

Cloud Risk Assessment Team Project

By: Aat Morrison, Abed Salekin Justin Min,
Selam Tekle, Olumide Solomon

Agenda

Introduction

About CSPM and CNAPP

Compliance Framework

NIST 800-171 and CIS AWS Foundation Benchmark

Risk and Risk Matrix

Risk Mitigation

Prowler and Qualys TotalCloud Failed Findings

Mitigation Plan/Schedule

Challenges Faced

•Introductions

:

**Aat Morrison
Abed Salekin
Justin Min
Olumide Solomon
Selam Tekle**



What is CSPM and CNAPP?

CSPM- Cloud Posture Management Tool:

- Finds, monitors, and fixes security issues in your cloud setup like AWS
- The main goal is compliance, visibility, and security posture
- Examples of some are Prowler, Prisma Cloud, and AWS Security Hub

CNAPP- Cloud Native Application Protection Platform:

- Protects your whole cloud application lifecycle — from code to runtime — using multiple security tools together, like AWS Config or CloudFormation Guard
- The main goal is end-to-end security from development to production
- Examples of some are Qualys Totalcloud, Wiz, Palo Alto Prisma Cloud

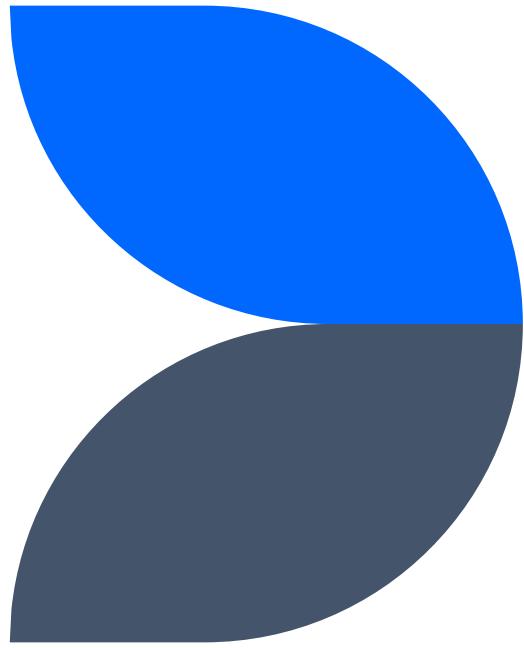
CSPM and CNAPP Comparison Table

Category	CSPM (Cloud Security Posture Management)	CNAPP (Cloud-Native Application Protection Platform)
Purpose	Finds and fixes cloud setup issues like wrong permissions or open storage (example: Prowler)	Secures the whole cloud app, from code to running systems, using multiple tools (example: TotalCloud)
Scope	Focuses on cloud infrastructure like S3, IAM, and VPC	Covers cloud infrastructure, workloads, CI/CD pipelines, and runtime environments
Deployment	Agentless — connects through cloud provider APIs	Mix of agentless scanning and agents inside workloads
Capabilities	Runs config checks, audits, and sends alerts for misconfigurations	Combines CSPM, workload protection (CWPP), identity management (CIEM), and code scanning (IaC)
Frameworks	Follows security standards like CIS, NIST, and ISO	Covers full development lifecycle (SDLC) and runtime security with DevSecOps integration

What is Compliance Framework?

A compliance framework is a structure of guidelines and best practices that organizations use to ensure the company meets security, privacy, and operational standards (e.g., NIST, ISO, SOC2).

Helps organizations prove they are handling data and risks properly.



About NIST 800-171

- NIST 800-171 is a cybersecurity framework developed by the National Institute of Standards and Technology (NIST)
- Its **focus** is on protecting Controlled Unclassified Information (CUI) in non-federal systems and organizations
- Its primary goal is to ensure sensitive government data remains secure when handled by contractors, universities, or private companies working with federal agencies.
- It provides a structured set of security requirements tailored for environments outside the federal government.

Key features:

- 14 Control Families
- 110 Individual Controls
- Focus on access control, incident response, system integrity, and more.

NIST 800-171 Control Families

1. Access Control (AC)
2. Awareness and Training (AT)
3. Audit and Accountability (AU)
4. Configuration Management (CM)
5. Identification and Authentication (IA)
6. Incident Response (IR)
7. Maintenance (MA)
8. Media Protection (MP)
9. Personnel Security (PS)
10. Physical Protection (PE)
11. Risk Assessment (RA)

CIS Amazon Web Service Foundation Benchmark

Control Families:

1. Identify and Access Management
2. Storage
3. Logging
4. Monitoring
5. Networking

Overview:

All CIS Benchmarks™ (Benchmarks) focus on technical configuration settings used to maintain and/or increase the security of the addressed technology, and they should be used in conjunction with other essential cyber hygiene tasks.

Risk is the possibility that something bad could happen that might hurt a company, a system, or people. In cybersecurity, risk happens when there is a weakness (called a vulnerability) in a system that could be attacked by a threat.

What is Risk?



How is it Calculated?

Risk = Likelihood × Impact

- **Likelihood:** How likely is it that something bad will happen?
- **Impact:** How bad would it be if it did happen? (e.g., data breach, service outage).

$$\text{risk} = \text{likelihood} \times \text{impact}$$

impact = g(business criticality)

likelihood = f(vulnerabilities, exposure, threats, mitigating controls)

Possible	Major	Major	Major
Likely	Moderate	Major	Major
Rare	Minor	Moderate	Major
	Low	Medium	High

3x3 Qualys Totalcloud Finding Matrix

Risk Matrix

4x4 NIST 800-171 Finding Matrix

Probable	Critical	Critical	Critical	Critical
Possible	Major	Critical	Critical	Critical
Unlikely	Moderate	Major	Critical	Critical
Rare	Minor	Moderate	Major	Critical
	Low	Medium	High	Very High

What is Risk Mitigation?

Reducing the likelihood of a risk occurring or lessening its impact if it does occur is known as risk mitigation.



AVOIDANCE

-Eliminate the risk completely
Ex. Not storing sensitive data in cloud



REDUCTION

-Lessen the likelihood/impact
Ex. Enabling MFA to reduce breach risk



TRANSFEREN

-Shift risk to third-party
Ex. Buying cyber insurance



ACCEPTAN

CE
-Acknowledge and monitor the risk
Ex. Accept minor risks of low impact

Prowler Finding 1:

Probable	Critical	Critical	Critical	Critical
Possible	Major	Critical	Critical	Critical
Unlikely	Moderate	Major	Critical	Critical
Rare	Minor	Moderate	Major	Critical
	Low	Medium	High	Very High

<i>Finding Name and Title</i>	iam_user_mfa_enabled_console_access Ensure multi-factor authentication (MFA) is enabled for all IAM users that have a console password
<i>Service</i>	IAM
<i>Risk, Severity, and Risk Mitigation Type</i>	Unauthorized users could access AWS accounts if only a password is required (no MFA) Risk Equation: Likelihood × Impact Likelihood: Possible – password based accounts are a frequent target for attacks Impact: High – account compromise could lead to full environment control
<i>Recommended Mitigation</i>	Type of Mitigation: The recommended risk mitigation type is Avoidance Why Risk Avoidance? Requiring MFA reduces the chance of unauthorized access close to zero, directly addressing both the likelihood and impact. Enforce MFA for all IAM users through IAM policies and AWS Configuration rules
<i>NIST 800-171 Family and Control</i>	Access Control (AC) → 3.1.1, 3.1.2, 3.1.3, 3.1.14, 3.1.20 Identification and Authentication → 3.5.2, 3.5.3
<i>Cross Framework Mapping</i>	NIST 800-53 - Mapped? Yes ISO 27001:2013 - Mapped? Yes SOC2 - Mapped? Yes

FAIL	high	iam	us-east-1	iam_user_mfa_enabled_console_access	Ensure multi-factor authentication (MFA) is enabled for all IAM users that have a console password.	User admin-tram has Console Password enabled but MFA disabled.	Unauthorized access to this critical account if password is not secure or it is disclosed in any way.	Enable MFA for the user's account. MFA is a simple best practice that adds an extra layer of protection on top of your user name and password. Recommended to use hardware keys over virtual MFA.	-CISA: your-systems-3, your-surroundings-2, booting-up-thing-to-do-first-2-FedRAMP-Low-Revision-4: ac-2, ia-2 -CIS-14: 1.10 -PCI-3.2.1: 8.3, 8.3.1, 8.3.1.a, 8.3.2, 8.3.2.a, 8.6, 8.6.c -KISA-ISMS-P-2023: 2.5.3, 2.6.1, 2.6.6 -KISA-ISMS-P-2023-korean: 2.5.3, 2.6.1, 2.6.6 -FFIEC: d3-pc-am-b-15, d3-pc-am-b-6 -MITRE-ATTACK: T1078, T1098, T1556, T1550, T1110, T1040, T1538 -CG-2.0: 1.10 -AWS-Well-Architected-Framework-Security-Pillar: SEC02-BP01 -AWS-Foundational-Security-Best-Practices: IAM.5, IAM.19 -NIST-800-53-Revision-4: ia_2.1, ia_2.2, ia_2.1f -FedRamp-Moderate-Revision-4: ac-2-1, ac-2-f, ac-2-i; ia-2.1, ia-2.1, ia-2.1 -NIST-800-53-Revision-5: ac_2.1, ac_3.2, ac_3.3, ac_3.3.a, ac_3.3.b., ac_3.3.b, ac_3.3.b.3, ac_3.3.b.4, ac_3.3.b.5, ac_3.3.c, ac_3.4, ac_3.4.a, ac_3.4.b, ac_3.4.c, ac_3.4.d, ac_3.4.e, ac_3.8, ac_3.12.a, ac_3.13, ac_3.15.a, ac_3.15.b, ac_4.28, ac_7.4, ac_7.4.a, ac_24, cm_5.j.a, cm_6.g, cm_9.b, ia_2.1, ia_2.2, ia_2.6, ia_2.6.b, ia_2.8, sc_23.3 -OxP-21-CFR-Part-11: 11.10-d, 11.10-g, 11.200 -NIST-800-171-Revision-2: 3.1.1, 3.1.2, 3.1.14, 3.5.2, 3.5.3 -GDPR article_25 -AWS-Foundational-Technical-Review: IAM-001, IAM-0012 -ENS-RD2022: op.acc.8.2.aws.iam.1, op.acc.8.4.aws.iam.1, op.acc.8.6.aws.iam.1 -CIS-3.0: 1.10 -HIPAA: 164_308_a_3_i_a, 164_312_a_a, 164_312_d -ISO27001-2013: A.9.2, A.9.3, A.9.4 -NIST-CSF-11: ac_3, ac_7 -PCI-4.0: 8.4.1.1, 8.4.1.2, 8.4.2.1, 8.4.2.2, 8.4.3.1, 8.4.3.2 -ISO27001-2022: A.5.15, A.5.17, A.8.5 -CIS-1.0: 1.10 -CIS-4.0.1: 1.10 -AWS-Audit-Manager-Control-Tower-Guardrails: 3.0.1, 3.0.2, 3.0.3
					Prowler Finding 1				

TotalCloud Finding 1:

Possible	Major	Major	Major
Likely	Moderate	Major	Major
Rare	Minor	Moderate	Major
	Low	Medium	High

Finding Name and Title	CID-32 - Ensure AWS Management Console authentication failures are monitored
Resource and Service	CloudTrail / CloudWatch
Risk, Severity, and Risk Mitigation Type	<p>The AWS environment is not currently monitoring authentication failures on the AWS Management Console. This lack of visibility means failed login attempts by unauthorized users may go undetected, increasing the likelihood of successful brute-force or credential stuffing attacks.</p> <p>Risk Equation: Likelihood x Impact</p>
Likelihood	<p>Likelihood: Likely - Unauthorized access attempts via the AWS Management Console are a common vector for intrusion and often go unnoticed if not monitored.</p> <p>Impact: Medium (potential data breach, delayed response, non-compliance)</p>
Recommended Mitigation	<p>Type of Mitigation? The recommended mitigation is remediation.</p> <p>Why? This is a configuration issue that can be remediated by enabling log monitoring for authentication failures. Specifically, ensure that AWS CloudTrail is logging management events and that CloudWatch Alarms or SIEM tools are configured to alert on failed login attempts.</p>
CIS AWS Foundation Benchmark	<ul style="list-style-type: none"> ● Control 4.6 - Ensure AWS Management Console authentication failures are monitored ● Profile Applicability: Level 2 ● Rationale: Monitoring failed console logins can detect brute-force attacks and provide actionable indicators like source IPs. <p>Audit: Ensure an active multi-region CloudTrail trail is configured, and CloudWatch metric filters and alarms exist for ConsoleLogin failures.</p>

CID-32 Ensure AWS Management Console authentication failures are monitored

[View Less](#)

Policy: [NIST Special Publication 800-171 rev 2](#) [1 more](#)

Platform: AWS

Evaluation: Check CloudWatch alarm exists for cloud trail events to monitor management console authentication failures This control checks for: Account h...

Service: CloudTrail

Manual Remediation: [View Steps](#)

Criticality: Medium

TotalCloud Finding
1

The screenshot shows a modal window titled "EXECL TUTORIAL REMEDIATION STEPS" with the sub-instruction "Follow these steps to remediate". The content provides a step-by-step guide for setting up a metric filter, alarm, SNS topic, and subscription via the AWS CLI. It includes specific commands and notes about choosing metric names and namespaces.

Policy: NIST Special Publication 800-171 rev 2 [1 more](#)

Evaluation: Check CloudWatch alarm exists for cloud trail events to monitor management console authentication failures This control checks for: Account h...

Manual Remediation: [View Steps](#)

Platform: AWS

Service: CloudTrail

Method1: Via CLI

1. Create a metric filter based on filter pattern provided which checks for unauthorized API calls and the <cloudtrail_log_group_name>
 - # aws logs put-metric-filter --log-group-name <cloudtrail_log_group_name> --filter-name <Name_for_metric> --metric-transformations metricName=<Name_for_metric>,metricNamespace='CISBenchmark',metricValue=1 --filter-pattern '\$(eventName = ConsoleLogin) && (\$errorMessage = "Failed authentication")'
 - Note:** You can choose your own metricName and metricNamespace strings. Using the same metricNamespace for all Foundations Benchmark metrics will group them together.
2. Create an SNS topic that the alarm will notify
 - # aws sns create-topic --name <sns_topic_name>
 - Note:** you can execute this command once and then re-use the same topic for all monitoring alarms.
3. Create an SNS subscription to the topic created in step 2 aws sns subscribe --topic-arn <sns_topic_arn> --protocol <protocol_for_sns> --notification-endpoint <sns_subscription_endpoints>
 - Note:** you can execute this command once and then re-use the SNS subscription for all monitoring alarms.
4. Create an alarm that is associated with the CloudWatch Logs Metric Filter created in step 1 and an SNS topic created in step 2
 - # aws cloudwatch put-metric-alarm --alarm-name <Name_for_alarm> --metric-name <Name_for_metric> --statistic Sum --period 300 --threshold 1 --comparison-operator GreaterThanOrEqualToThreshold --evaluation-periods 1 --namespace 'CISBenchmark' --alarm-actions <sns_topic_arn>
 - Note:** set the period and threshold to values that fit your organization.

OK

Prowler Finding 2:

Probable	Critical	Critical	Critical	Critical
Possible	Major	Critical	Critical	Critical
Unlikely	Moderate	Major	Critical	Critical
Rare	Minor	Moderate	Major	Critical
	Low	Medium	High	Very High

<i>Finding Name and Title</i>	s3 account level public access blocks Check S3 Account Level Public Access Block
<i>Service</i>	IAM
<i>Risk, Severity, and Risk Mitigation Type</i>	Public access policies may be applied to sensitive data buckets Risk Equation: Likelihood × Impact Likelihood: Possible – Misconfigured S3 buckets are one of the most common cloud misconfigurations, and attackers routinely scan them. Impact: High – If exposed data is sensitive or regulated, the fallout can be severe
<i>Recommended Mitigation</i>	Type of Mitigation: The recommended risk mitigation type is Avoidance (secondary: Reduction) Why Risk Avoidance? This solution is suggested because instead of having to check every individual bucket's permissions, enabling account-level Public Access Block applies the restriction globally
<i>NIST 800-171 Family and Control</i>	Access Control (AC) → 3.1.1, 3.1.2, 3.1.3, 3.1.14, 3.1.20 Audit and Accountability → 3.3.8 Configuration Management → 3.4.6 System and Communication Protection → 3.13.5
<i>Cross Framework Mapping</i>	NIST 800-53 - Mapped? Yes ISO 27001:2013 - Mapped? Yes SOC2 - Mapped? No

FAIL	high	s3	us-east-1	s3_account_level_public	Check S3 Account Level		Block Public Access is not configured for the account 822621041052.	Public access policies may be applied to sensitive data buckets.	You can enable Public Access B read more...	•AWS-Audit-Manager-Cont... •CISA: your-sys... 3, your-data-2 Cyber-Security Framework: annex_i_1_3 •A... Account-Secur... Onboarding: S... Block Public Ac... •CIS-3.0: 2.1.4 •HIPAA: 164_308_a_1_ii, 164_308_a_3_i •FedRamp- Moderate-Rev... 4: ac-3, ac-6, ac-1, ac-21-b, cm-4, sc-7-3, sc-7 Well-Architecte... Framework-Se... Pillar: SEC03-B... •CIS-1.5: 2.1.5 ISMS-P-2023: 2.9.4 •NIST-800-53... Revision-2: 3_1_1, 3_1_2, 3_1_3, 3_1_20, 3_3_8, 3_13_2, 3_13_5 •ISO27001-2022 A.8.1 •CIS-2.0: 11: 11.10-d, 11.11 •NIST-CSF-1.1: ac_5, ds_5, ip_8 •AWS-Foundatio... Security-Best-Practices: S3.1 4.0.1: 2.1.4 •FFI
------	------	----	-----------	-------------------------	------------------------	--	---	--	---	--

Prowler Finding 2

TotalCloud Finding 2:

Possible	Major	Major	Major
Likely	Moderate	Major	Major
Rare	Minor	Moderate	Major
	Low	Medium	High

<i>Finding Name and Title</i>	CID-30 - IAM policy changes are not monitored
<i>Service</i>	CloudTrail
<i>Risk, Severity, and Risk Mitigation Type</i>	If IAM policy changes are not monitored, someone could make unauthorized changes to who can access what and we wouldn't know. This could lead to sensitive data exposure or attackers giving themselves too much access.
	Risk Equation: Likelihood x Impact
	Likelihood: Likely - Unauthorized access attempts via the AWS Management Console are a common vector for intrusion and often go unnoticed if not monitored.
	Impact: High - Since it deals with permissions and identity access
<i>Recommended Mitigation</i>	Type of Mitigation? The recommended mitigation is Reduction.
	Why? Risk Reduction - Set up a CloudWatch alarm to monitor for IAM policy changes. This reduces the chance of someone making a dangerous change without being noticed
<i>CIS AWS Foundation Benchmark</i>	Control Family: 3 - Logging <ul style="list-style-type: none">● 3.1 Ensure Cloud Trail is enabled in all regions● 3.2 - Ensure CloudTrail log file validation is enabled● 3.3 - Ensure the S3 bucket used to store CloudTrail logs is not publicly accessible● 3.4 - Ensure CloudTrail trails are integrated with CloudWatch Logs
	This Control 3.4 one matches best Control ID (CID).

← Control Evaluation: Ensure IAM policy changes are mo...

CID-30 Ensure IAM policy changes are monitored

[View Less](#)

Policy: CIS Amazon Web Services Foundations Benchmark

Platform: AWS

Evaluation: Check CloudWatch alarm exists for cloud trail events for monitoring IAM policy changes This control checks for: Accou...

Service: CloudTrail

Manual Remediation:

[View Steps](#)

Criticality: ■ High

**TotalCloud Finding
2**

← Control Evaluation: Ensure IAM policy changes are mo...

CID-30 Ensure IAM

[View Less](#)

Policy: CIS Amazon

Evaluation: Check Cloud

Manual Remediation: [View Steps](#)

Run Time Remediation Steps

Follow these steps to remediate

Perform the following to setup the metric filter, alarm, SNS topic, and subscription:

Method1: Via CLI

1. Create a metric filter based on filter pattern provided which checks for unauthorized API calls and the <cloudtrail_log_group_name>


```
# aws logs put-metric-filter --log-group-name <cloudtrail_log_group_name> --filter-name <Name_for_metric> --metric-transformations metricName=<Name_for_metric>,metricNamespace='CISBenchmark',metricValue=1 --filter-pattern '$(eventName=DeleteGroupPolicy) || $(eventName=DeleteRolePolicy) || $(eventName=DeleteUserPolicy) || $(eventName=PutGroupPolicy) || $(eventName=PutRolePolicy) || $(eventName=PutUserPolicy) || $(eventName/CreatePolicy) || $(eventName=DeletePolicy) || $(eventName=CreatePolicyVersion) || $(eventName=DeletePolicyVersion) || $(eventName=AttachRolePolicy) || $(eventName=DetachRolePolicy) || $(eventName=AttachUserPolicy) || $(eventName=DetachUserPolicy) || $(eventName=AttachGroupPolicy) || $(eventName=DetachGroupPolicy)'
```

Note: You can choose your own metricName and metricNamespace strings. Using the same metricNamespace for all Foundations Benchmark metrics will group them together.

2. Create an SNS topic that the alarm will notify

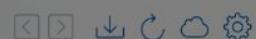

```
# aws sns create-topic --name <sns_topic_name>
```

Note: you can execute this command once and then re-use the same topic for all monitoring alarms.
3. Create an SNS subscription to the topic created in step 2


```
aws sns subscribe --topic-arn <sns_topic_arn> --protocol <protocol_for_sns> --notification-endpoint <sns_subscription_endpoints>
```

Note: you can execute this command once and then re-use the SNS subscription for all monitoring alarms.
4. Create an alarm that is associated with the CloudWatch Logs Metric Filter created in step 1 and an SNS topic created in step 2

Last 24 Hrs



Evidence

[OK](#)

Prowler Finding

Probable	Critical	Critical	Critical	Critical
Possible	Major	Critical	Critical	Critical
Unlikely	Moderate	Major	Critical	Critical
Rare	Minor	Moderate	Major	Critical
	Low	Medium	High	Very High

<i>Finding Name and Title</i>	iam_root.hardware_mfa_enabled Ensure only hardware MFA is enabled for the root account
<i>Service</i>	IAM
<i>Risk, Severity, and Risk Mitigation Type</i>	The AWS root account is highly privileged, and enabling Multi-Factor Authentication (MFA) adds an essential layer of security beyond just a username and password. For Level 2 security, it is recommended to protect the root account specifically with a hardware MFA device. Risk Equation: Likelihood × Impact Likelihood: Possible – Due to the privilege it is most probable to be compromised Impact: Critical – The root account is the most important account to protect
<i>Recommended Mitigation</i>	Type of Mitigation: The recommended risk mitigation type is Reduction Why Risk Avoidance? The recommended solution is using IAM console navigate to Dashboard and expand Activate MFA on your root account
<i>NIST 800-171 Family and Control</i>	Access Control (AC) → 3.1.1, 3.1.2 Identification and Authentication → 3.5.3
<i>Cross Framework Mapping</i>	NIST 800-53 - Mapped? Yes ISO 27001:2013 - Mapped? Yes SOC2 - Mapped? No

All	critical	iam	us-east-1	iam_root_hardware_mfa_enabled	Ensure only hardware MFA is enabled for the root account	[REDACTED]	MFA is not enabled for root account. The root account is the most privileged user in an AWS account. MFA adds an extra layer of protection on top of a user name and password. With MFA enabled when a user signs in to an AWS website they will be prompted for their user name and password as well as for an authentication code from their AWS MFA device. For Level 2 it is recommended that the root account be protected with only a hardware MFA.	Using IAM console navigate to Dashboard and expand Activate MFA on your root account. <input checked="" type="checkbox"/>	•CISA: your-systems-3, your-surroundings-2 •FedRAMP-Low-Revision-4: ac-2, ia-2 •CIS-1.4: 1.6 •KISA-ISMS-P-2023: 2.5.3, 2.5.5 •KISA-ISMS-P-2023-korean: 2.5.3, 2.5.5 •FFIEC: d3-pc-am-b-15, d3-pc-am-b-3, d3-pc-am-b-6 •MITRE-ATTACK: T1078, T1098, T1556, T1550, T1110, T1040 •CIS-2.0: 1.6 •AWS-Well-Architected-Framework-Security-Pillar: SEC01-BP02 •AWS-Foundational-Security-Best-Practices: IAM.6 •NIST-800-53-Revision-4: ia_2_1, ia_2_11 •FedRamp-Moderate-Revision-4: ac-2-1, ac-2-f, ia-2-1-2, ia-2-1 •NIST-800-53-Revision-5: ac_2_1, ac_3_2, ac_3_3, ac_3_3_a, ac_3_3_b_1, ac_3_3_b_2, ac_3_3_b_3, ac_3_3_b_4, ac_3_3_b_5, ac_3_3_c, ac_3_4, ac_3_4_a, ac_3_4_b, ac_3_4_c, ac_3_4_d, ac_3_4_e, ac_3_8, ac_3_12_a, ac_3_13, ac_3_15_a, ac_3_15_b, ac_4_28, ac_7_4, ac_7_4_a, ac_24, cm_5_1_a,
-----	----------	-----	-----------	-------------------------------	--	------------	---	---	---

Prowler Finding 3

TotalCloud Findings 3:

Possible	Major	Major	Major
Likely	Moderate	Major	Major
Rare	Minor	Moderate	Major
	Low	Medium	High

<i>Finding Name and Title</i>	CID-24 - Ensure S3 bucket access logging is enabled on CloudTrail S3 bucket
<i>Resource and Service</i>	sandbox-management-events CloudTrail
<i>Risk, Severity, and Risk Mitigation Type</i>	Server access logging generates a log that contains access records for each request made to your S3 bucket. An access log record contains details about the request, such as the request type, the resources specified in the request worked, and the time and date the request was processed. It is recommended that server access logging be enabled on the CloudTrail S3 bucket. Risk Equation: Likelihood x Impact
	Likelihood: Likely - This can happen often without the proper monitoring Impact: High - This will not log anyone who accesses sensitive data if not enabled allowing for breaches to go unknown
<i>Recommended Mitigation</i>	Type of Mitigation? The recommended mitigation is Reduction. Why? Risk Reduction - To enable server access logging for an Amazon S3 bucket, sign in to the AWS Management Console and navigate to the S3 console. Select the bucket you wish to monitor, go to its Properties , and in the Server access logging section, click Edit . Enable logging and specify a target bucket—ideally a different one in the same AWS Region without a default retention policy—to store the logs. Optionally, set a prefix to organize log files. Save your changes; logs will start appearing in the target bucket within a few hours..
<i>CIS AWS Foundation Benchmark</i>	Control Family: 3 - Logging <ul style="list-style-type: none">3.4 Ensure that server access logging is enabled on the CloudTrail S3 bucket (Manual)Profile Applicability: Level 1Rationale: By enabling server access logging on target S3 buckets, it is possible to capture all events that may affect objects within any target bucket

Prowler Finding 4

Probable	Critical	Critical	Critical	Critical
Possible	Major	Critical	Critical	Critical
Unlikely	Moderate	Major	Critical	Critical
Rare	Minor	Moderate	Major	Critical
	Low	Medium	High	Very High

<i>Finding Name and Title</i>	iam_aws_attached_policy_no_administration_privileges Ensure IAM AWS-Managed policies that allow full “.” administrative privileges are attached
<i>Service</i>	IAM
<i>Risk, Severity, and Risk Mitigation Type</i>	The root account is the most privileged user in an AWS account. MFA adds an extra layer of protection on top of a user name and password. With MFA enabled when a user signs in to an AWS website they will be prompted for their user name and password as well as for an authentication code from their AWS MFA device. For Level 2 it is recommended that the root account be protected with only a hardware MFA Risk Equation: Likelihood × Impact Likelihood: Probable – Due to the privilege it is most probable to be compromised Impact: Critical – The root account is the most important account to protect
<i>Recommended Mitigation</i>	Type of Mitigation: The recommended risk mitigation type is Reduction Why Risk Reduction? The recommended solution is using IAM console navigate to Dashboard and expand Activate MFA on your root account
<i>NIST 800-171 Family and Control</i>	Access Control (AC) → 3.1.1, 3.1.2 Configuration Management → 3.4.6 System and Communication Protection → 3.13.3
<i>Cross Framework Mapping</i>	NIST 800-53 - Mapped? Yes ISO 27001:2013 - Mapped? Yes SOC2 - Mapped? No

					Block.			
FAIL	high	iam	us-east-1	iam_aws_attached_AWS-policy_no	Ensure IAM AWS- Managed administrative policies that privileges allow full "*" administrative privileges are not attached	AWS policy AdministratorAccess [REDACTED]	IAM policies are attached and allows '*' which privileges are granted to users, groups, or roles. It is recommended and considered a standard security advice to grant least privilege—that is, granting only the permissions required to perform a task. Determine what users need to do and then craft policies for them that let the users perform only those tasks instead of allowing full administrative privileges. Providing full administrative privileges instead of restricting to the minimum set of	<p>It is more secure to start with a minimum set of permissions and grant additional permissions as necessary, rather than starting with permissions that are too lenient and then trying to tighten them later. List policies and analyze if permissions are the least possible to conduct business activities.</p> <p><input checked="" type="checkbox"/></p> <ul style="list-style-type: none"> •CISA: your-systems-3, your-surroundings-3 •RBI-Cyber-Security-Framework: annex_i_7_1 •CIS-3.0: 1.16 •HIPAA: 164_308_a_1_ii_b, 164_308_a_3_i, 164_308_a_3_ii_b, 164_308_a_4_i, 164_308_a_4_ii_b, 164_308_a_4_ii_c, 164_312_a_1 •FedRamp-Moderate-Revision-4: ac-2-1, ac-2-f, ac-2-j, ac-3, ac-5-c, ac-6-10, ac-6, sc-2 •AWS-Well-Architected-Framework-Security-Pillar: SEC03-BP02 •CIS-1.5: 1.16 •SOC2: cc_1_3, cc_6_3 •GDPR: article_25 •KISA-ISMS-P-2023: 2.5.1 •NIST-800-171-Revision-2: 3_1_1, 3_1_2, 3_1_4, 3_1_5, 3_1_6, 3_1_7, 3_4_6, 3_13_3 •ISO27001-

Prowler Finding 4

TotalCloud Findings 4:

Possible	Major	Major	Major
Likely	Moderate	Major	Major
Rare	Minor	Moderate	Major
	Low	Medium	High

<i>Finding Name and Title</i>	CID-27 - Ensure unauthorized API calls are monitored
<i>Service</i>	CloudTrail
<i>Risk, Severity, and Risk Mitigation Type</i>	Real-time monitoring of API calls can be achieved by directing CloudTrail Logs to CloudWatch Logs or an external Security Information and Event Management (SIEM) environment, and establishing corresponding metric filters and alarms.
	Risk Equation: Likelihood x Impact
	Likelihood: Possible - This can happen often without the proper monitoring
	Impact: High - This alert may be triggered by normal read-only console activities that attempt to opportunistically gather optional information but gracefully fail if they lack the necessary permissions.
<i>Recommended Mitigation</i>	Type of Mitigation? The recommended mitigation is Reduction. Why? Risk Reduction - To monitor unauthorized API calls in AWS, you can set up a CloudWatch alarm using the AWS Command Line Interface (CLI). This process involves creating a metric filter to detect specific error codes in CloudTrail logs, establishing an Amazon Simple Notification Service (SNS) topic for notifications, subscribing endpoints to this topic, and configuring a CloudWatch alarm to trigger notifications upon detecting unauthorized activities. It is recommended that a metric filter and alarm be established for unauthorized API calls.
<i>CIS AWS Foundation Benchmark</i>	Control Family: 4 - Monitoring <ul style="list-style-type: none"> 4.1 Ensure unauthorized API calls are monitored (Manual) Profile Applicability: Level 2 Rationale: CloudWatch is an AWS native service that allows you to observe and monitor resources and applications. CloudTrail logs can also be sent to an external Security Information and Event Management (SIEM) environment for monitoring and alerting

← Control Evaluation: Ensure unauthorized API calls are mo...

CID-27 Ensure unauthorized API calls are monitored

Policy: NIST Special Publication 800-171 rev 2 [\[1 more\]](#)

Evaluation: Check CloudWatch alarm exists for cloud trail events to monitor unauthorized API calls This control checks for: Account has at least one Active Multi-region Cloud trail that captures all management ev...

Manual Remediation: [View Steps](#)

Platform:

AWS

Service:

CloudTrail

Criticality:

■ High

[View Less](#) ▲

TotalCloud Finding 4

← Control Evaluation: Ensure unauthorized API calls are mo...

CID-27 Ensure unauthorized API calls are monitored

Policy: NIST Special Publication 800-171 rev 2 [\[1 more\]](#)

Evaluation: Check CloudWatch alarm exists for cloud trail events to monitor unauthorized API calls This control checks for: Account has at least one Active Multi-region Cloud trail that captures all management ev...

Manual Remediation: [View Steps](#)

Platform:

AWS

Service:

CloudTrail

■ High

[View Less](#) ▲

policy.name:"CIS Amazon Web Services Foundations Benchmark" and control.result:"FAI

Actions (0) ▼

RESOURCE

822621041052

Run Time Remediation Steps

Follow these steps to remediate

Perform the following to setup the metric filter, alarm, SNS topic, and subscription:

Method1: Via CLI

1. Create a metric filter based on filter pattern provided which checks for unauthorized API calls and the <cloudtrail_log_group_name>


```
# aws logs put-metric-filter --log-group-name <cloudtrail_log_group_name> --filter-name <Name_for_metric> --metric-transformations metricName=<Name_for_metric>,metricNamespace='CISBenchmark',metricValue=1 --filter-pattern '{ ($.errorCode = "UnauthenticatedOperation") || ($.errorCode = "AccessDenied") }'
```

Note: You can choose your own metricName and metricNamespace strings. Using the same metricNamespace for all Foundations Benchmark metrics will group them together.
2. Create an SNS topic that the alarm will notify


```
# aws sns create-topic --name <sns_topic_name>
```

Note: you can execute this command once and then re-use the same topic for all monitoring alarms.
3. Create an SNS subscription to the topic created in step 2


```
# aws sns subscribe --topic-arn <sns_topic_arn> --protocol <protocol_for_sns> --notification-endpoint <sns_subscription_endpoints>
```

Note: you can execute this command once and then re-use the SNS subscription for all monitoring alarms.
4. Create an alarm that is associated with the CloudWatch Logs Metric Filter created in step 1 and an SNS topic created in step 2


```
# aws cloudwatch put-metric-alarm --alarm-name <Name_for_alarm> --metric-name <Name_for_metric> --statistic Sum --period 300 --threshold 1 --comparison-operator GreaterThanOrEqualToThreshold --evaluation-periods 1 --namespace 'CISBenchmark' --alarm-actions <sns_topic_arn>
```

Note: set the period and threshold to values that fit your organization.

OK

X ↑ ? Last 24 Hrs ▼

1 - 1 of 1 ◀ ▶ Download Edit CloudWatch Metrics CloudWatch Logs

RESULT

FAIL

Evidence



Mitigation Plan/ Schedule

- Organize findings by severity (Critical → High → Medium → Low).
- Build a simple plan: Critical findings = immediate remediation (0-2 weeks), High = 2-4 weeks, Medium = 1-2 months, Low = 2-3 months.

Challenges Faced

1. Prowler has the exact mappings in the compliance column, but with Qualys TotalCloud, we had to search for each control family and the specific mapping.
2. Prowler takes more time to set up; you have to install it in CloudShell each time.
3. With TotalCloud, we switched from AWS Best Practices to CIS Amazon Web Service Foundation Benchmark policy to find all our failed findings.

Thank you

Any Further Questions?