

Cloud Security Assessment

Prepared for: Sample Inc

COMPLETE REPORT

May 18, 2026



This engagement was performed in accordance with the Statement of Work, and the procedures were limited to those described in that agreement. The findings and recommendations in this report reflect the systems, scope, and timeframe defined in the Statement of Work.

While every effort was made to provide thorough coverage, no security assessment can identify every possible issue, and results should not be interpreted as a guarantee of complete security.

Contact Information

+1 (888) 337-0467

hello@redsentry.com

TABLE OF CONTENTS

Summary	4
Engagement Overview	4
Goals	4
Dates	4
Findings Overview	5
Scope	4
Executive Summary	6
Tests Performed	7
Cloud Security Assessment Findings	10
Privilege Escalation [T8KEQEF-F0001]	10
Ensure a log metric filter and alarm exist for unauthorized API calls [T8KEQEF-F0002]	13
Ensure a log metric filter and alarm exist for AWS Config configuration changes [T8KEQEF-F0003]	15
Ensure credentials unused for 45 days or greater are disabled [T8KEQEF-F0004]	17
Ensure EBS Volume Encryption is Enabled in all Regions [T8KEQEF-F0005]	22
Ensure IAM password policy prevents password reuse. [T8KEQEF-F0006]	25
Appendix (A): Severity Description	28

SUMMARY

Engagement Overview

Sample Company engaged Red Sentry to assess the security of its cloud environment. The following report details the findings identified during the engagement.

Goals

Identify vulnerabilities in the company’s cloud environment based on the CIS benchmarks.

Dates

April 08, 2026
KICKOFF

April 08, 2026 - April 17, 2026
TESTING PERIOD

May 18, 2026
DELIVERY

Scope

Amazon Web Services environment

Findings Overview

During the assessment, the testing team identified multiple security findings across the environment. A finding represents a unique security issue with a distinct root cause. Instances represent the number of instances where that issue appears within the assessed environment. The following table summarizes the identified findings and the number of Instances observed during testing.

Severity	Finding ID	Instances
CRITICAL	T8KEQEF-F0001	1
	T8KEQEF-F0002	2
HIGH	T8KEQEF-F0003	2
	T8KEQEF-F0004	16
LOW	T8KEQEF-F0005	5
	T8KEQEF-F0006	3

Overall Grade



GRADE



EXECUTIVE SUMMARY

The assessment team conducted a penetration test on Sample Company's Amazon Web Services (AWS) cloud environment. Overall, the environment security configurations need improvement as 19 issues were detected.

Regarding the severity of the findings, 1 instance of Critical Vulnerability was found. This is related to Privilege Escalation due to AWS policies.

Regarding the severity of the findings, 2 instances of high vulnerabilities were found. These consist of not having configured log metric filters and alarms for unauthorized API calls and IAM policy changes.

Following the order of severity, 3 types of low vulnerabilities were identified. These belong, briefly speaking, to the encryption of Elastic Block Store (EBS), unused credentials for passwords and access keys, and lastly, an IAM password policy that prevents the reuse of passwords.

In summary, to improve the organization's security posture, the assessment team recommends restricting and reviewing AWS actions related to attached policies, policy versions, etc. Enabling the encryption of EBSs, disabling or removing unused passwords and access keys, and improving IAM policies related to the reuse of passwords.

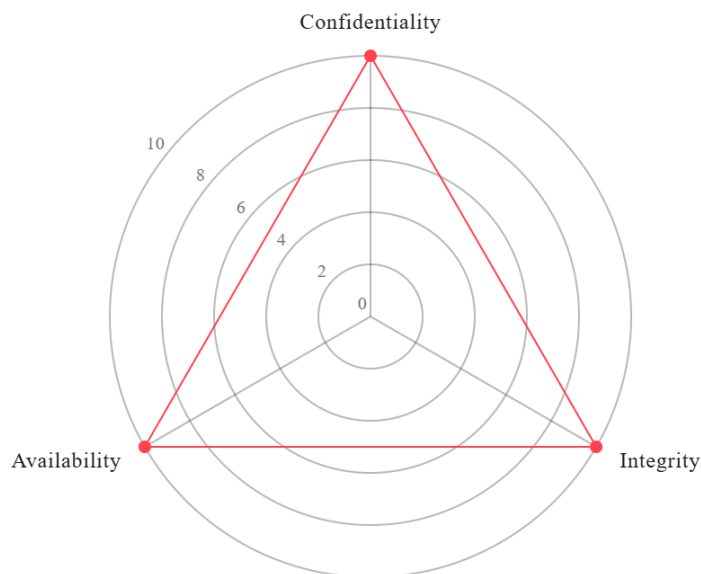


Figure 1: Security risk chart

As the image above shows, the overall risk value is high as the average score is 10.

As a general rule, an area below 4 shows low business impact, between 4 and 7 medium impact, and above 7 high impact.

Tests Performed

In a cloud assessment, it is important to test the security from both an unauthenticated and an authenticated perspective in order to have a better understanding of the security posture. This includes the following components:

Authenticated Asset Enumeration

from an authenticated perspective the assessment team enumerates all of the environment assets to have a better understanding of the attack surface.

Credential Leak Search

using internal sources, the team searches leaked information about the company based on the company's domain, high-level employees and address.

External Perimeter Analysis

from an unauthenticated perspective the assessment team enumerates all of the publicly facing assets/services related to the Cloud environment. This typically includes public storage objects, remote databases, exposed VMs, and web applications.

Misconfiguration Assessment

from an authenticated perspective and depending on the cloud provider, the assessment team verifies that best practices are followed, based on CIS Benchmarks guidelines, this is from a manual perspective, at the same time automated tools are also used for this checking process.

OSINT research:

in order to be more effective in the exploitation phase, the assessment teams conduct research on the company and extract data about its location, current employees, business logic, and most important assets.

Password Spraying

using the collected data, the assessment team enumerates valid users and tests them against the in-scope Cloud providers by using a small set of password candidates.

Privilege Escalation

the team attempts to exploit identified weaknesses to gain higher privileges. This involves testing for common privilege escalation paths, such as overly permissive roles or vulnerable services, to assess the potential impact of compromised accounts.

Sensitive Data Exposure

a weak access control policy can lead to the disclosure of sensitive data, which could then be weaponized to harm the security posture of the in-scope cloud assets. This includes improper configurations, insufficient encryption, and inadequate identity and access management (IAM) practices.

User Enumeration

based on the previous component, the team elaborates a list with different user combinations based on the company's domains and email formulas.

Vulnerability Scanning:

Targets cloud environments to uncover misconfigurations, exposed services, and potential SSRF vulnerabilities that could lead to unauthorized access to metadata and other sensitive information.

CLOUD SECURITY ASSESSMENT FINDINGS

T8KEQEF-F0001 – Privilege Escalation

CRITICAL

Identifier	T8KEQEF-F0001
Severity	Critical
Instances	1

Description

Privilege escalation in AWS environments refers to the process by which an attacker or a malicious user gains elevated access to resources that they are not normally permitted to access. This can occur through various means, such as exploiting vulnerabilities in applications, misconfigurations, or using compromised credentials to gain higher-level permissions. Once an attacker escalates their privileges, they can potentially gain control over critical AWS resources, modify or delete data, disrupt services, and further infiltrate the network. To mitigate these risks, it is essential to implement strict access controls, regularly audit and monitor permissions, enforce the principle of least privilege, and apply security best practices such as multi-factor authentication and automated alerts for unusual activities.

Instances (1)

- Clients AWS Environment

Evidence

During the penetration test, the security team employed a tool named “Pacu” to perform many scans across the AWS infrastructure. To do this, the security team had to supply their access key and secret access key in order to perform testing from the command line. To kick off the testing with Pacu, the security team immediately went after a possible Privilege Escalation.

```
[iam_privesc_scan] CONFIRMED: CreateNewPolicyVersion
[iam_privesc_scan] CONFIRMED: SetExistingDefaultPolicyVersion
[iam_privesc_scan] CONFIRMED: CreateEC2WithExistingIP
[iam_privesc_scan] CONFIRMED: AttachUserPolicy
[iam_privesc_scan] CONFIRMED: AttachGroupPolicy
[iam_privesc_scan] CONFIRMED: PutUserPolicy
[iam_privesc_scan] CONFIRMED: PutGroupPolicy
[iam_privesc_scan] CONFIRMED: AddUserToGroup
[iam_privesc_scan] CONFIRMED: PassExistingRoleToNewLambdaThenInvoke
[iam_privesc_scan] CONFIRMED: PassExistingRoleToNewCloudFormation
[iam_privesc_scan] CONFIRMED: PassExistingRoleToNewCodeStarProject
```

Figure 2: Exploitable Vectors for Privilege Escalation

To fully test out, the security team created a “low-level” user with its own access and secret access keys. Additionally, a new IAM policy named “privesc” was created and attached to the lower-level user. After deploying Pacu with the newly acquired keys and IAM policy, the team ran “run iam__privesc_scan”

```
[iam_privesc_scan] Is there a specific user you want to target? They must not already have two sets of access keys. Enter their user name now or just hit enter to enumerate users and view a list of options:
[iam_privesc_scan] Data (IAM > Users) not found, run module "iam_enum_users_roles_policies_groups" to fetch it?
[iam_privesc_scan] Running module iam_enum_users_roles_policies_groups...
[iam_enum_users_roles_policies_groups] No Users Found
[iam_enum_users_roles_policies_groups] FAILURE: MISSING NEEDED PERMISSIONS
[iam_enum_users_roles_policies_groups] iam_enum_users_roles_policies_groups completed.

[iam_enum_users_roles_policies_groups] MODULE SUMMARY:
  0 Users Enumerated
  IAM resources saved in Pacu database.

[iam_privesc_scan] Found 0 user(s). Choose a user below.
[iam_privesc_scan] [0] Other (Manually enter user name)
[iam_privesc_scan] Choose an option: 0
[iam_privesc_scan] Enter a user name:
[iam_privesc_scan] Running module iam_backdoor_users_keys...
[iam_backdoor_users_keys] Backdoor the following users?
[iam_backdoor_users_keys] FAILURE: MISSING REQUIRED AWS PERMISSIONS
[iam_backdoor_users_keys] iam_backdoor_users_keys completed.

[iam_backdoor_users_keys] MODULE SUMMARY:
  0 user key(s) successfully backdoored.

[iam_privesc_scan] iam_privesc_scan completed.
[iam_privesc_scan] MODULE SUMMARY:
  Privilege escalation was successful
```

Figure 3: Successful Privilege Escalation

After deploying a few enumeration modules, Pacu returns with a summary saying that the exploit was successful. To verify, it was advised to refresh the policies page on the user “low-level” to see if a policy was added “inline.”

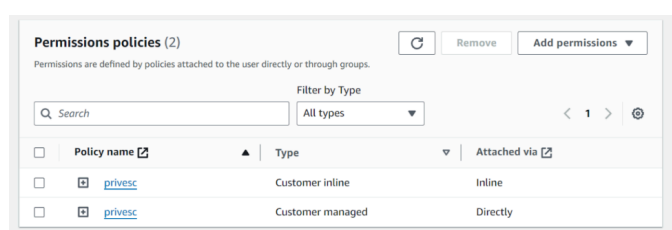


Figure 4: Inline Policy

The policy confirmed that the privilege escalation attempt worked, and the lower-level user had their privileges escalated to the administrator level, giving them free rein over the AWS infrastructure as a whole.

Steps for Recreation

1. Download Pacu and add the credentials of the environment
2. Run the modules to enumerate services
3. Run the following module:
 - run iam__privesc_scan
4. Check the policies of the users to check if the privilege escalation worked

Recommendations

Restrict Sensitive IAM Actions:

- Actions like “iam:CreatePolicyVersion,” “iam: SetDefaultPolicyVersion”, “iam:PassRole”, “iam: AttachUserPolicy”, etc., should be carefully controlled.

Automate Key Rotation:

Use AWS IAM Access Analyzer to identify unused keys and automate their rotation.

Resources

Avoiding PrivEsc in AWS

<https://stackoverflow.com/questions/70580724/how-to-avoid-privilege-escalation-in-aws>

T8KEQEF-F0002 – Ensure a log metric filter and alarm exist for unauthorized API calls

HIGH

Identifier	T8KEQEF-F0002
Severity	High
Instances	2

Description

Real-time monitoring of API calls can be achieved by directing CloudTrail Logs to CloudWatch Logs and establishing corresponding metric filters and alarms. It is recommended that a metric filter and alarm be established for unauthorized API calls.

Instances (2)

- data
- prod

Evidence

The assessment team discovered that some roles had misconfigured the log metric filter and alarm.

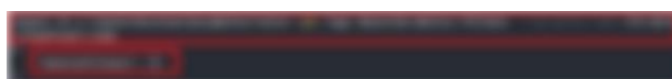


Figure 5: Actual configuration of log metric filter and alarm for the “data” role

The image above shows that the data role doesn't have any log metric filter or alarm configured.

Alerts can be triggered by normal read-only console activities that attempt to opportunistically gather optional information; not having them configured would not generate any alert in the described case.

Steps for Recreation

1. Run the following command:

- `aws logs describe-metric-filters --log-group-name <value>`

Check the result

Recommendations

To mitigate the risk of this issue, the assessment team recommends the following steps:

- Create a metric filter based on the filter pattern provided, which checks for unauthorized API calls and the `<cloudtrail_log_group_name>` taken from audit step 1.
- Create an SNS topic that the alarm will notify.
- Create an SNS subscription to the topic created in step 2.
- Create an alarm that is associated with the CloudWatch Logs Metric
- Filter created in step 1 and an SNS topic created in step 2.

Resources

About Metric filters and alarms

<https://docs.aws.amazon.com/awscloudtrail/latest/userguide/cloudwatch-alarms-for-cloudtrail.html>

T8KEQEF-F0003 – Ensure a log metric filter and alarm exist for AWS Config configuration changes

HIGH

Identifier	T8KEQEF-F0003
Severity	High
Instances	2

Description

Real-time monitoring of API calls can be achieved by directing CloudTrail Logs to CloudWatch Logs and establishing corresponding metric filters and alarms. It is recommended that a metric filter and alarm be established for changes made to Identity and Access Management (IAM) policies.

Instances (2)

- data
- prod

Evidence

The assessment team discovered that some roles had misconfigured the log metric filter and alarm.

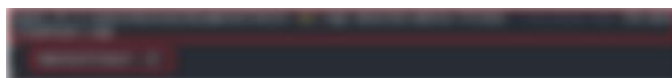


Figure 6: Log metric filter and alarm IAM policy changes for the “data” role

The image from above shows that the role doesn't have any log metric filter or alarm configured, by not having IAM policy changes monitored, it cannot be ensured that authentication and authorization controls remain intact.

Steps for Recreation

1. Run the following command:

- `aws logs describe-metric-filters --log-group-name <value>`

Check the result

Recommendations

To mitigate the risk of this issue, the assessment team recommends the following steps:

- Create a metric filter based on the filter pattern provided, which checks for IAM policy changes and the `<cloudtrail_log_group_name>` taken from audit step 1.
- Create an SNS topic that the alarm will notify
- Create an SNS subscription to the topic created in step 2
- Create an alarm that is associated with the CloudWatch Logs Metric
- Filter created in step 1 and an SNS topic created in step 2

Resources

About Metric filters and alarms

<https://docs.aws.amazon.com/awscloudtrail/latest/userguide/cloudwatch-alarms-for-cloudtrail.html>

T8KEQEF-F0004 – Ensure credentials unused for 45 days or greater are disabled

LOW

Identifier	T8KEQEF-F0004
Severity	Low
Instances	16

Description

AWS IAM users can access AWS resources using different types of credentials, such as passwords or access keys. It is recommended that all credentials that have been unused in 45 or greater days be deactivated or removed.

Instances (16)

- (dev) administrator
- (dev) USR-DataLoadUser
- (dev) USR-API-Deployment
- (staging) administrator
- (staging) USR-DataLoadUser
- (staging) USR-Application-Deployment
- (demo) administrator
- (demo) USR-DEMO-Logs
- (demo) USR-DataLoadUser
- (prod) USR-AnalyticsEmailServiceAccount
- (prod) USR-DataLoadUser
- (prod) USR-API-Deployment
- (dev-data) administrator
- (dev-data) USR-Application-Deployment-User
- (dev-data) USR-DataLoadUser
- (data) USR-Application-Deployment

Evidence

While the assessment team was enumerating the existing credentials, it was discovered that some users of the different roles have credentials that are unused for more than 45 days.



Figure 7: Actual configuration of unused credentials for the “data” role

The image from above shows that there is an instance of unused credentials for the user “USR-Application-Deployment” specifically for access key 1 that was last used on February 2nd, 2021.

The storage of unused credentials increases the opportunity for an attacker to use those abandoned or compromised credentials to gain access to that user and his functionalities.

Note: To avoid duplicating the current issue locations, the findings are displayed by role but only counted once per artifact name.

Steps for Recreation

1. Login to the AWS Management Console
2. Click Services
3. Click IAM
4. Click on Users
5. Click the Settings (gear) icon.
6. Select Console last sign-in, Access key last used, and Access Key Id
7. Click on Close
8. Check and ensure that Console last sign-in is less than 45 days ago.
Note - Never means the user has never logged in.
9. Check and ensure that Access key age is less than 45 days and that Access key last used does not say None.
If the user hasn't signed into the Console in the last 45 days or Access keys are over 45 days old refer to the remediation.

Recommendations

To mitigate the risk of this issue, the assessment team recommends the following steps:

Unused Password

- Login to the AWS Management Console:
- Click Services
- Click IAM
- Click on Users
- Click on Security Credentials
- Select user whose Console last sign-in is greater than 45 days
- Click Security credentials
- In section Sign-in credentials, Console password click Manage
- Under Console Access, select Disable
- Click Apply

Access Keys

- Login to the AWS Management Console:
- Click Services
- Click IAM
- Click on Users
- Click on Security Credentials
- Select any access keys that are over 45 days old and that have been used and click on Make Inactive
- Select any access keys that are over 45 days old and that have not been used and click the X to Delete

Resources

About unused credentials

<https://docs.aws.amazon.com/IAM/latest/UserGuide/best-practices.html#remove-credentials>

Find unused credentials

https://docs.aws.amazon.com/IAM/latest/UserGuide/id_credentials_finding-unused.html

T8KEQEF-F0005 – Ensure EBS Volume Encryption is Enabled in all Regions

LOW

Identifier	T8KEQEF-F0005
Severity	Low
Instances	5

Description

Elastic Compute Cloud (EC2) supports encryption at rest when using the Elastic Block Store (EBS) service. While disabled by default, forcing encryption at EBS volume creation is encouraged.

Instances (5)

- demo - us-east-1
- dev- us-east-1
- dev-data - us-east-1
- prod - us-east-1
- staging - us-east-1

Evidence

The assessment team discovered that some roles have a misconfigured encryption related to the Elastic Block Store (EBS), for the us-east-1 region.

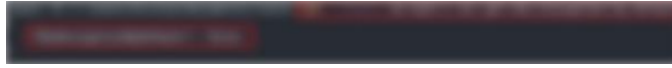


Figure 8: Actual configuration of EBS Encryption for the “demo” role

The image from above shows that for the demo role and the us-east-1 region, The variable called “EbsEncryptionByDefault” is false, and according to the CIS benchmark document, it should be true since this would indicate that there is encryption for the EBS.

Losing access or removing the Key Management Service (KMS) key in use by the EBS volumes will result in no longer being able to access the volumes, for that reason it is important to encrypt the data at rest because this reduces the impact of data that is unintentionally exposed and can nullify the impact of disclosure.

Steps for Recreation

1. Login to the AWS Management Console and open the Amazon EC2 console using <https://console.aws.amazon.com/ec2/>.
2. Under Settings, click EBS encryption.
3. Verify Always encrypt new EBS volumes displays Enabled.
4. Repeat for each region in use.

Recommendations

To mitigate the risk of this issue, the assessment team recommends the following steps:

- Login to AWS Management Console and open the Amazon EC2 console using <https://console.aws.amazon.com/ec2/>
- Under Account attributes, click EBS encryption.
- Click Manage.
- Click the Enable checkbox.
- Click Update EBS encryption
- Repeat for every region requiring the change

Resources

About EBS encryption

<https://aws.amazon.com/blogs/aws/new-opt-in-to-default-encryption-for-new-ebs-volumes/>

EBS Encryption User Guide

<https://docs.aws.amazon.com/ebs/latest/userguide/ebs-encryption.html>

T8KEQEF-F0006 – Ensure IAM password policy prevents password reuse.

LOW

Identifier	T8KEQEF-F0006
Severity	Low
Instances	3

Description

IAM password policies can prevent the reuse of a given password by the same user. It is recommended that the password policy prevent the reuse of passwords.

Instances (3)

- dev
- staging
- demo

Evidence

The assessment team discovered that some roles have a misconfigured password policy, specifically to prevent the reuse of a password.

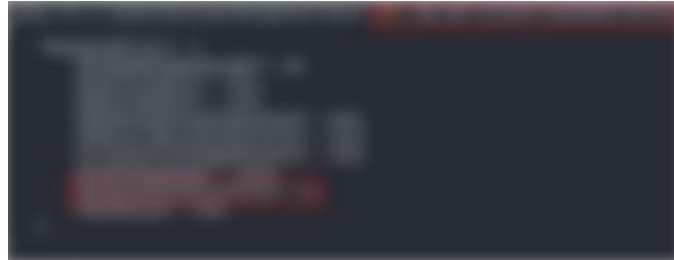


Figure 9: Actual configuration of Password Reuse Prevention

At the time of the test, it was discovered that the password reuse attempts are not what is recommended according to the CIS benchmarks since it should be 24, attackers can exploit this by performing brute-force login attacks.

Steps for Recreation

1. Login to AWS Console (with appropriate permissions to View Identity Access Management Account Settings)
2. Go to IAM Service on the AWS Console
3. Click on Account Settings on the Left Pane
4. Ensure "Prevent password reuse" is checked
5. Ensure "Number of passwords to remember" is set to 24

Recommendations

To mitigate the risk of this issue, the assessment team recommends the following steps:

- Login to AWS Console (with appropriate permissions to View Identity Access Management Account Settings)
- Go to IAM Service on the AWS Console
- Click on Account Settings on the Left Pane
- Check "Prevent password reuse"
- Set "Number of passwords to remember" is set to 24

Resources

About Password policies

https://docs.aws.amazon.com/IAM/latest/UserGuide/id_credentials_passwords_account-policy.html#password-policy-rules

APPENDIX (A): SEVERITY DESCRIPTION

The assessment team used the following criteria to rate the findings in this report. Red Sentry derived these risk ratings from the industry and organizations such as OWASP.

The severity of each finding in this report is independent. Finding severity ratings combine direct technical and business impact with the worst-case scenario in an attack chain. The more significant the impact, and the fewer vulnerabilities that must be exploited to achieve that impact, the higher the severity.

Critical

CRITICAL

Vulnerability is an otherwise high-severity issue with additional security implications that could lead to exceptional business impact. Examples: trivial exploit difficulty, business-critical data compromised, bypass of security controls, direct violation of communicated security objectives, and large-scale vulnerability exposure.

High

HIGH

Vulnerability may result in direct exposure including, but not limited to: the loss of application control, execution of malicious code, or compromise of underlying host systems. The issue may also create a breach in the confidentiality or integrity of sensitive business data, customer information, and administrative and user accounts. In some instances, this exposure may extend farther in the infrastructure beyond the data and systems associated with the application.

Medium

MEDIUM

Vulnerability does not lead directly to the exposure of critical application functionality, sensitive business and customer data, or application credentials. However, it can be executed multiple times or leveraged in conjunction with another issue to cause direct exposure. Examples include brute-forcing and client-side input validation.

Low

LOW



Vulnerability may result in limited exposure of application control, sensitive business and customer data, or system information. This type of issue provides value only when combined with one or more issues of a higher risk classification. Examples include overly detailed error messages, the disclosure of system versioning information, and minor reliability issues.

Informational

INFO

Finding does not have a direct security impact but represents an opportunity for additional layers of security, is considered a best practice, or has the possibility of turning into an issue over time. Finding is a security-relevant observation that has no direct business impact or exploitability, but may lead to exploitable vulnerabilities. Examples include poor communication between organizations, documentation encouraging poor security practices, or lack of security training.