CryptoGen Nepal

Vulnerability Assessment & Penetration Testing (VAPT) Report

[ORG NAME]

Document Control

Document Name	Final Report of Vulnerability Assessment and Penetration Testing of [ORG NAME]
Abstract	This document details the approaches and vulnerabilities identified in the existing assets of the "[ORG NAME]" from security perspective.
Security Classification	Confidential
Location	[ORG NAME], [ADDRESS]

Authorization		
Document Owner	**Reviewed by**	**Authorized by**
Cryptogen Nepal	[Reviewer's Name]	[Authorizer's Name]

Amendment Log				
Version	**Modification Date**	**Section**	**Amendment/ Modification/ Deletion**	**Brief description of the change**
1.1	[Date]	-	-	VAPT Final Report

Distribution list	
Name	**Designation**
[Name]	[Position], Cryptogen Nepal Pvt. Ltd.

Date: **[Date]**

To,
[Receiver's Name],
[ORG NAME]
[ADDRESS],

Dear Sir,

We hereby submit to you the final draft report of Vulnerability Assessment and Penetration Testing of ABC Company. The security assessment was carried out from [Start Date] to [End Date]. We specialize in manual assessments that go beyond basic automated tests to identify real attack vectors that can be used against your AWS environment. The report includes an executive summary, vulnerability summary and findings with technical details. We believe that the evidence obtained from our analysis provides a reasonable basis for our conclusions and findings regarding the VAPT objectives and scope. The findings included in the report should be treated as samples. Similar vulnerabilities can exist if similar modules have been used in any other services which should be verified and fixed by the relevant development team.

We would like to express our appreciation to [ORG NAME] for being courteous, helpful, and professional without which the completion of the Security Assessment would be difficult.

Regards,
[Sender's Name]
[Position]
CryptoGen Nepal Pvt. Ltd.
[Email]
[Number]

Table of Contents	Page

Executive Summary

We were engaged by [ORG NAME] Limited to perform Vulnerability Assessment and Penetration Testing on Company's AWS Environment. As a result, we found different vulnerabilities in the AWS Infrastructure of ABC Company.

The purpose of this vulnerability assessment and penetration testing was to identify the security vulnerabilities in the AWS Environment and suggest the best recommendation for it.

The purpose of this vulnerability assessment and penetration testing was to identify the security vulnerabilities in the AWS infrastructure and suggest the best recommendation for it. We found one Critical, two High, one medium, two low and two informational vulnerabilities in the AWS environment.

The criticality of those issues was identified using the CVSS v3.1 based scoring system (If applicable). The impacts and recommendations of those vulnerabilities are described thoroughly below in the finding sections.

Objective

We have defined the following objectives for vulnerability assessment and penetration test:

- Evaluate the current security posture of the organization by evaluating the implemented controls.
- Prevent by identifying and addressing risks before security breaches occur.
- Determine the root cause of the vulnerability.
- Evaluate the implemented security on the client's AWS Infrastructure.
- Demonstrate vulnerabilities and perform Penetration tests to clarify the impact that it carries.
- Avoid incidents that put the organization’s goodwill reputation at stake.

VAPT Methodology

Information Gathering

Here, the goal of a penetration tester is to learn as much as possible about their target. They'll collect data about end users, systems, and apps, among other things. The data will be used to be more specific in the penetration test, with a complete and detailed list of systems to understand what must be handled and reviewed. Search engine inquiries, domain name searches, internet foot printing, social engineering, and even checking up tax records to find personal information are some of the strategies utilized during this phase.

Threat Modeling

Threat modeling identifies the different forms of threat agents that can affect a computer application or system. It puts itself in the shoes of a

malicious hacker to see how much damage it may cause. When done correctly, threat modeling can help justify security efforts by providing a clear line of sight throughout a software project. The goal of threat modeling is to increase security awareness throughout the entire team. It's the first step toward everyone sharing responsibility for security.

Vulnerability Analysis

The goal of this stage is to determine the source and root cause of the vulnerabilities discovered in the previous step. It entails determining which system components are responsible for each vulnerability, as well as the vulnerability's fundamental cause. An obsolete version of an open-source library, for example, could be the source of a vulnerability. This creates a clear path for remediation – library upgrades. When conducting any type of vulnerability analysis, the tester should correctly scope the testing for appropriate depth and breadth to fulfill the desired outcome's goals or requirements.

Exploitation

The pentester starts by testing the exploits located within your network, apps, and data, using a map of all available vulnerabilities and entry points. The ethical hacker's purpose is to see how far they can go into your system, find high-value targets, and avoid being detected. Exploits are created to obtain sensitive data or allow pentester to infiltrate a system and manifest themselves on it, for example. Once a system has been successfully compromised, it is frequently possible to penetrate additional systems because the pen-testers now have access to more potential targets that were previously unavailable, such as the compromised system's ability to interact with internal systems that are not accessible via the Internet.

Post-Exploitation

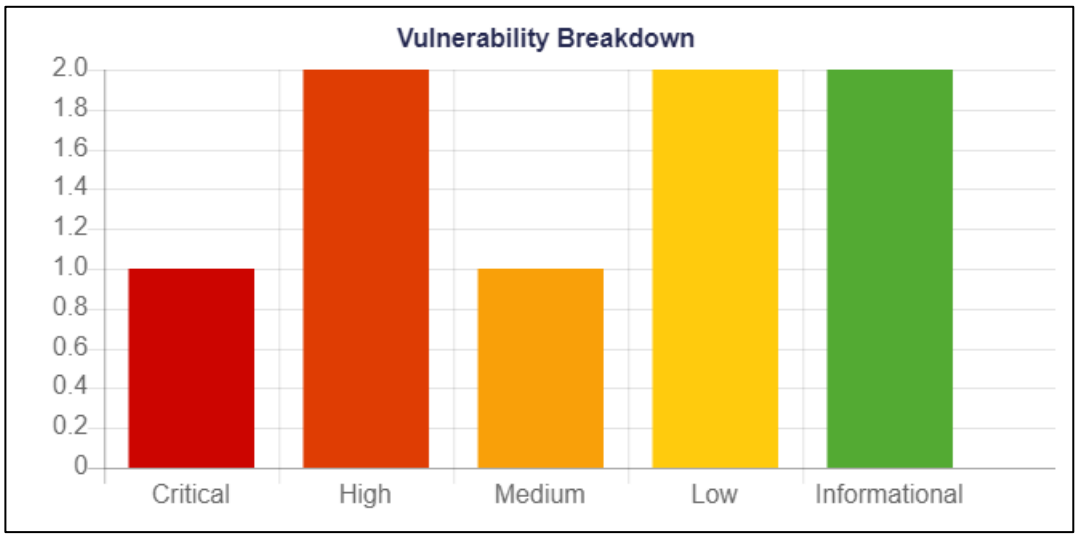
The pentesters next task is to establish the value of the entry point after he or she has exploited a vulnerability and located an entry point into the system. The tester can use the exploitation and post-exploitation phases to get access, locate sensitive data, and establish communication channels, among other things. They can also try to broaden the breach by exploiting the connectivity between other systems within the network. The rules of engagement agreed upon at the pre-engagement stage govern the extent to which a pentester may exploit a particular vulnerability.

Reporting

A Penetration Testing report is a document that offers a full analysis of the security flaws discovered during the test. It keeps track of the flaws, the threat they pose, and the steps that could be taken to address them. The Penetration Testing report provides a comprehensive summary of vulnerabilities, as well as a POC (Proof of Concept) and remedial recommendations to address such vulnerabilities as quickly as possible. It also assigns a score to each discovered flaw, based on how serious it is to your application/website.

Vulnerability Summary

The charts below are designed to provide a quick snapshot of the assessment. For information regarding risk ratings. Otherwise, for vulnerabilities because of this assessment, please see the Findings section.



Vulnerability List

ID	Vulnerability	Severity	Identified Date	Status
C1	Privilege Escalation to AWS Administrator	Critical (9.0)	13th February, 2023	Not Remediated
H1	EC2 User Data Sensitive Information Leakage	High (7.7)	12th February, 2023	Not Remediated
H2	AWS S3 Bucket Data Leakage	High (7.1)	13th February, 2023	Not Remediated
M1	No IAM User Access Key Rotation	Medium	17th February, 2023	Not Remediated
L1	S3 Bucket Access Logging Disabled	Low	17th February, 2023	Not Remediated
I2	Elastic Block Store (EBS) Snapshot Encryption Disabled	Low	17th February, 2023	Not Remediated
I1	EC2 Instance Termination Protection Disabled	Informational	16th February, 2023	Not Remediated
I2	CloudTrail Logging Disabled	Informational	16th February, 2023	Not Remediated

Findings

Privilege Escalation to AWS Administrator	9.0 (Critical)

Description

During the VAPT assessment of the target system, a privilege escalation vulnerability was identified that could potentially allow an attacker to escalate their privileges to gain AWS Administrator access. The vulnerability arises from inadequate access controls and misconfigured permissions within the AWS environment.

Impact

An attacker who successfully escalates their privileges to AWS Administrator level would gain full control over the AWS resources and services within the compromised environment. This could lead to unauthorized access, data breaches, loss of sensitive information, and disruption of critical services.

Recommendation

- Consider Implementation of principle of least privilege (PoLP) to grant users and applications the minimal permissions necessary to carry out their functions. Avoid assigning overly permissive policies.
- Utilize IAM roles with appropriate permissions instead of relying on long-term access keys whenever possible. This limits the exposure of sensitive credentials.
- Don't store private SSH keys in S3 buckets that are accessible by all AWS users.

Affected System and Endpoint

- S3 Buckets
  - backup-dev
- RDS Database
  - john-mgmt | arn:aws:rds:us-west-2:000000000000:db:branch
- IAM Users
  - ABC | arn:aws:iam::000000000000:user/john.doe
  - ABC | arn:aws:iam::000000000000:user/tony.stark
- EC2 Elastic Block Store Volumes
  - John-mgmt-ue1c | vol-5fce2f0b056348dc1 | us-west-2

- EC2 Instances
    - staging-ABC - station1 | i-5efc2e2bcca120f1f | us-west-2

Status

Not Remediated

CVSS:3.1/AV:N/AC:H/PR:N/UI:N/S:C/C:H/I:H/A:H

Steps for Reproduction

- List the Files In the s3 bucket from the test account.
- The "staging-ABC.pem" SSH private key was discovered, which was subsequently used to gain unauthorized access to the AWS environment.
- Utilizing the compromised private key, the assessor gained unauthorized access to the EC2 instance named "staging-ABC - station1" (instance ID: i0cba9e1bbbbb120f1f).
- Once inside the compromised instance, the assessor executed commands to create snapshots of existing Elastic Block Store (EBS) volumes.
- New EBS volumes were generated from the snapshots without detaching them from their associated instances, thus avoiding service disruption.
- The newly created EBS volumes were attached to the compromised EC2 instance.
- These volumes were mounted to a folder created by the assessor within the instance's file system.
- Among the explored volumes, a specific EBS volume (vol-0f06ef0a8562989c9) was identified.
- Within this volume, we discovered a ".bash_history" file containing a chronological record of executed commands.
- Within the ".bash_history" file, a command was found that stored database credentials for the "john-mgmt" RDS database in cleartext.
- During the analysis of the compromised "john-mgmt" RDS database, we identified a MySQL table named "audit_logs."
- The "audit_logs" table contained highly sensitive information, including an AWS access key ID and secret access key.
- The access key ID was associated with the IAM user "john.doe" (ARN: arn:aws:iam::0000000000:user/john.doe).
- We determined that the IAM user "john.doe" possessed administrator-

level access to the AWS environment.

Evidence

[Evidence Retracted]

References

Sensitive Information Leakage: EC2 User Data	7.7 (High)

Description

During the VAPT assessment of the target system, a vulnerability related to EC2 user data was identified. The vulnerability allowed sensitive information contained within user data scripts to be leaked. The vulnerability arises from inadequate handling of user data and lack of proper security measures.

Impact

The presence of hardcoded sensitive credentials in user data could lead to unauthorized access and potential data breaches. Attackers could use these credentials to gain unauthorized access to external resources, potentially leading to data leaks, unauthorized account access, and other security breaches.

Recommendation

- Consider not hardcoding sensitive information, such as usernames and passwords, in user data scripts.
- Utilize secure credential storage mechanisms, such as AWS Secrets Manager, to manage and retrieve sensitive information.

Affected System and Endpoint

- John-doe | i-f367e543 | us-west-2
- Foo-Bar | i-234e5fd4 | us-west-2
- Tony-Stark | i-64a9ca3e2 | us-west-2

Status

Not Remediated

CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:C/C:H/I:N/A:N

Steps for Reproduction

- We developed a script to scrape and decode user data associated with EC2 instances in the target AWS account.
- The script was executed to systematically extract and decode user data from each affected EC2 instance.
- During the manual verification process, we discovered username

and password within the user data.

- The username and password combination discovered in the user data were found to be identical across multiple affected EC2 instances.

Evidence
[Evidence Retracted]

References

AWS S3 Bucket Data Leakage	7.1 (High)

Description

During the VAPT assessment of the target AWS environment, we found a misconfiguration in the permissions for that bucket. Some common misconfigurations include public read access for files and public listing access for the buckets themselves. This can allow people to list the files in the bucket, as well as read the contents of them.

Impact

Misconfigured S3 bucket permissions can lead to unauthorized access to sensitive data, including proprietary information, customer data, financial records, and other confidential resources.

Recommendation

- Consider Implementing proper S3 bucket to ensure that only authorized users and services have access to the buckets. Remove unnecessary permissions and public access.
- Utilize Identity and Access Management (IAM) policies and S3 bucket policies to control access to buckets and objects. Apply the principle of least privilege to grant only the necessary permissions.

Affected System and Endpoint

- S3 Bucket (backup-dev)

Status

Not Remediated

CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:L/A:N

Steps for Reproduction

- Navigate to the endpoint s3://abc-devops/john-backup/*.

Evidence
[Evidence Retracted]

References

No IAM User Access Key Rotation	Medium

Description

During the VAPT assessment of the target AWS environment, a vulnerability related to the lack of IAM user access key rotation was identified. The vulnerability arises from the failure to regularly rotate access keys for IAM users, increasing the risk of unauthorized access and potential data breaches.

Impact

Failure to rotate access keys leaves the organization vulnerable to security breaches, as compromised or leaked access keys can be exploited by attackers to gain unauthorized access to AWS resources. This could result in data theft, service disruption, and unauthorized actions.

Recommendation

- Consider Implementing policy to regularly rotate IAM user access keys, requiring keys to be changed at predefined intervals.
- Ensure the IAM users are using use MFA for key rotation and other sensitive actions, enhancing security.

Affected System and Endpoint

- IAM Users
  - John-doe
  - Tony-Stark

Status

Not Remediated

Steps for Reproduction

- Navigate to the Admin console of the AWS account.
- Analyze the "Access Key Age" settings for the affected IAM users.

Evidence
[Evidence Retracted]

References

S3 Bucket Access Logging Disabled	Low

Description

During the VAPT assessment of the target AWS environment, a vulnerability related to S3 bucket access logging was identified. The vulnerability stems from the absence of access logging for S3 buckets, leading to a lack of visibility into who accessed the buckets and what actions were performed.

Impact

Without access logging, the organization is unable to monitor and audit actions performed on S3 buckets. This lack of visibility increases the risk of unauthorized access, data breaches, and undetected malicious activities.

Recommendation

- Configure access logging for all relevant S3 buckets to capture detailed records of all requests made to the buckets.

Affected System and Endpoint

- S3 Buckets
  - John-backup
  - abc-prod
  - dc-dev

Status

Not Remediated

Steps for Reproduction

- Navigate to the Admin console of the AWS account.
- Analyze the settings associated with the affected buckets, the access logging was disabled.

Evidence
[Evidence Retracted]

References

Elastic Block Store (EBS) Snapshot Encryption Disabled	Low

Description

During the VAPT assessment of the target AWS environment, a vulnerability related to Elastic Block Store (EBS) snapshot encryption was identified. The vulnerability arises from the absence of encryption for EBS snapshots, potentially exposing sensitive data stored within these snapshots.

Impact

EBS snapshot encryption provides a critical layer of security, safeguarding data against unauthorized access and potential data breaches. Without encryption, sensitive information stored in EBS snapshots could be exposed if snapshots fall into the wrong hands, leading to data leaks.

Recommendation

- Enable encryptions for all EBS snapshots, ensuring that data is encrypted at rest.

Affected System and Endpoint

- EBS Snapshots
  - snap-12320a | us-west-2
  - snap-12301a | us-west-2
  - snap-122312a | us-west-2

Status

Not Remediated

Steps for Reproduction

- Navigate to the Admin console of the AWS account.
- Analyze the settings associated with the affected EBS snapshots which were not encrypted

Evidence
[Evidence Retracted]

References

EC2 Instance Termination Protection Disabled	Informational

Description

During the VAPT assessment of the target AWS environment, a vulnerability related to EC2 instance termination protection was identified. The vulnerability stems from the absence of termination protection for EC2 instances, which can lead to accidental or unauthorized termination of critical instances.

Impact

Without termination protection, EC2 instances are susceptible to accidental or intentional termination, leading to service disruptions, data loss, and potential security breaches. Instances hosting critical applications or services may be rendered inaccessible.

Recommendation

- Consider enabling termination protection for all critical EC2 instances that should not be terminated accidentally or without proper authorization.

Affected System and Endpoint

- EC2 Instances
  - north-side | i-0adbfff21sq3a7f9 | us-west-2
  - south-side | i-0adbee1221sq3a7f9 | us-west-2

Status

Not Remediated

Steps for Reproduction

- Navigate to the Admin console of the AWS account.
- Review the termination protection setting for the affected EC2 instances.

Evidence
[Evidence Retracted]

References

CloudTrail Logging Disabled	Informational

Description

During the VAPT assessment of the target AWS environment, a vulnerability related to CloudTrail logging was identified. It was observed that no active CloudTrail trails were enabled through the CloudTrail API. This means that critical AWS activity and event logs were not being captured, creating a blind spot in security monitoring and incident response.

Impact

The impact of this vulnerability can be significant. Disabling CloudTrail logging leaves the environment susceptible to undetected unauthorized activities, security breaches, and potential data exfiltration. The lack of comprehensive logs hampers the organization's ability to conduct effective incident investigations, trace the origin of security incidents, and maintain compliance with regulatory requirements.

Recommendation

- Consider enabling termination protection for all critical EC2 instances that should not be terminated accidentally or without proper authorization.
- Set up monitoring and alerts for CloudTrail events to promptly detect and respond to any security incidents or unauthorized activities.

Affected System and Endpoint

- CloudTrail Service

Status

Not Remediated

Steps for Reproduction

- No Active Trails were enabled through the CloudTrail API.

Evidence
[Evidence Retracted]

References

Test Cases

- Test for Unauthenticated database access
- Test for Improper permissions for Database
- Test for compromising access keys
- Test for extracting keys from a VM / instance
- Test for exploits due to improper configs.
- Testing for public exploits in VM / instances
- Test for backdoors exploitation internally
- Test for Subdomain Takeover
- Test for access mgmt. Privilege Escalation
- Test for Remote Code Execution (RCE)
- Test for Role Enumeration
- Test for VM service Privilege Escalation
- Test for IAM Enumeration
- Test for local Windows/Linux logs change
- Test for Management Service Privileges
- Test for loopholes that add root certificates and SSH private keys to VMs and users.
- Test to create or reset a login, access key, or temporary credential belonging to a high privilege user (like IAM: CreateAccessKey, STS, or IAM: UpdateLoginProfile)
- Test for disabled network traffic analysis/logging (VPC Flow Logs)
- Test for disabled Cloud Alerting to prevent detection and response.
- Test for disabled data store access logging to prevent detection and
- response (CloudTrail Data Access, S3 Access Logging, etc.)
- Test for unauthenticated obtaining of the VM images from storage accounts and do an analysis for passwords, keys, certificates to penetrate and access live resources.
- Test to alter log retention or damage the integrity of logs (S3 lifecycle,
- KMS decryption, CMK key deletion/role privilege lockout)

Conclusion

During the period of Vulnerability Assessment and Penetration Testing, we found a total of one critical, two high, one medium, two low and two informational issues in the AWS Infrastructure. We have identified vulnerabilities mainly due to misconfigured IAM policies granting excessive permissions, Failure to encrypt sensitive information, public access to S3 bucket, Not monitoring access activities and events and Failing to enable CloudTrail logging for monitoring and auditing.

Here, to mitigate these vulnerabilities, we recommend our client to perform the following recommendations:

- Consider Implementation of principle of least privilege (PoLP) to grant users and applications the minimal permissions necessary to carry out their functions. Avoid assigning overly permissive policies.
- Utilize IAM roles with appropriate permissions instead of relying on long-term access keys whenever possible. This limits the exposure of sensitive credentials.
- Don't store private SSH keys in S3 buckets that are accessible by all AWS users.
- Consider Implementing proper S3 bucket to ensure that only authorized users and services have access to the buckets. Remove unnecessary permissions and public access.
- Consider Implementing policy to regularly rotate IAM user access keys, requiring keys to be changed at predefined intervals.
- Configure access logging for all relevant S3 buckets to capture detailed records of all requests made to the buckets.
- Enable encryptions for all EBS snapshots, ensuring that data is encrypted at rest.
- Consider enabling termination protection for all critical EC2 instances that should not be terminated accidentally or without proper authorization.
- Consider enabling termination protection for all critical EC2 instances that should not be terminated accidentally or without proper authorization.

[ORG NAME] should ensure that the vulnerabilities mentioned in this report will be patched and should re-test each issue to verify that the vulnerabilities have been successfully patched. The current implementation of the recommendations contained in this report, along with continued diligence on the part of [ORG NAME], will result in the improvement of the security posture.

Appendix

Common vulnerability and Exposure

Common Vulnerabilities and Exposures (CVE) is a dictionary-type list of standardized names for vulnerabilities and other information related to security exposures. CVE aims to standardize the names for all publicly known vulnerabilities and security exposures. The goal of CVE is to make it easier to share data across separate vulnerability capabilities (tools, repositories, and services) with this common enumeration.

CVSS

The Common Vulnerability Scoring System (CVSS) is an open framework for communicating the characteristics and severity of software vulnerabilities. CVSS attempts to assign severity scores to vulnerabilities, allowing responders to prioritize responses and resources according to threat. Scores are calculated based on a formula that depends on several metrics that approximate ease of exploit and the impact of exploit. Scores range from 0 to 10, with 10 being the most severe.

CVSS V3.1 Severity Rating

Severity	Score
Critical	**9.0-10**
High	**7.0-8.9**
Medium	**4.0-6.9**
Low	**0.1-3.9**
Informational	

OUR SERVICES

Our Services As Information Security Company Includes:

- SECURITY OPERATIONS CENTER
- INFORMATION SECURITY AUDIT
- SWIFT CSP ASSESSMENT
- DARKWEB MONITORING & BRAND PROTECTION
- VULNERABILITY MANAGEMENT
- PENETRATION TESTING
- INCIDENT RESPONSE
- THREAT ANALYSIS
- SERVER CONFIGURATION ASSESSMENT
- CYBER SECURITY CONSULTANT
- INFORMATION SECURITY TRAINING

CryptoGen Nepal

/cryptogennepal
www.cryptogennepal.com
+977-1-4528928
whois@cryptogennepal.com