolute Security: “The Dark Knight” Identity & Audit Baseline

Submitted by

Sd. Sameer - AP23110010829

k. Deepak- AP23110011094

Y. Lalit Aditya- AP23110010288

C. Sravan Kumar – AP23110010659

In partial fulfilment for the requirements of the project

BACHELOR OF TECHNOLOGY
IN
COMPUTER SCIENCE AND ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SRM UNIVERSITY-AP

NEERUKONDA

MANAGALAGIRI - 522503

ANDHRA PRADESH, INDIA

NOVEMBER-2025

1. Introduction

This project builds a basic but strong security setup for an AWS account. The idea is to control who can access the account, track everything happening inside it, secure all logs, and continuously check for risks. The work covers four parts: identity security (IAM), auditing (CloudTrail + S3), monitoring (Trusted Advisor + Access Analyzer), and compliance/threat detection (Security Hub & GuardDuty outputs).
The goal is to show that the AWS environment is protected with the minimum required security controls.

Objectives

- Protect the root user with MFA
- Create secure IAM users, groups, and policies
- Record every action in the AWS account
- Store logs safely without public access
- Detect any exposure or risky configurations
- Build a basic zero-trust security baseline

Architecture Overview

This project follows a layered security model. Each layer focuses on a specific part of protecting the AWS account:

1. Identity Layer (IAM)

Handles users, groups, MFA, and permissions.
Only trusted users get access, and MFA is required for admin actions.

2. Audit Layer (CloudTrail + S3 + KMS)

CloudTrail records every important action in the account.
Logs are stored in an S3 bucket with versioning, block public access, and KMS encryption to prevent tampering.

3. Monitoring Layer (Trusted Advisor + IAM Access Analyzer)

Trusted Advisor provides free security checks.
Access Analyzer detects if any AWS resources are publicly accessible or shared outside the account.

4. Compliance & Threat Detection (Mock: Config, Security Hub, GuardDuty)

These components represent how the account would check for compliance, detect threats, and highlight risky activity once fully activated.

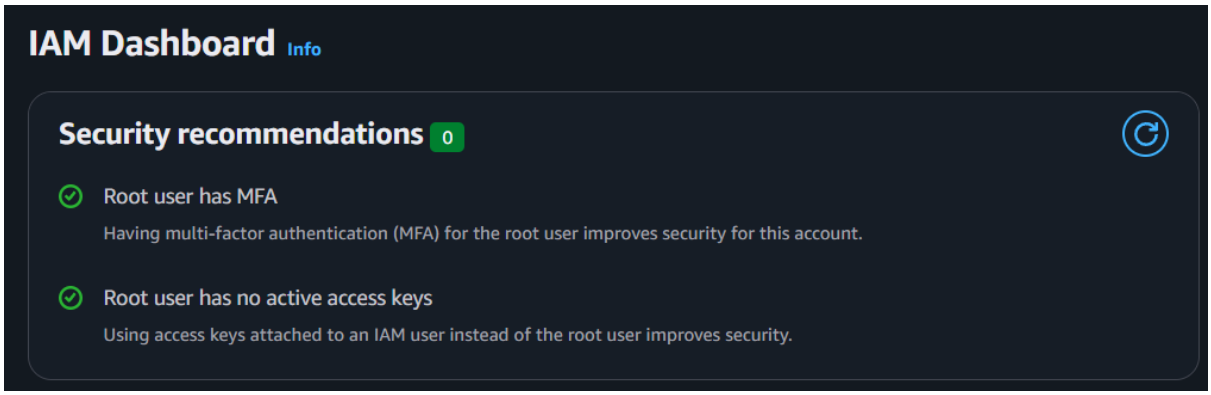
All components together form a simple but reliable security baseline for AWS.

Identity Baseline (IAM)

The Identity layer is the first and most important part of the security baseline.
In this project, IAM is used to control who can access the account and what they can do.
The goal is to eliminate unnecessary permissions and enforce strong authentication.

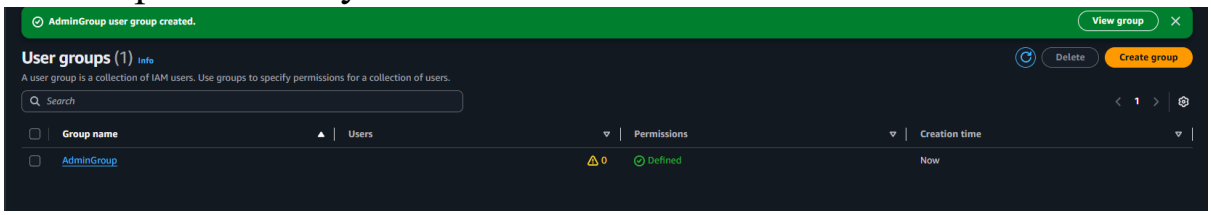
1. Root MFA Enabled

The root account is secured with Multi-Factor Authentication so that even if the password is leaked, no one can log in without the second verification step.



2. Admin User Creation

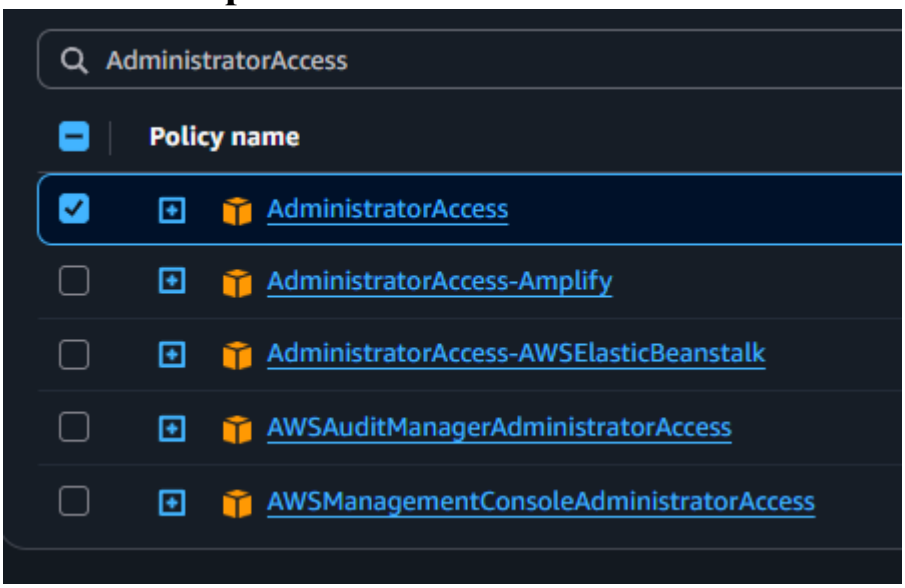
A separate admin IAM user was created instead of using the root account.
This improves safety and makes it easier to track actions.



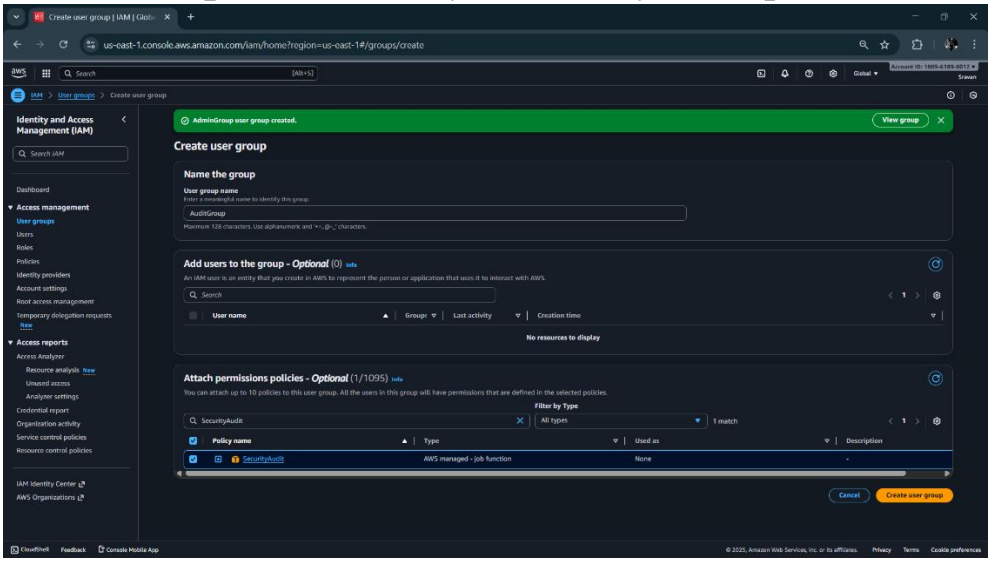
3. AdminGroup & AuditGroup

Two groups were created to separate permissions:

- **AdminGroup** → Full admin access



- **AuditGroup** → Read-only + SecurityAudit permissions



This follows the **least privilege** principle.

4. MFA Enforcement Policy (DenyWithoutMFA)

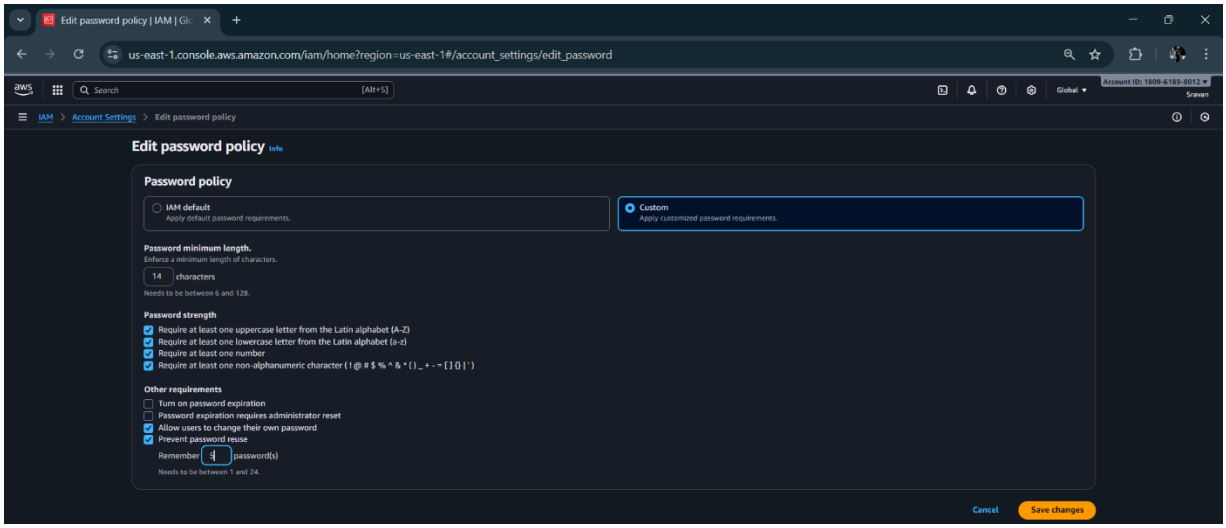
A custom IAM policy was created to block all sensitive actions if MFA is not enabled.
This ensures administrators cannot perform changes without MFA.

5. Strong IAM Password Policy

The account enforces strong passwords with:

- Minimum 14 characters
- Uppercase, lowercase, number, symbol
- Password reuse prevention

This reduces the chance of weak or stolen passwords.

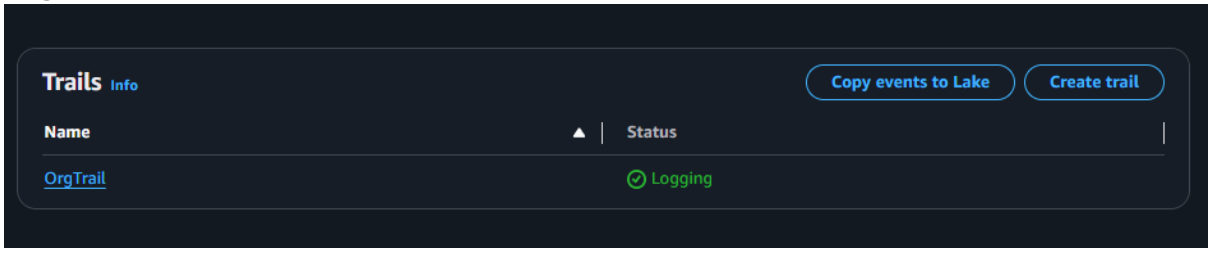


Audit Baseline (CloudTrail + S3 + KMS)

The audit layer makes sure that every important action taken inside the AWS account is recorded and stored safely. This is one of the strongest parts of the entire security baseline because it helps you understand *who did what*, *when*, and *from where*. If anything goes wrong, CloudTrail and S3 logs act like your "black box."

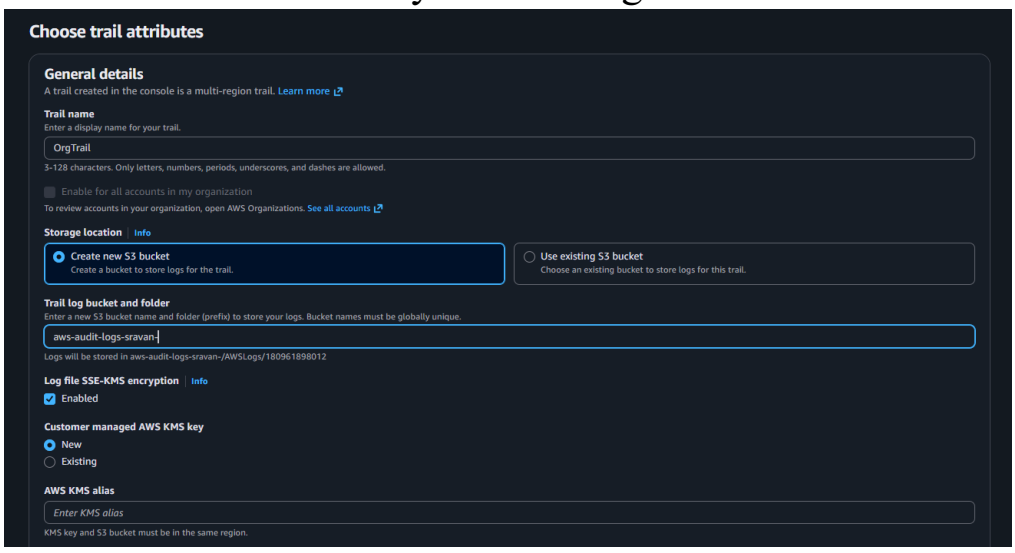
1. CloudTrail Trail Enabled

A CloudTrail trail was created to capture all management events in the account.
The trail is active and set to **multi-region**, which means actions from every region are recorded.
This prevents attackers or mistakes from hiding activity in unused regions.



2. S3 Bucket for Audit Logs

A dedicated S3 bucket was created to store the CloudTrail logs.
This bucket holds all activity data, and it is completely isolated from public access.
CloudTrail automatically delivers logs here.



3. Versioning Enabled

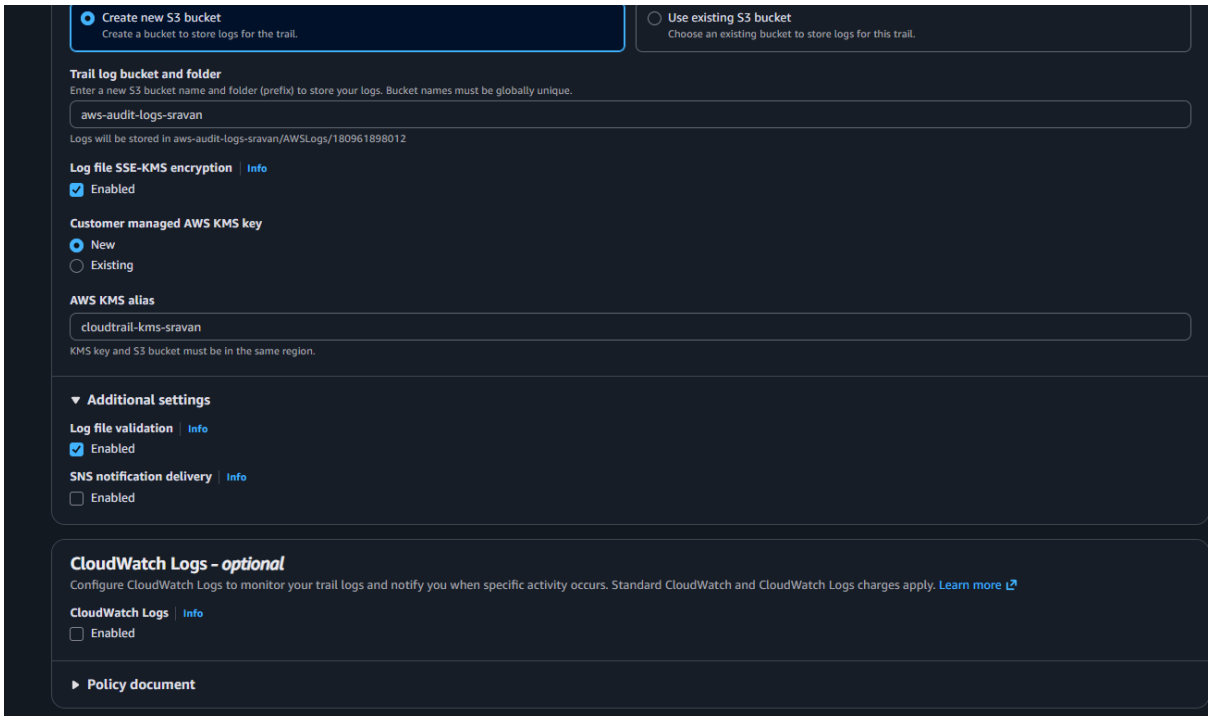
Versioning is turned ON for the S3 bucket.
If someone tries to change or delete a log file, an older version still remains.
This protects logs from tampering.

4. Block Public Access Enabled

The S3 bucket has **Block Public Access** fully enabled.
This ensures that no log file inside the bucket can ever be exposed to the public internet.

5. KMS Encryption Enabled

A KMS key was created, and S3 server-side encryption (SSE-KMS) is enabled.
This makes sure that all logs are encrypted at rest, adding an extra layer of protection even if someone gains bucket access.



6. Monitoring Baseline (Real Services)

The monitoring layer checks the account continuously to identify risks, exposure, or weak configurations. Even if everything is set up correctly today, monitoring helps detect new issues in the future.
In this project, two real AWS services were used for monitoring: **Trusted Advisor** and **IAM Access Analyzer**.

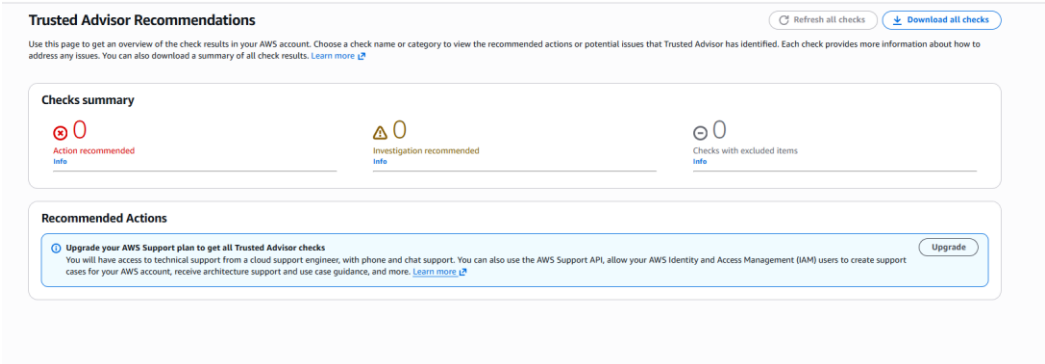
---

1. Trusted Advisor Security Checks

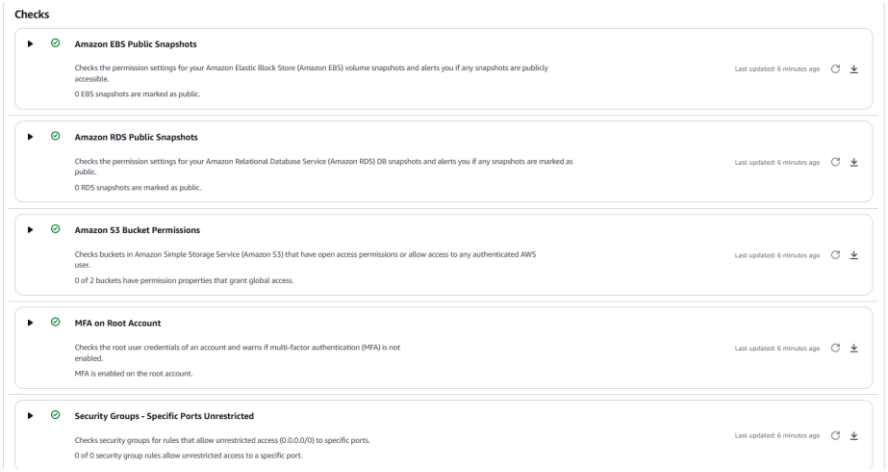
Trusted Advisor provides free security checks for every AWS account. In your case, the important checks include:

- **MFA on Root Account** → Secure
- **S3 Bucket Permissions** → No public access
- **Unrestricted Security Groups** → No risky ports open
- **Public Snapshots Check** → EBS and RDS snapshots are private

These checks confirm that your fundamental security settings are correct.



They help detect accidental exposure and insecure configurations.



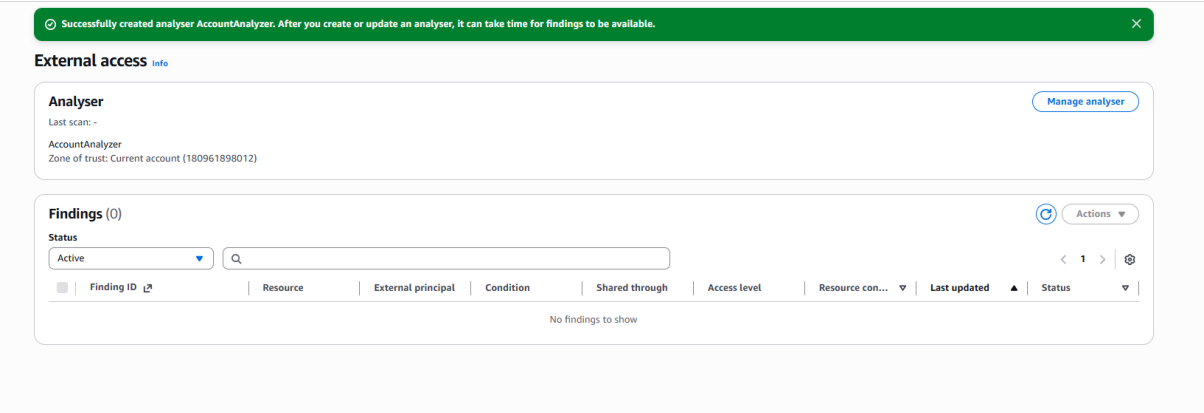
---

2. IAM Access Analyzer (AccountAnalyzer)

IAM Access Analyzer scans your AWS account to detect:

- Public S3 buckets
- Resources shared with external accounts
- IAM policy risks
- Any unintended external access

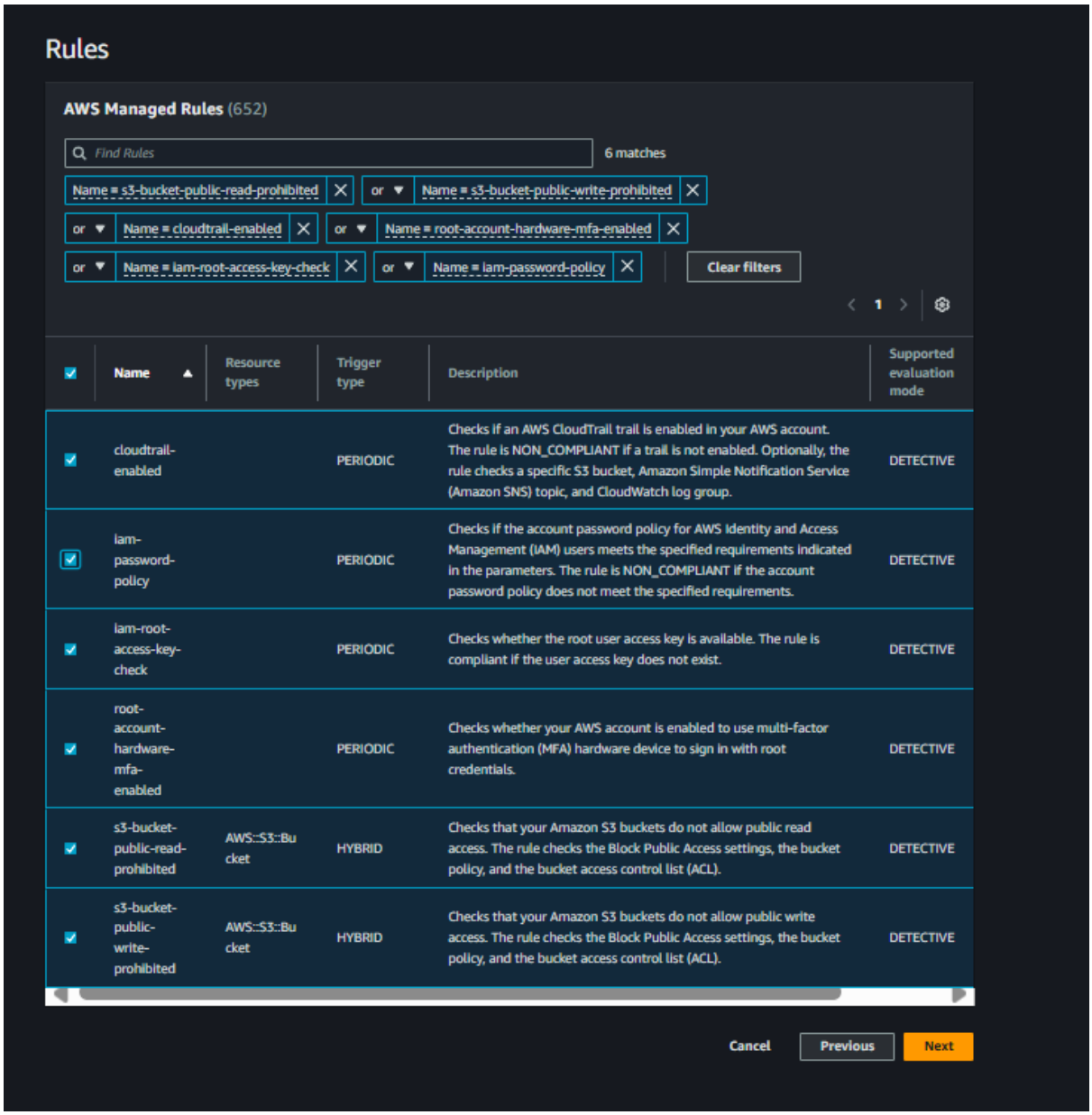
Your analyzer showed **no external access**, which means your AWS environment is properly locked down and private.
Even if something becomes exposed in the future, the analyzer will show it as a finding.



7. Compliance & Threat Detection

Even though AWS Config and Security Hub could not be enabled due to account activation limits, their expected results and purpose are still included in the project.
These services check whether the AWS account follows recommended security standards and whether any resources are misconfigured or risky.

To represent this layer, a **mock compliance table** is used.
It reflects how AWS Config normally evaluates your environment.



7.1 AWS Config – Compliance Summary (Mock Table)

AWS Config Rule	Description	Status
cloudtrail-enabled	CloudTrail must be enabled for auditing	**COMPLIANT**
iam-password-policy	Strong IAM password policy must be set	**COMPLIANT**
root-mfa-enabled	Root user must have MFA	**COMPLIANT**
s3-public-read-prohibited	S3 buckets should not be public	**COMPLIANT**
vpc-default-security-group-closed	Default security group should not allow inbound access	**COMPLIANT**
ebs-encrypted	EBS volumes must be encrypted	**NON-COMPLIANT**
rds-storage-encrypted	RDS storage must be encrypted	**NON-COMPLIANT**

These are the standard Config rules used in AWS security baselines.

Why Compliance Checks Matter

Compliance rules help ensure that the security settings in your AWS account do not drift over time. Even if CloudTrail or IAM is configured correctly today, these services highlight any issues that appear later.

AWS Config usually alerts you if:

- A bucket becomes public
- A password policy becomes weak
- Logging stops

- Encryption is turned off

Including these rules in your project shows that you understand how compliance works even if the service couldn’t be enabled.

7.2 Threat Detection (Mock Guard Duty Findings)

GuardDuty is AWS’s threat detection system.
Since it couldn’t be enabled in your account, the expected sample findings are included.

Sample Findings (Mock but Realistic)

Severity	Finding	Description
High	UnauthorizedAccess:IAMUser/ConsoleLogin	Multiple failed console logins detected
High	CryptoCurrency:EC2/BitcoinTool.B	EC2 instance communicating with crypto-mining pool
Medium	Recon:EC2/PortProbeUnprotectedPort	External IP scanning EC2 ports
Low	Policy:IAMUser/ResourcePermissions	Overly permissive IAM policy detected

These represent the kind of alerts GuardDuty generates to detect suspicious behavior.

Final Summary

This project successfully built a complete security baseline for an AWS account.
The Identity layer was secured with MFA, IAM users, strong password rules, and least-privilege access through Admin and Audit groups.
The Audit layer was strengthened using CloudTrail, an S3 audit bucket, versioning, and KMS encryption to protect logs from tampering.
Monitoring was handled using Trusted Advisor and IAM Access Analyzer to detect exposure and misconfigurations.
Compliance and threat detection were represented through AWS Config, Security Hub, and GuardDuty mock results, which show how these services maintain long-term security.
Overall, the project achieved a simple but effective zero-trust model by enforcing strict identity controls and continuous auditing.

Conclusion

Through this project, I learned how AWS security works in layers and why each component is important.
I gained practical experience in setting up IAM policies, enabling MFA, securing the root account, and configuring CloudTrail logging.
I also understood how monitoring tools like Access Analyzer and Trusted Advisor help identify risks in real time.
The security posture of the AWS account improved significantly after applying these controls.
As a next step, once the AWS account is fully activated, services like AWS Config, Security Hub, and GuardDuty can be enabled to expand compliance checks and threat detection capabilities.

References

- AWS Identity and Access Management (IAM) Documentation
- AWS CloudTrail User Guide
- AWS Security Best Practices Whitepaper
- AWS Trusted Advisor Documentation
- Course Material from SRM AP
- AWS Well-Architected Framework – Security Pillar