CIS. Center for Internet Security®

CIS Benchmarks™

CIS AWS Database Services Benchmark

v2.0.0 - 12-16-2025

Terms of Use

Please see the below link for our current terms of use:

https://www.cisecurity.org/cis-securesuite/cis-securesuite-membership-terms-of-use/

For information on referencing and/or citing CIS Benchmarks in 3rd party documentation (including using portions of Benchmark Recommendations) please contact CIS Legal (legalnotices@cisecurity.org) and request guidance on copyright usage.

NOTE: It is ***NEVER*** acceptable to host a CIS Benchmark in ***ANY*** format (PDF, etc.) on a 3rd party (non-CIS owned) site.

Internal Only - General

Table of Contents

Internal Only - General

Internal Only - General

Internal Only - General

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 like:

- Monitoring the base operating system and applications for vulnerabilities and quickly updating with the latest security patches.
- End-point protection (Antivirus software, Endpoint Detection and Response (EDR), etc.).
- Logging and monitoring user and system activity.

In the end, the Benchmarks are designed to be a key **component** of a comprehensive cybersecurity program.

Important Usage Information

All Benchmarks are available free for non-commercial use from the CIS Website. They can be used to manually assess and remediate systems and applications. In lieu of manual assessment and remediation, there are several tools available to assist with assessment:

- CIS Configuration Assessment Tool (CIS-CAT® Pro Assessor)
- CIS Benchmarks™ Certified 3rd Party Tooling

These tools make the hardening process much more scalable for large numbers of systems and applications.

NOTE: Some tooling focuses only on the Benchmark Recommendations that can be fully automated (skipping ones marked **Manual**). It is important that ***ALL*** Recommendations (**Automated** and **Manual**) be addressed since all are important for properly securing systems and are typically in scope for audits.

Key Stakeholders

Cybersecurity is a collaborative effort, and cross functional cooperation is imperative within an organization to discuss, test, and deploy Benchmarks in an effective and efficient way. The Benchmarks are developed to be best practice configuration guidelines applicable to a wide range of use cases. In some organizations, exceptions to specific Recommendations will be needed, and this team should work to prioritize the problematic Recommendations based on several factors like risk, time, cost, and labor. These exceptions should be properly categorized and documented for auditing purposes.

Internal Only - General

Apply the Correct Version of a Benchmark

Benchmarks are developed and tested for a specific set of products and versions and applying an incorrect Benchmark to a system can cause the resulting pass/fail score to be incorrect. This is due to the assessment of settings that do not apply to the target systems. To assure the correct Benchmark is being assessed:

- **Deploy the Benchmark applicable to the way settings are managed in the environment:** An example of this is the Microsoft Windows family of Benchmarks, which have separate Benchmarks for Group Policy, Intune, and Stand-alone systems based upon how system management is deployed. Applying the wrong Benchmark in this case will give invalid results.

- **Use the most recent version of a Benchmark**: This is true for all Benchmarks, but especially true for cloud technologies. Cloud technologies change frequently and using an older version of a Benchmark may have invalid methods for auditing and remediation.

Exceptions

The guidance items in the Benchmarks are called recommendations and not requirements, and exceptions to some of them are expected and acceptable. The Benchmarks strive to be a secure baseline, or starting point, for a specific technology, with known issues identified during Benchmark development are documented in the Impact section of each Recommendation. In addition, organizational, system specific requirements, or local site policy may require changes as well, or an exception to a Recommendation or group of Recommendations (e.g. A Benchmark could Recommend that a Web server not be installed on the system, but if a system's primary purpose is to function as a Webserver, there should be a documented exception to this Recommendation for that specific server).

In the end, exceptions to some Benchmark Recommendations are common and acceptable, and should be handled as follows:

- The reasons for the exception should be reviewed cross-functionally and be well documented for audit purposes.
- A plan should be developed for mitigating, or eliminating, the exception in the future, if applicable.
- If the organization decides to accept the risk of this exception (not work toward mitigation or elimination), this should be documented for audit purposes.

It is the responsibility of the organization to determine their overall security policy, and which settings are applicable to their unique needs based on the overall risk profile for the organization.

Internal Only - General

Remediation

CIS has developed Build Kits for many technologies to assist in the automation of hardening systems. Build Kits are designed to correspond to Benchmark's "Remediation" section, which provides the manual remediation steps necessary to make that Recommendation compliant to the Benchmark.

When remediating systems (changing configuration settings on deployed systems as per the Benchmark's Recommendations), please approach this with caution and test thoroughly.

The following is a reasonable remediation approach to follow:

- CIS Build Kits, or internally developed remediation methods should never be applied to production systems without proper testing.
- Proper testing consists of the following:
  - Understand the configuration (including installed applications) of the targeted systems. Various parts of the organization may need different configurations (e.g., software developers vs standard office workers).
  - Read the Impact section of the given Recommendation to help determine if there might be an issue with the targeted systems.
  - Test the configuration changes with representative lab system(s). If issues arise during testing, they can be resolved prior to deploying to any production systems.
  - When testing is complete, initially deploy to a small sub-set of production systems and monitor closely for issues. If there are issues, they can be resolved prior to deploying more broadly.
  - When the initial deployment above is completes successfully, iteratively deploy to additional systems and monitor closely for issues. Repeat this process until the full deployment is complete.

Summary

Using the Benchmarks Certified tools, working as a team with key stakeholders, being selective with exceptions, and being careful with remediation deployment, it is possible to harden large numbers of deployed systems in a cost effective, efficient, and safe manner.

NOTE: As previously stated, the PDF versions of the CIS Benchmarks™ are available for free, non-commercial use on the CIS Website. All other formats of the CIS Benchmarks™ (MS Word, Excel, and Build Kits) are available for CIS SecureSuite® members.

CIS-CAT® Pro is also available to CIS SecureSuite® members.

Internal Only - General

Target Technology Details

This document provides prescriptive guidance for configuring security options for the services within the Database category in AWS. This Benchmark is intended to be used in conjunction with the CIS Amazon Web Services Foundations Benchmark. For more information about this approach see the Introduction section of this document.

The specific AWS Services in scope for this document include:

- Amazon Aurora
- Amazon DocumentDB
- Amazon DynamoDB
- Amazon ElastiCache
- Amazon Keyspaces (for Apache Cassandra)
- Amazon MemoryDB for Redis
- Amazon Neptune
- Amazon RDS
- Amazon Timestream

To obtain the latest version of this guide, please visit http://benchmarks.cisecurity.org. If you have questions, comments, or have identified ways to improve this guide, please write us at benchmarkinfo@cisecurity.org.

Intended Audience

This document is intended for system and application administrators, security specialists, auditors, help desk, platform deployment, and/or DevOps personnel who plan to develop, deploy, assess, or secure solutions in Amazon Web Services.

Internal Only - General

Consensus Guidance

This CIS Benchmark™ was created using a consensus review process comprised of a global community of subject matter experts. The process combines real world experience with data-based information to create technology specific guidance to assist users to secure their environments. Consensus participants provide perspective from a diverse set of backgrounds including consulting, software development, audit and compliance, security research, operations, government, and legal.

Each CIS Benchmark undergoes two phases of consensus review. The first phase occurs during initial Benchmark development. During this phase, subject matter experts convene to discuss, create, and test working drafts of the Benchmark. This discussion occurs until consensus has been reached on Benchmark recommendations. The second phase begins after the Benchmark has been published. During this phase, all feedback provided by the Internet community is reviewed by the consensus team for incorporation in the Benchmark. If you are interested in participating in the consensus process, please visit https://workbench.cisecurity.org/.

Typographical Conventions

The following typographical conventions are used throughout this guide:

Convention	Meaning
`Stylized Monospace font`	Used for blocks of code, command, and script examples. Text should be interpreted exactly as presented.
`Monospace font`	Used for inline code, commands, UI/Menu selections or examples. Text should be interpreted exactly as presented.
`<Monospace font in brackets>`	Text set in angle brackets denote a variable requiring substitution for a real value.
Italic font	Used to reference other relevant settings, CIS Benchmarks and/or Benchmark Communities. Also, used to denote the title of a book, article, or other publication.
Bold font	Additional information or caveats things like **Notes**, **Warnings**, or **Cautions** (usually just the word itself and the rest of the text normal).

Recommendation Definitions

The following defines the various components included in a CIS recommendation as applicable. If any of the components are not applicable it will be noted, or the component will not be included in the recommendation.

Title

Concise description for the recommendation's intended configuration.

Assessment Status

An assessment status is included for every recommendation. The assessment status indicates whether the given recommendation can be automated or requires manual steps to implement. Both statuses are equally important and are determined and supported as defined below:

Automated

Represents recommendations for which assessment of a technical control can be fully automated and validated to a pass/fail state. Recommendations will include the necessary information to implement automation.

Manual

Represents recommendations for which assessment of a technical control cannot be fully automated and requires all or some manual steps to validate that the configured state is set as expected. The expected state can vary depending on the environment.

Profile

A collection of recommendations for securing a technology or a supporting platform. Most benchmarks include at least a Level 1 and Level 2 Profile. Level 2 extends Level 1 recommendations and is not a standalone profile. The Profile Definitions section in the benchmark provides the definitions as they pertain to the recommendations included for the technology.

Description

Detailed information pertaining to the setting with which the recommendation is concerned. In some cases, the description will include the recommended value.

Rationale Statement

Detailed reasoning for the recommendation to provide the user a clear and concise understanding on the importance of the recommendation.

Impact Statement

Any security, functionality, or operational consequences that can result from following the recommendation.

Audit Procedure

Systematic instructions for determining if the target system complies with the recommendation.

Remediation Procedure

Systematic instructions for applying recommendations to the target system to bring it into compliance according to the recommendation.

Default Value

Default value for the given setting in this recommendation, if known. If not known, either not configured or not defined will be applied.

References

Additional documentation relative to the recommendation.

CIS Critical Security Controls® (CIS Controls®)

The mapping between a recommendation and the CIS Controls is organized by CIS Controls version, Safeguard, and Implementation Group (IG). The Benchmark in its entirety addresses the CIS Controls safeguards of (v7) “5.1 - Establish Secure Configurations” and (v8) '4.1 - Establish and Maintain a Secure Configuration Process” so individual recommendations will not be mapped to these safeguards.

Additional Information

Supplementary information that does not correspond to any other field but may be useful to the user.

Internal Only - General

Profile Definitions

The following configuration profiles are defined by this Benchmark:

- **Level 1**

  Items in this profile intend to:

  - be practical and prudent;
  - provide security-focused best practice hardening of a technology; and
  - limit the impact to the utility of the technology beyond acceptable means.

- **Level 2**

  This profile extends the "Level 1" profile. Items in this profile exhibit one or more of the following characteristics:

  - are intended for environments or use cases where security is more critical than manageability and usability
  - acts as a defense in depth measure
  - may impact the utility or performance of the technology
  - may include additional licensing, cost, or addition of third-party software

Internal Only - General

Acknowledgements

This Benchmark exemplifies the great things a community of users, vendors, and subject matter experts can accomplish through consensus collaboration. The CIS community thanks the entire consensus team with special recognition to the following individuals who contributed greatly to the creation of this guide:

Gregory Carpenter, Mike Wicks, Michelle Peterson, Chantel Duckworth

Author
Chantel Duckworth

Contributor
Ian McRee
Mike Wicks
Krishna Rayavaram
Gregory Carpenter
Jun Woo Lee
Anunay Bhatt

Recommendations

1 Introduction

Benchmark Approach:

The suggested approach for securing your cloud environment is to start with the CIS Amazon Web Services Foundations Benchmark found here: https://www.cisecurity.org/benchmark/amazon_web_services/. The CIS Foundations benchmark provides prescriptive guidance for configuring a subset of Amazon Web Services with an emphasis on foundational, testable, and architecture agnostic settings including:

- AWS Identity and Access Management (IAM)
- AWS Config
- AWS CloudTrail
- AWS CloudWatch
- AWS Simple Notification Service (SNS)
- AWS Simple Storage Service (S3)
- AWS VPC (Default)

The Amazon Web Services Foundation Benchmark is what you should start with when setting up your AWS environment. It is also the foundation for which all other AWS service based benchmarks are built on so that as you grow your cloud presence and usage of the services offered you have the necessary guidance to securely configure your environment as it fits with your company's policy.

After configuring your environment to the CIS Amazon Web Services Foundations Benchmark, we suggest implementing the necessary configurations for the services utilized as defined in the associated product and service level benchmarks. The CIS AWS Database Services Benchmark provides prescriptive guidance for configuring security options for the services within Databases in AWS. The specific AWS Services in scope for this document include:

- Amazon Aurora
- Amazon DocumentDB
- Amazon DynamoDB
- Amazon ElastiCache
- Amazon Keyspaces (for Apache Cassandra)
- Amazon MemoryDB for Redis
- Amazon Neptune
- Amazon RDS
- Amazon Timestream

All CIS Benchmarks are created and maintained through consensus-based collaboration. Should you have feedback, suggested changes, or just like to get involved in the continued maintenance and development of CIS Amazon Web Services Benchmarks, please register on CIS WorkBench at https://workbench.cisecurity.org and join the CIS Amazon Web Services Benchmarks community.

2 Amazon Aurora

Amazon Aurora is a relational database service provided by Amazon Web Services (AWS) that is designed for high performance, availability, and scalability. It is compatible with MySQL and PostgreSQL, which means you can use existing MySQL or PostgreSQL applications, drivers, and tools with Aurora with minimal modification.

Internal Only - General

2.1 Ensure the Use of Security Groups (Manual)

Profile Applicability:

- Level 1

Description:

Security groups act as a firewall for associated Amazon RDS DB instances, controlling both inbound and outbound traffic.

Rationale:

Creating your severity group either inbound or outbound rules. Inbound rules allow an individual to create a rule that permits the traffic to go to a specific port depending on which source it's coming from. Outbound rules enable your instances to connect with one another allow them to connect to the internet. If needed, you can limit the outgoing traffic.

Audit:

1. Open the Amazon Console
2. Go to Aurora and RDS (https://console.aws.amazon.com/rds/)
3. Click on Databases
4. For each database instance click the name of the instance and check that there is at least one VPC security group under Connectivity & security -> Security -> VPC security groups

Remediation:

Here is a step-by-step guide on how to create and use Security Groups for an Amazon Aurora instance:

1. Sign in to AWS Management Console

   If you do not already have an AWS account, you'll need to create one at https://aws.amazon.com.

2. Navigate to Amazon EC2 Dashboard

   Once you have logged in to the AWS Management Console, navigate to the EC2 service. You can find this under the `Compute` category.

3. Create a New Security Group

- In the EC2 Dashboard, find the `Network & Security` section on the left-side navigation pane, then click `Security Groups`.
- Click on the `Create Security Group` button.

4. Configure the New Security Group

- In the `Create Security Group` panel, give your new security group a name and a description.
- Select the VPC in which your Amazon Aurora instance will be deployed.
- Then click `Create`.

5. Add Rules to the Security Group

   After creating the Security Group, you can add inbound and outbound rules. For Inbound Rules:

- Click on the `Inbound rules` tab, then click `Edit inbound rules`.
- Click `Add Rule`. For the type, select MYSQL/Aurora. For the source, you can specify the IP addresses allowed to access your Amazon Aurora instance.

   For Outbound Rules:

- Click on the `Outbound rules` tab, then click `Edit outbound rules`. Outbound rules allow your instances to communicate with other instances or access the internet. You can restrict outbound traffic if necessary. In most cases, you can leave the default setting, which allows all outbound traffic.

6. Assign the Security Group to Amazon Aurora

- When launching a new Amazon Aurora instance (in the Amazon RDS dashboard), you can select your new security group in the `Configure advanced settings` step.
- If your Aurora instance has already been launched, you can modify it to use the new security group by selecting the instance.
- Click `Modify`, and then select the new security group.

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●

Controls Version	Control	IG 1	IG 2	IG 3
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

2.2 Ensure Data at Rest is Encrypted (Manual)

Profile Applicability:

- Level 1

Description:

Amazon Aurora allows you to encrypt your databases using keys you manage through AWS Key Management Service (KMS).

Rationale:

Databases are likely to hold sensitive and critical data; therefore, it is highly recommended to implement encryption to protect your data from unauthorized access or disclosure.

Impact:

Unauthorized users cannot access the data because it is protected by an encryption key that only authorized users can use.

Audit:

1. Sign in to the AWS Management Console where the Aurora database cluster you are auditing resides.
2. Navigate to the Amazon Aurora and RDS Dashboard:

- You can find this under the Database category.

3. Select the DB cluster name you wish to audit:

- This opens the details page for your specific Aurora cluster.

4. Check the encryption status under the Configuration section.

- Confirm that the field Encryption is marked as Enabled.

5. Verify KMS key usage (if your organization's standards require a customer-managed key):

- In the Encryption section, identify the AWS KMS key associated with the cluster. Click the key link to open its details page. Confirm that it is a customer-managed KMS key, not an AWS-managed key.
- Review additional key attributes to ensure compliance, including Key policy, Key rotation status, etc.

Internal Only - General

Remediation:

For existing Aurora databases: In order to encrypt an existing Aurora instance that was not initially created with encryption enabled, you will need to create a snapshot of the instance, make a copy of the snapshot with encryption enabled, and then restore the DB instance from the copied snapshot.

For creating new Aurora databases with encryption at rest enabled:

1. Sign in to AWS Management Console

   If you do not already have an AWS account, you'll need to create one at https://aws.amazon.com

2. Navigate to the Amazon Aurora and RDS Dashboard:

- You can find this under the Database category.

3. Click on `Create Database` and choose Aurora as your engine option.
4. In the `Additional Configuration` section, you will see an option labeled `Enable encryption`. Check this box to enable encryption for data at rest.

- You will also need to select a master key to use for encryption. You can choose the default AWS managed key for RDS or a pre-created customer-managed AWS Key Management Service (KMS) that aligns with your organization’s key management policies.

5. Launch the DB Instance

- After you have selected the appropriate encryption settings, click `Create database`.
- Review your settings on the following page, and if everything looks correct, click `Launch DB Instance`.

References:

1. https://aws.amazon.com/products/databases/

Internal Only - General

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.11 Encrypt Sensitive Data at Rest Encrypt sensitive data at rest on servers, applications, and databases containing sensitive data. Storage-layer encryption, also known as server-side encryption, meets the minimum requirement of this Safeguard. Additional encryption methods may include application-layer encryption, also known as client-side encryption, where access to the data storage device(s) does not permit access to the plain-text data.		●	●
v7	14.8 Encrypt Sensitive Information at Rest Encrypt all sensitive information at rest using a tool that requires a secondary authentication mechanism not integrated into the operating system, in order to access the information.			●

Internal Only - General

2.3 Ensure Data in Transit Encryption is Enforced (Manual)

Profile Applicability:

- Level 1

Description:

Use TLS (Transport Layer Security) to secure data in transit. Aurora supports TLS-encrypted connections between your application and your DB instance, and this configuration can be enforced so non-TLS connections are prohibited.

Rationale:

Encrypting data in transit protects sensitive information from interception and tampering by unauthorized parties. Aurora supports TLS for securing client connections, however it is essential to ensure that client applications are properly configured to use TLS and that the database enforces encrypted connections.

Impact:

Disabling or failing to properly configure TLS can expose the data to be compromised by malicious actors, potentially resulting in data breaches, credential theft, or other security compromises.

Audit:

1. Sign in to the AWS Management Console where the Aurora database cluster you are auditing resides.
2. Navigate to the Amazon Aurora and RDS Dashboard:

- You can find this under the Database category.

3. Select the DB cluster name you wish to audit:

- This opens the details page for your specific Aurora cluster.

4. Under Configuration, locate the DB cluster parameter group attached to this cluster:

- Click on the parameter group name to review its parameters.

5. Verify the following engine-specific parameters to confirm encryption in transit is enforced:

- 5a. PostgreSQL: Confirm that rds.force_ssl = 1
- 5b. MySQL: Confirm that require_secure_transport = ON (Only applicable for Aurora MySQL versions 2 and 3)

Notes:

- Make sure the parameter changes are applied and the cluster has been rebooted if necessary for the parameters to take effect.

Remediation:

1. Sign in to the AWS Management Console where the Aurora database cluster you are remediating resides.
2. Navigate to the Amazon Aurora and RDS Dashboard:

- You can find this under the Database category.

3. Select the DB cluster name you wish to remediate:

- This opens the details page for your specific Aurora cluster.

4. Under Configuration, locate the DB cluster parameter group attached to this cluster:

- Click on the parameter group name to remediate its parameters.

5. Update the following engine-specific parameters to enforce encryption in transit at the database level:

- PostgreSQL: Set rds.force_ssl = 1
- MySQL: Set require_secure_transport = ON (applicable to Aurora MySQL versions 2 and 3 only).

6. Reboot the database cluster to apply the parameter changes.
7. Configure your client application for SSL/TLS connections:

- Download the appropriate AWS-provided SSL/TLS certificates:

  For MySQL-compatible Aurora, Amazon provides an SSL certificate that you can download from their documentation. PostgreSQL-compatible Aurora uses the default PostgreSQL SSL certificate.

- Once you have the appropriate certificate, you must configure your client application to use SSL/TLS. For example, in MySQL, you might use a command like this:

```
mysql -h <myinstance.123456789012.us-east-1.rds.amazonaws.com> --ssl-
ca=</path_to_certificate/rds-combined-ca-bundle.pem> --ssl-
mode=VERIFY_IDENTITY
```

For PostgreSQL, you might use a command like this:

```
psql "host=<myinstance.123456789012.us-east-1.rds.amazonaws.com>
sslmode=verify-ca sslrootcert=</path_to_certificate/rds-combined-ca-
bundle.pem>"
```

Replace <myinstance.123456789012.us-east-1.rds.amazonaws.com> with the endpoint for your DB instance, and replace </path_to_certificate/rds-combined-ca-bundle.pem> with the path to the SSL certificate on your local machine.

8. Verify Encryption After configuring your client to use SSL/TLS, you should verify that encryption in transit is working correctly. You can do this by checking the status of the SSL connection from within the database itself. For example, in MySQL, you can run the following command:

```
SHOW STATUS LIKE 'Ssl_cipher';
```

In PostgreSQL, you can run the following command:

```
SHOW ssl;
```

In both cases, if SSL is enabled, you should see a non-empty cipher suite or on as a result.

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.10 Encrypt Sensitive Data in Transit Encrypt sensitive data in transit. Example implementations can include: Transport Layer Security (TLS) and Open Secure Shell (OpenSSH).		●	●
v7	14.4 Encrypt All Sensitive Information in Transit Encrypt all sensitive information in transit.		●	●

2.4 Ensure IAM Roles and Policies are Created (Manual)

Profile Applicability:

- Level 1

Description:

AWS Identity and Access Management (IAM) helps manage access to AWS resources. While you cannot directly associate IAM roles with Amazon Aurora instances, you can use IAM roles and policies to define which AWS IAM users and groups have management permissions for Amazon RDS resources and what actions they can perform. Here is a guide:

Rationale:

Individual creates IAM roles and polices that define specific permission given to that role. This determines what the identity or instance can and cannot do.

Impact:

If an IAM Role is not created, then it would be challenging to access AWS resources.

Audit:

1. Sign in to AWS Management Console

- If you do not already have an AWS account, you will need to create one at https://aws.amazon.com.

2. Navigate to IAM Dashboard

- Navigate to the IAM service once logged in to the AWS Management Console.
- This is under the `Security, Identity, & Compliance` category.

3. Create a New IAM Role

- In the IAM Dashboard, find the `Roles` section on the left-side navigation pane and click on it. Then, click on the `Create Role` button.

4. Select the Service that will Use the Role

- Choose `RDS` as the AWS service that will use this new role, then click `Next: Permissions`.

5. Attach Policy

- In the next screen, you can attach policies defining this role's permissions. You can use the filter to find existing policies like `AmazonRDSFullAccess` or `AmazonRDSReadOnlyAccess`.
- Select the appropriate policy and then click `Next: Tags`.

6. Add Tags (Optional)

- You can add metadata to the role by attaching tags as key-value pairs. This is optional, and you can proceed to the next step if you do not wish to add tags.

7. Review

- Provide a name and a description for the role. Review the role and then click `Create Role`.

8. Creating IAM Policy (Optional)

- You can create a custom IAM policy if the predefined policies do not meet your requirements.
- Navigate to `Policies` in the IAM dashboard and click `Create Policy`.
- Use the visual editor or JSON editor to define the permissions.
- Once done, click `Review Policy`, give it a name and a description, and click `Create Policy`.
- You can then attach this custom policy to the IAM role.

9. Assign the IAM Role to an IAM User or Group

   To assign the newly created role to an IAM User or Group.

- Navigate to the user or group in the IAM dashboard.
- Click `Add permissions`.
- Then `Attach existing policies directly`.
- Use the filter to find your new role and select it.
- Click `Next: Review` and then `Add permissions`.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

2.5 Ensure Database Audit Logging is Enabled (Manual)

Profile Applicability:

- Level 1

Description:

Amazon Aurora provides advanced auditing capabilities through AWS CloudTrail and Amazon RDS Database Activity Streams. Here is a step-by-step guide on how to enable and use these features:

Rationale:

Allows individuals to access and retrieve their old logs, log their new events, and store their log.

Audit:

Below are the instructions for enabling logging through AWS CloudTrail:

1. Sign in to AWS Management Console

- If you do not already have an AWS account, you will need to create one at https://aws.amazon.com.

2. Navigate to CloudTrail Dashboard

- Navigate to the CloudTrail service.
- You can find this under the `Management & Governance` category.

3. Create a new trail

- In the CloudTrail Dashboard, click on `Create trail`.
- Provide a name for the trail, and specify the S3 bucket where you want the logs to be stored.

4. Configure trail settings

- Choose the settings that meet your requirements. For instance, you can log events for all regions, or you can log management events, data events, or both.

5. Create the trail

- After specifying the trail settings, click `Create`.

Below are the instructions for enabling logging through Amazon Database Activity Streams:

Internal Only - General

1. Navigate to Amazon RDS Dashboard

- In the AWS Management Console, navigate to the RDS service.
- You can find this under the `Database` category.

2. Choose your Aurora DB instance

- In the RDS Dashboard, click on `Databases`, and then click on the name of your Aurora DB instance.

3. Enable Database Activity Streams

- In the `Connectivity & Security` tab, find the `Database Activity Streams` section. Click `Create stream`.
- In the `Create Stream` panel, choose the settings that meet your requirements and click `Create`.

Note: Enabling Database Activity Streams can impact the performance of your DB instance, so you should test this feature in a non-production environment before enabling it in production.

4. View the Database Activity Stream

- You can view the Database Activity Stream using Amazon Kinesis Data Streams.
- In the Kinesis Data Streams dashboard, click on the stream's name and then click `View data`.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	8.1 Establish and Maintain an Audit Log Management Process Establish and maintain an audit log management process that defines the enterprise's logging requirements. At a minimum, address the collection, review, and retention of audit logs for enterprise assets. Review and update documentation annually, or when significant enterprise changes occur that could impact this Safeguard.	●	●	●

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●

2.6 Ensure Passwords are Regularly Rotated (Manual)

Profile Applicability:

- Level 1

Description:

Regularly rotating your Aurora passwords is critical to access management, contributing to maintaining system security. The database password can be rotated in Amazon Aurora, but the access keys refer to the rotation of AWS IAM User access keys.

Rationale:

Updating your password is critical to access AWS resources. This also ensures that your account is being kept safe from a potential threat.

Impact:

Having the passwords updated frequently allows only the authorized individual to access the AWS resources.

Audit:

1. Sign in to AWS Management Console

- If you do not already have an AWS account, you will need to create one at https://aws.amazon.com.

2. Navigate to Amazon RDS Dashboard

- Navigate to the RDS service once logged in to the AWS Management Console. You can find this under the `Database` category.

3. Choose your Aurora DB instance

- In the RDS Dashboard, click on `Databases`, and then click on the name of your Aurora DB instance.

4. Modify the instance

- Click `Modify`.
- In the `Settings` section, enter a new password in the `Master password` and `Confirm password` fields.

5. Apply the changes

Internal Only - General

- Scroll to the bottom and choose when to apply the changes. You can apply them immediately or schedule them for the next maintenance window.
- Then, click `Continue` and `Modify DB Instance`.

Note: Changing the master password will reboot the DB instance if you apply the change immediately.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	5.2 Use Unique Passwords Use unique passwords for all enterprise assets. Best practice implementation includes, at a minimum, an 8-character password for accounts using MFA and a 14-character password for accounts not using MFA.	●	●	●
v7	4.4 Use Unique Passwords Where multi-factor authentication is not supported (such as local administrator, root, or service accounts), accounts will use passwords that are unique to that system.		●	●

2.7 Ensure Least Privilege Access (Manual)

Profile Applicability:

- Level 1

Description:

Use the principle of least privilege when granting access to your Amazon Aurora resources. This principle of least privilege (POLP) is a computer security concept where users are given the minimum access levels necessary to complete their job functions.

In Amazon Aurora, this can be implemented at various levels, including AWS IAM for managing AWS resources and within the database for managing database users and roles.

Here is a step-by-step guide for each:

Rationale:

POLP limits the user interaction on the database, and it only gives the database permission to complete the necessary or mandatory task. AWS IAM gives permission for what the entity can and cannot do. Incorporating both POLP and AWS IAM in a database gives limited permission to the user to complete the tasks.

Impact:

Users would need to create a IAM role to implement POLP into their database.

Audit:

Implementing POLP with AWS IAM

1. Sign in to AWS Management Console

- If you do not already have an AWS account, you will need to create one at https://aws.amazon.com.

2. Navigate to IAM Dashboard

- Navigate to the IAM service once logged in to the AWS Management Console.
- You can find this under the `Security, Identity, & Compliance` category.

3. Create a new IAM role or user

- If creating a new IAM role or user, click `Roles` or `Users`.
- Then `Create role` or `Create user`.

4. Attach minimum necessary permissions

- When attaching policies, give only the permissions necessary to perform the intended tasks.
- AWS provides many predefined policies designed following the POLP. You can create a custom policy with precise - permissions if none suits your needs.

Implementing POLP within Amazon Aurora

1. Log into your Aurora Database

   Depending on your Aurora database engine, you can log in through the terminal using a MySQL or PostgreSQL client. You'll need your host endpoint, username, and password to log in.

2. Create a new user

   You can create a new user with the CREATE USER command in SQL.

For example,

```
CREATE USER '<username>'@'<localhost>' IDENTIFIED BY 'password';
```

3. Grant minimal necessary privileges

   After creating the user, you can grant privileges using the GRANT command. The privileges should be as limited as possible for the user to perform their necessary functions.

For example,

```
GRANT SELECT, INSERT ON <mydb.mytbl> TO '<username>'@'<localhost>';
```

4. Regularly review permissions

   It is essential to regularly review and update permissions to make sure they adhere to the principle of least privilege. You can view a user's permissions with the SHOW GRANTS command; for example,

```
SHOW GRANTS FOR '<username>'@'<localhost>';
```

Remediation:

This is important because it reduces and secures any possible threat that an unauthorized user can gain by hacking into the system.

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user’s need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

2.8 Ensure Automatic Backups and Retention Policies are configured (Manual)

Profile Applicability:

- Level 1

Description:

Backups help protect your data from accidental loss or database failure. With Amazon Aurora, you can turn on automatic backups and specify a retention period. The backups include a daily snapshot of the entire DB instance and transaction logs.

Rationale:

The individual logs into their account and chooses their database once selected they can modify the backup settings. To have the database being backed up automatically the individual is encouraged to select from 1 to 35 days. This ensures that the file is being saved automatically and can prevent it from accidental loss. This ensures that the individual can restore their files quickly in the event of a data loss.

Impact:

It would result in having the files protected and being able to retrieve those files in the event of an accidental loss.

Audit:

1. Sign in to AWS Management Console

- If you do not already have an AWS account, you will need to create one at https://aws.amazon.com.

2. Navigate to Amazon RDS Dashboard

- Navigate to the RDS service once logged in to the AWS Management Console.
- You can find this under the `Database` category.

3. Choose your Aurora DB instance

- In the RDS Dashboard, click on `Databases`.
- Then click on the name of your Aurora DB instance.

4. Check or modify the backup settings

- In the `Details` section, find the `Backup` section.

Here, you can see if automatic backups are enabled (the `Backup retention period` is more than 0 days) and when the backup window is.

- To modify these settings, click `Modify`.
- In the `Backup` section of the `Modify DB instance` screen, you can change the `Backup retention period` and the `Backup window`.
- The retention period can be between 1 and 35 days. To disable automatic backups, set the retention period to 0 days.

5. Apply the changes

- Scroll to the bottom and choose when to apply the changes. You can apply them immediately or schedule them for the next maintenance window.
- Then, click `Continue` and `Modify DB Instance`.

Remediation:

This is important because it would allow the user to automatically save their files and instantly have access to their files when needed.

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	11 Data Recovery Establish and maintain data recovery practices sufficient to restore in-scope enterprise assets to a pre-incident and trusted state.			
v7	10 Data Recovery Capabilities Data Recovery Capabilities			

2.9 Ensure Database is not Publicly accessible (Manual)

Profile Applicability:

- Level 1

Description:

AuroraDB databases must not be publicly accessible. This means the database's network configuration should prevent assignment of public IP addresses or exposure to the public internet, ensuring that connections are only permitted from trusted internal networks.

Rationale:

Restricting public access to databases greatly reduces the attack surface for malicious actors. Publicly accessible databases are highly vulnerable to unauthorized login attempts, exploitation of software vulnerabilities and data breaches. Enforcing private access restricts connectivity and enforces the principle of least privilege and network segmentation.

Impact:

If public access is not properly restricted on databases, data stored in the database is at risk of exposure to the internet, increasing the likelihood of data loss and service disruption.

Audit:

1. Sign in to the AWS Management Console where the Aurora database cluster you are auditing resides.
2. Navigate to the Amazon Aurora and RDS Dashboard.

- You can find this under the Database category.

3. Select the DB instance name you wish to audit.

- This opens the details page for your specific Aurora DB instance.

4. Under the Connectivity & security tab, check the value of Publicly accessible:

- If Set to No, the instance is not publicly accessible; no further network verification is needed.
- If Set to Yes, continue with additional steps to fully assess exposure.

5. In the Networking section under Connectivity & security, locate the Subnets for the database:

- Right-click on the subnet link and open it in a new tab for further inspection.

6. With the subnet selected, review the attached Route Table:

- Check for routes with Destination: 0.0.0.0/0 and Target: an Internet Gateway (ID starts with igw-).
- If such a route exists, it enables access to the database from the public internet.

If the database is marked as "Publicly accessible: Yes" and the subnets contain a route to 0.0.0.0/0 via an Internet Gateway, the instance is exposed to the public internet.

Remediation:

Remediation Instructions: Make Aurora DB Instance Non-Publicly Accessible (AWS CLI) Identify all DB instances in the Aurora cluster

1. List your Aurora cluster’s DB instances with:

```
aws rds describe-db-instances --query
"DBInstances[?DBClusterIdentifier=='<your-cluster-
identifier>'].DBInstanceIdentifier"
```

Replace with your actual Aurora cluster identifier.

2. For each DB instance, run the following command:

```
aws rds modify-db-instance --db-instance-identifier <db-instance-identifier>
--no-publicly-accessible --apply-immediately
```

Replace with the name of your DB instance. The --apply-immediately flag ensures the change is applied right away.

3. Verify changes - confirm that “Publicly Accessible” is now set to “No” for each DB instance:

```
aws rds describe-db-instances --db-instance-identifier <db-instance-
identifier> --query "DBInstances[0].PubliclyAccessible"
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.12 Segment Data Processing and Storage Based on Sensitivity Segment data processing and storage based on the sensitivity of the data. Do not process sensitive data on enterprise assets intended for lower sensitivity data.		●	●

Controls Version	Control	IG 1	IG 2	IG 3
v8	4 Secure Configuration of Enterprise Assets and Software Establish and maintain the secure configuration of enterprise assets (end-user devices, including portable and mobile; network devices; non-computing/IoT devices; and servers) and software (operating systems and applications).			
v8	12.2 Establish and Maintain a Secure Network Architecture Establish and maintain a secure network architecture. A secure network architecture must address segmentation, least privilege, and availability, at a minimum.		●	●

2.10 Ensure Database has IAM Auth is Enabled (Manual)

Profile Applicability:

- Level 2

Description:

AuroraDB clusters should be configured to leverage AWS IAM authentication for database connections. This ensures that users authenticate using temporary IAM-based tokens rather than static long-lived passwords.

Rationale:

Enabling IAM authentication for AuroraDB centralizes and strengthens access control by integrating database authentication with broader AWS IAM identity management. This approach eliminates the risks associated with hard-coded credentials, reduces administrative overhead for password rotation, and allows precise access management using IAM identities and policies.

Impact:

With the usage of IAM database authentication instead of static passwords, AuroraDB clusters are protected against credential leakage and weak password practices, making unauthorized access significantly more difficult. This reduces the attack surface, supports audit and compliance, and ensures that database access is tightly aligned with enterprise identity governance and cloud security standards.

Audit:

1. List AuroraDB clusters and check IAM authentication status:

```
aws rds describe-db-clusters \
  --query
"DBClusters[*].{DBClusterIdentifier:DBClusterIdentifier,IAMDatabaseAuthentica
tionEnabled:IAMDatabaseAuthenticationEnabled}" \
  --output table
```

2. List database users enabled for IAM authentication (Aurora MySQL):

- For MySQL, connect to the cluster and run below command to ensure that required database users exist for IAM token logins.

```
SELECT user, plugin FROM mysql.user WHERE plugin='AWSAuthenticationPlugin';
```

- For Postgres, connect to the cluster and run below command to verify that necessary users are granted the rds_iam role.

```
SELECT r.rolname
FROM pg_roles AS r
JOIN pg_auth_members AS m ON r.oid = m.member
JOIN pg_roles AS g ON m.roleid = g.oid
WHERE g.rolname = 'rds_iam';
```

The cluster must have IAM database authentication enabled and users intended for IAM authentication must exist in the database engine with appropriate privileges.

Remediation:

1. Enable IAM Database Authentication on the Aurora Cluster

- Use the AWS CLI to enable IAM authentication for an existing Aurora cluster:

```
aws rds modify-db-cluster \
  --db-cluster-identifier <your-cluster-name> \
  --enable-iam-database-authentication \
  --apply-immediately
```

2. Create Local Database Users Configured for IAM Authentication

- For Aurora MySQL: Connect to the database and create or alter users as follows:

```
CREATE USER 'jane_doe' IDENTIFIED WITH AWSAuthenticationPlugin as 'RDS';
```

or

ALTER USER 'jane_doe' IDENTIFIED WITH AWSAuthenticationPlugin as 'RDS';

- For Aurora PostgreSQL:

Grant the rds_iam role to eligible users:

```
GRANT rds_iam TO jane_doe;
```

3. Grant Necessary Privileges to the Database Users

- Assign required permissions and roles to the users within your DB engine via standard SQL GRANT commands.

4. Ensure IAM Users/Roles Have Required AWS Permissions

- The IAM principal that connects needs the following AWS permission in their policy:

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "rds-db:connect"
            ],
            "Resource": [
                "arn:aws:rds-db:region:account-
id:dbuser:DbClusterResourceId/db-user-name"
            ]
        }
    ]
}
```

- region is the AWS Region for the DB cluster
- account-id is the AWS account number for the DB cluster.
- DbClusterResourceId is the identifier for the DB cluster. This identifier is unique to an AWS Region and never changes. To find a DB cluster resource ID in the AWS Management Console for Amazon Aurora, choose the DB cluster to see its details. Then choose the Configuration tab. The Resource ID is shown in the Configuration section.

5. Test the Configuration

- Use the AWS CLI to generate an authentication token:

```
aws rds generate-db-auth-token \
  --hostname <cluster-endpoint> \
  --port 3306 \
  --region <region> \
  --username <db_user>
```

- Use the generated token to authenticate to the database, confirming successful login.

For mysql:

```
mysql --host= <cluster-endpoint> \
  --port=3306 \
  --ssl-mode=REQUIRED \
  --enable-cleartext-plugin \
  --user= <db_user_name> \
  --password= '<generated_db_auth_token>'
```

For postgres:

```
psql "host=<cluster-endpoint> port=5432 dbname=<database_name>
user=<>db_user_name password='<generated_db_auth_token>' sslmode=require"
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	6.7 Centralize Access Control Centralize access control for all enterprise assets through a directory service or SSO provider, where supported.		●	●

2.11 Ensure Database has delete protection enabled (Manual)

Profile Applicability:

- Level 1

Description:

Ensure that delete protection is enabled on database instances to prevent accidental or unauthorized deletion. This setting safeguards critical databases by requiring explicit disabling of delete protection before deletion, reducing the risk of data loss through human error or malicious activity.

Rationale:

Delete protection provides a safeguard against inadvertent or malicious deletion of critical databases. By requiring deliberate action to disable deletion protection, organizations mitigate risks associated with accidental data deletion and enhance the overall resilience of their data storage platform.

Impact:

Failure to enable delete protection increases the risk of irreversible data loss, potential service disruption, and operational downtime.

Audit:

Run the following command to check if deletion protection is enabled on your Aurora DB cluster(s):

```
aws rds describe-db-clusters \
  --query
"DBClusters[*].{DBClusterIdentifier:DBClusterIdentifier,DeletionProtection:De
letionProtection}" \
  --output table
```

- Review each cluster's DeletionProtection status.
- Clusters marked as true have deletion protection enabled.
- Identify any clusters with deletion protection disabled for remediation.

Remediation:

1. To enable deletion protection on an existing Aurora DB cluster:

```
aws rds modify-db-cluster \
  --db-cluster-identifier <your-cluster-name> \
  --deletion-protection \
  --apply-immediately
```

- Replace with your Aurora cluster ID.

- This change is applied immediately without downtime.
- Once enabled, the cluster cannot be deleted without first disabling deletion protection.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4 Secure Configuration of Enterprise Assets and Software Establish and maintain the secure configuration of enterprise assets (end-user devices, including portable and mobile; network devices; non-computing/IoT devices; and servers) and software (operating systems and applications).			

3 Amazon RDS

Amazon Relational Database Service (Amazon RDS) is a fully managed relational database service provided by Amazon Web Services (AWS). It simplifies the setup, operation, and scaling of relational databases, making it easier for developers to deploy, manage, and scale database instances without the overhead of traditional database administration tasks. Amazon RDS supports several popular relational database engines, including MySQL, PostgreSQL, MariaDB, Oracle Database, and Microsoft SQL Server.

3.1 Ensure to Choose the Appropriate Database Engine (Manual)

Profile Applicability:

- Level 1

Description:

Rationale:

Audit:

1. Evaluate Your Requirements

- Understand your application's specific requirements, such as performance, scalability, data volume, and compatibility with existing systems.
- Consider factors like data structure, workload type (OLTP or OLAP), and specific features required by your application.

2. Research Available Database Engines

- Familiarize yourself with the available database engine options supported by Amazon RDS.
- Research each database engine's capabilities, features, performance characteristics, and licensing models.

3. Compare Features and Compatibility

- Compare the features and capabilities of each database engine with your application's requirements.
- Evaluate data types, indexing options, query optimization, high availability, replication, and backup and restore capabilities.
- Consider compatibility with your existing applications, frameworks, and tools.

4. Evaluate Performance and Scalability

- Consider the performance characteristics of each database engine, including throughput, latency, and concurrency capabilities.
- Evaluate scalability options, such as horizontal scaling or vertical scaling.
- Analyze benchmarks, customer reviews, and case studies to gain insights into the performance of each database engine.

5. Consider Managed Database Services

- Assess the benefits of Amazon RDS managed database services, such as Amazon Aurora, which offers high performance, scalability, and built-in fault tolerance.

- Evaluate the additional features and optimizations Amazon Aurora provides compared to traditional database engines.

6. Evaluate Licensing and Costs

- Consider the licensing models and costs associated with each database engine, including license fees and support costs.
- Evaluate the pricing structure of the database engines in terms of instance types, storage, data transfer, and other factors.

7. Determine Vendor Support

- Evaluate the level of support the database engine vendors provide, including documentation, forums, community support, and enterprise support options.
- Consider the vendor's reputation, track record, and commitment to security and compliance.

8. Make an Informed Decision

- Select the database engine that best aligns with your application requirements, performance needs, scalability goals, compatibility, and budget based on your evaluation and analysis.
- Consider long-term considerations such as potential future growth, flexibility, and ease of migration to other database engines if needed.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	2.2 Ensure Authorized Software is Currently Supported Ensure that only currently supported software is designated as authorized in the software inventory for enterprise assets. If software is unsupported, yet necessary for the fulfillment of the enterprise's mission, document an exception detailing mitigating controls and residual risk acceptance. For any unsupported software without an exception documentation, designate as unauthorized. Review the software list to verify software support at least monthly, or more frequently.	●	●	●
v7	2.2 Ensure Software is Supported by Vendor Ensure that only software applications or operating systems currently supported by the software's vendor are added to the organization's authorized software inventory. Unsupported software should be tagged as unsupported in the inventory system.	●	●	●

3.2 Ensure to Create The Appropriate Deployment Configuration (Manual)

Profile Applicability:

- Level 1

Description:

This control is important and helps businesses to choose from two deployment options, either single or multi-AZ deployment. Depending on the business factor and their security needs the organization is then encouraged to make a decision that would benefit them.

Rationale:

Audit:

1. Evaluate High Availability Requirements

- Assess the high availability needs of your application. Consider factors such as uptime requirements, business continuity, and disaster recovery.
- Determine if your application requires automatic failover, data durability, and minimal downtime during maintenance or outages.

2. Understand RDS Deployment Options

- Familiarize yourself with the deployment options available on Amazon RDS. These include single-AZ (Availability Zone) and multi-AZ deployments.
- Understand the differences between these options regarding availability, durability, and cost.

3. Single-AZ Deployment

- Consider a single-AZ deployment if high availability is not a critical requirement for your application.
- In a single-AZ deployment, your database runs in a single Availability Zone, providing basic durability and availability.

4. Multi-AZ Deployment

- Choose a multi-AZ deployment if high availability and automatic failover are crucial for your application.
- In a multi-AZ deployment, your database is replicated synchronously to a standby replica in a different Availability Zone, providing automatic failover in the event of a primary database failure.

- Multi-AZ deployments provide enhanced availability and durability, ensuring minimal downtime during maintenance or outages.

5. Evaluate Cost Implications

- Consider the cost implications of your deployment choice.
- Multi-AZ deployments incur additional costs than single-AZ deployments due to the replication and standby infrastructure.

6. Make a Deployment Decision

- Based on your evaluation of high availability requirements, consider the trade-offs between single-AZ and multi-AZ deployments.
- Choose the appropriate deployment configuration that meets your application's availability, durability, and cost requirements.

7. Configure RDS Deployment

- Once you have determined the deployment configuration, go to the Amazon RDS console.
- Create a new database instance or modify an existing one to match your chosen deployment configuration.
- Follow the prompts and configure the deployment options, selecting the desired AZs and replication settings.
- Adjust other configuration settings, such as instance type, storage, and backup options, based on your application's needs.

8. Test and Monitor

- After the deployment is set up, thoroughly test your application's functionality and performance.
- Monitor the RDS instance and replication status using the Amazon RDS console or CloudWatch metrics.
- Ensure that the database failover and automatic maintenance operations work as expected.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

Internal Only - General

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	0.0 Explicitly Not Mapped Explicitly Not Mapped			
v7	0.0 Explicitly Not Mapped Explicitly Not Mapped			

3.3 Ensure to Create a Virtual Private Cloud (VPC) (Manual)

Profile Applicability:

- Level 1

Description:

Setting up a Virtual Private Cloud (VPC) protects the private network that has been established from any external networks from interfering. It allows internal networks to communicate with one another with the network that has been established.

Rationale:

Impact:

Builds a strong connection between internal networks, and the internet, and it secures your data from getting into the hand of an unauthorized party.

Audit:

1. Sign in to the AWS Management Console

- Sign in to the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon VPC Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/vpc/.

3. Create a VPC

- In the Amazon VPC console, click `Your VPCs` in the left-side menu.
- Click on `Create VPC` to begin creating a new VPC.
- Provide a name and the desired IPv4 CIDR block for your VPC.
- Configure additional settings, such as IPv6 CIDR block, tenancy, and DNS resolution.
- Click `Create` to create the VPC.

4. Create Subnets

- In the Amazon VPC console, click `Subnets` in the left-side menu.
- Click on `Create subnet` to create a subnet within the VPC.
- Select the VPC you created in the previous step.
- Provide a name, choose an availability zone, and specify the IPv4 CIDR block for the subnet.
- Configure additional settings, such as IPv6 CIDR block and availability zone.

- Click `Create` to create the subnet.

5. Configure Route Tables

- In the Amazon VPC console, click on `Route Tables` in the left-side menu.
- Click on `Create route table` to create a new route table.
- Provide a name for the route table and select the VPC you created earlier.
- Click `Create` to create the route table.
- Associate the route table with the desired subnets by selecting the route table and clicking on the `Subnet associations` tab.
- Click `Edit subnet associations` and select the desired subnets to associate them with the route table.

6. Configure Security Groups

- In the Amazon VPC console, click `Security Groups` in the left-side menu.
- Click on `Create security group` to create a new security group.
- Provide a name and description for the security group.
- Select the VPC you created earlier.
- Configure inbound and outbound rules to control network traffic to and from your RDS instances.
- Click `Create` to create the security group.

7. Configure Network Access Control Lists (ACLs)

- In the Amazon VPC console, click on `Network ACLs` in the left-side menu.
- Click on `Create network ACL` to create a new network ACL.
- Provide a name for the network ACL and select the VPC you created earlier.
- Configure inbound and outbound rules to allow or deny specific types of traffic.
- Associate the network ACL with the desired subnets by selecting the network ACL and clicking on the `Subnet associations` tab.
- Click `Edit subnet associations` and select the desired subnets to associate them with the network ACL.

8. Use the VPC with Amazon RDS

- Select the appropriate VPC, subnets, and security groups when creating an RDS instance.
- Configure the database instance with the desired network and security settings within the chosen VPC.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	12.2 Establish and Maintain a Secure Network Architecture Establish and maintain a secure network architecture. A secure network architecture must address segmentation, least privilege, and availability, at a minimum.		●	●
v7	11.7 Manage Network Infrastructure Through a Dedicated Network Manage the network infrastructure across network connections that are separated from the business use of that network, relying on separate VLANs or, preferably, on entirely different physical connectivity for management sessions for network devices.		●	●

3.4 Ensure to Configure Security Groups (Manual)

Profile Applicability:

- Level 1

Description:

Configuring security groups benefits the user because it helps manage networks within the database and gives only certain permission for traffic that leaves and enters the database.

Rationale:

Impact:

Allows certain users to access the instance and it only allows them to work within that network.

Audit:

1. Sign into the AWS Management Console

- Sign into the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon RDS Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/rds/.

3. Select the RDS Instance

- Choose the Amazon RDS instance for which you want to configure security groups. Click on the instance name to access its details page.

4. Navigate to the `Connectivity & Security` Section

- In the instance details page, navigate to the `Connectivity & Security` or "Security" section.

5. View and Modify Existing Security Groups

- Under the `Security` section, you will see the existing security groups associated with the RDS instance.
- Take note of the existing security groups and their inbound and outbound rules.

6. Create a New Security Group

- If you need to create a new security group for the RDS instance
- Click the `Create New Security Group` button.
- Provide a name and description for the new security group.
- Configure the inbound and outbound rules to control network traffic to and from the RDS instance.
- Click "Create" to create the new security group.

7. Modify Security Group Rules

- To modify the rules of an existing security group, click on the security group name or the `Modify` button next to it.
- You can add, edit, or delete inbound and outbound rules on the security group details page.
- Specify each rule's source IP addresses, port ranges, and protocols.
- Click `Save` or `Apply Changes` to update the security group rules.

8. Associate Security Groups

- To associate a security group with the RDS instance, navigate to the `Connectivity & Security` or `Security` section of the instance details page.
- Click `Modify` next to the `VPC security groups` option.
- Select the desired security groups from the list.
- Click `Save` or `Apply Changes` to associate them with the RDS instance.

9. Verify and Test Security Group Configuration

- Review the security group settings to match your network access requirements.
- Test the connectivity to the RDS instance by attempting to access it from authorized IP addresses or applications.

10. Monitor and Update Security Groups

- Regularly monitor the network traffic and access patterns to your RDS instance.
- Update the security group rules as needed to reflect changes in your network access requirements.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user’s need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

3.5 Enable Encryption at Rest (Manual)

Profile Applicability:

- Level 1

Description:

This helps ensure that the data is kept secure and protected when at rest. The user must choose from two key options which then determine when the data is encrypted at rest.

Rationale:

Impact:

If an unauthorized user steals the data, it would be unreadable for them because a key would be required to decrypt the message into plaintext.

Audit:

1. Sign into the AWS Management Console

- Sign into the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon RDS Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/rds/.

3. Select the RDS Instance

- Choose the Amazon RDS instance you want to enable encryption at rest.
- Click on the instance name to access its details page.
- In the instance details page, navigate to the `Configuration` or `Encryption & Security` section.

4. Enable Encryption at Rest

- Under the `Encryption` or `Encryption at Rest` section
- Click on the `Modify` button or the `Enable` option to enable encryption at rest.
- Choose the desired encryption option, either `AWS managed keys (default)` or `Customer managed keys using AWS Key Management Service (KMS)`.
- If selecting `AWS managed keys`, you do not need to perform additional configuration steps.
- If selecting `Customer managed keys` you will need to specify the KMS key you want to use for encryption.

Internal Only - General

- Select the appropriate KMS key or create a new KMS key if necessary.
- Click `Continue` or `Save` to apply the changes.

5. Monitor the Encryption Status

- After enabling encryption at rest, monitor the encryption status of your RDS instance.
- In the RDS console, check the `Encryption` or `Encryption at Rest` section to ensure that encryption is enabled, and the status is `In Progress` or `Enabled`.

6. Verify Encryption at Rest

- Validate that data at rest is encrypted by accessing the RDS instance and examining the database files.
- Confirm that the data is stored in an encrypted format.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.11 Encrypt Sensitive Data at Rest Encrypt sensitive data at rest on servers, applications, and databases containing sensitive data. Storage-layer encryption, also known as server-side encryption, meets the minimum requirement of this Safeguard. Additional encryption methods may include application-layer encryption, also known as client-side encryption, where access to the data storage device(s) does not permit access to the plain-text data.		●	●
v7	14.8 Encrypt Sensitive Information at Rest Encrypt all sensitive information at rest using a tool that requires a secondary authentication mechanism not integrated into the operating system, in order to access the information.			●

3.6 Enable Encryption in Transit (Manual)

Profile Applicability:

- Level 1

Description:

Amazon Relational Database uses SSL/TLS to encrypt data during transit. To secure your data in transit the individual should identify their client application and what is supported by SSL/TLS to configure it correctly.

Rationale:

Audit:

1. Sign into the AWS Management Console

- Sign into the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon RDS Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/rds/.

3. Select the RDS Instance

- Choose the Amazon RDS instance you want to implement encryption in transit.
- Click on the instance name to access its details page.
- In the instance details page, navigate to the `Configuration` or `Encryption & Security` section.

4. Enable SSL/TLS

- Under the `Connectivity` or `Encryption in Transit` section
- Click the `Modify` or `Edit` option to enable SSL/TLS encryption.
- Select the option to enable SSL/TLS encryption.
- Choose the SSL/TLS certificate authority (CA) certificate option that best suits your needs:
  - If you have an existing certificate, select `Use a certificate from ACM (AWS Certificate Manager)` or `Use a certificate from AWS Secrets Manager`.
  - If you do not have a certificate, select `Generate a new certificate`.

    Click `Continue` or `Save` to apply the changes.

5. Verify SSL/TLS Encryption

- After enabling SSL/TLS encryption, monitor the encryption status of your RDS instance.
- In the RDS console, check the `Connectivity` or "Encryption in Transit" section to ensure that SSL/TLS encryption is enabled, and the status is "In Progress" or "Enabled."

6. Test SSL/TLS Encryption

- Connect to your RDS instance using a database client or application that supports SSL/TLS encryption.
- Configure the client or application to use SSL/TLS encryption by specifying the SSL/TLS certificate details.
- Verify that the connection is established successfully with SSL/TLS encryption.

7. Monitor and Manage SSL/TLS Certificates

- Regularly monitor the SSL/TLS certificates associated with your RDS instances.
- Manage certificate expiration and renewal to ensure uninterrupted SSL/TLS encryption.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.10 Encrypt Sensitive Data in Transit Encrypt sensitive data in transit. Example implementations can include: Transport Layer Security (TLS) and Open Secure Shell (OpenSSH).		●	●
v7	14.4 Encrypt All Sensitive Information in Transit Encrypt all sensitive information in transit.		●	●

Internal Only - General

3.7 Ensure to Implement Access Control and Authentication (Manual)

Profile Applicability:

- Level 1

Description:

Users should select whether they like to enable authentication. If they want to authenticate a password would be required, which would only allow the authorized person to access the database. Defining access control allows specific workers in a business access to the database.

Rationale:

Audit:

1. Sign into the AWS Management Console

- Sign into the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon RDS Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/rds/.

3. Select the RDS Instance

- Choose the Amazon RDS instance you want to implement access control and authentication.
- Click on the instance name to access its details page.
- In the instance details page, navigate to the `Configuration` or `Connectivity & Security` section.

4. Enable IAM Database Authentication

- Under the `Connectivity` or `Connectivity & Security` section.
- Click the `Modify` or `Edit` option to enable IAM Database Authentication.
- Select the option to enable IAM Database Authentication.
- Click `Continue` or `Save` to apply the changes.

5. Create and Configure IAM Database Users

- Click `Users` in the left-side menu in the Amazon RDS console.
- Click `Create database user` to create a new IAM database user.

- Provide a username and select the IAM role or IAM user that will be associated with the database user.
- Configure the authentication type, either `Password-based` or `IAM authentication`.
- Set the desired password or leave it blank for IAM authentication.
- Configure the database user's privileges and permissions based on your application's requirements.
- Click `Create` to create the IAM database user.

6. Configure Database User Privileges

- Click `Users` in the left-side menu in the Amazon RDS console.
- Select the database user you created in the previous step.
- Click on `Modify` to modify the user's settings and permissions.
- Configure the user's access privileges, including database access, object permissions, and privileges.
- Click `Save` or `Apply Changes` to update the user's privileges.

7. Test Access and Authentication

- Test the access and authentication by connecting to the RDS instance using the IAM database user's credentials or IAM role.
- Verify that the authentication and access control mechanisms are functioning correctly.

8. Monitor and Manage IAM Database Users

- Regularly monitor and review the IAM database users and their access privileges.
- Adjust user privileges as needed based on changes in your application requirements.
- Remove or disable database users when they are no longer needed.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user’s need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

3.8 Ensure to Regularly Patch Systems (Manual)

Profile Applicability:

- Level 1

Description:

Rationale:

Impact:

Helps the organization reduce their security risk by regularly updating and patching their database and database engine. Regularly updating and scanning for any weaknesses in the company can bring up possible vulnerabilities that could have led to potential cyber-attack.

Audit:

1. Stay Informed about Database Engine Updates

- Stay up-to-date with the latest information regarding database engine updates and patches provided by the respective database engine vendors (e.g., MySQL, PostgreSQL, Oracle, SQL Server).
- Subscribe to release announcements, security bulletins, and updates from the database engine vendor or AWS.

2. Review the Database Engine Documentation

- Refer to the documentation provided by the database engine vendor to understand the recommended patching and update processes specific to the database engine you use on Amazon RDS.
- Review the vendor's guidelines and best practices for applying updates and patches.

3. Plan for Maintenance Windows

- Determine regular maintenance windows during which you can schedule updates and patches for your RDS instances.
- Coordinate with your team to ensure minimal disruption to your applications and users during the maintenance window.

4. Enable Automated Minor Version Upgrades

- In the Amazon RDS console, select the RDS instance you want to enable automated upgrades.
- Under the `Maintenance & backups` or `Maintenance` section.
- Enable the `Auto minor version upgrade` option.

Internal Only - General

- This allows Amazon RDS to automatically apply eligible minor version upgrades to your RDS instances during the maintenance window.

5. Monitor Available Updates

- Regularly monitor the `Pending Maintenance` section in the Amazon RDS console for any updates or patches for your RDS instances.
- Pay attention to notifications and alerts from AWS about pending updates.

6. Schedule Updates and Patches

- Review the available updates and patches and their associated release notes and security advisories.
- Please select the appropriate updates based on their impact, criticality, and compatibility with your applications.
- Schedule the updates and patches to be applied during the designated maintenance window.

7. Apply Updates and Patches

- During the scheduled maintenance window, Amazon RDS automatically applies the eligible updates and patches to your RDS instances.
- Monitor the progress of the updates and patches through the Amazon RDS console.

8. Test and Validate

- After the updates and patches are applied, thoroughly test your applications to ensure they function as expected.
- Validate the database performance, data integrity, and application functionality.

9. Monitor for Issues

- Monitor the performance and behavior of your RDS instances after the updates and patches are applied.
- Keep an eye out for any issues or anomalies and address them promptly.

10. Review and Document

- Review the release notes and documentation of the applied updates and patches to understand the changes and improvements they bring.
- Document the update and patching process, including the applied versions, dates, and any issues encountered.

Internal Only - General

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	7 Continuous Vulnerability Management Develop a plan to continuously assess and track vulnerabilities on all enterprise assets within the enterprise's infrastructure, in order to remediate, and minimize, the window of opportunity for attackers. Monitor public and private industry sources for new threat and vulnerability information.			
v7	3 Continuous Vulnerability Management Continuous Vulnerability Management			

Internal Only - General

3.9 Ensure Monitoring and Logging is Enabled (Manual)

Profile Applicability:

- Level 1

Description:

Rationale:

Impact:

If the individual is not monitoring and logging their activity it allows the attacker to attack the system and extract or destroy data.

Audit:

1. Sign into the AWS Management Console

- Sign into the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon RDS Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/rds/.

3. Select the RDS Instance

- Choose the Amazon RDS instance you want to enable monitoring and logging.
- Click on the instance name to access its details page.
- In the instance details page, navigate to the `Configuration` or `Monitoring & Logs` section.

4. Enable Enhanced Monitoring

- Under the `Monitoring` section.
- Click on the `Modify` button or `Edit` option to enable enhanced monitoring.
- Choose the desired monitoring granularity (1-minute or 5-minute intervals) and the retention period for the monitoring data.
- Click `Continue` or `Save` to apply the changes.

5. Enable Enhanced Logging

- Under the `Logs` or `Monitoring & Logs` section.
- Click on the `Modify` button or `Edit` option to enable enhanced logging.

Internal Only - General

- Choose the desired log types to enable, such as general, error, slow query, or audit logs.
- Configure the log file retention period based on your needs.
- Select the destination for the logs, such as Amazon CloudWatch Logs or an Amazon S3 bucket.
- Configure the log format and other settings if applicable.
- Click `Continue` or `Save` to apply the changes.

6. Configure CloudWatch Alarms (Optional)

- Click `Alarms` in the Amazon RDS console menu.
- Click `Create alarm` to create a CloudWatch alarm to monitor specific metrics or log events.
- Configure the alarm threshold, actions to take when the threshold is breached, and notification settings.
- Click `Create` to create the CloudWatch alarm.

7. Monitor and Analyze the Metrics and Logs

- Monitor the metrics and logs in the Amazon RDS console or by accessing CloudWatch or the configured log destination.
- Use the metrics and logs to gain insights into your RDS instance's performance, behavior, and issues.
- Analyze the metrics and logs to identify areas for optimization, troubleshoot problems, or detect anomalies.

8. Set Up Automated Actions (Optional)

- In the Amazon RDS console, click on `Event subscriptions` in the left-side menu.
- Click `Create event subscription` to set up automated actions based on specific events or log entries.
- Configure the event pattern, target actions, and notification settings.
- Click `Create` to create the event subscription.

9. Monitor and Respond to Alerts

- Monitor the CloudWatch alarms and event notifications for any alerts or triggers based on the configured thresholds.
- Respond to alerts promptly by investigating and resolving the underlying issues or taking appropriate actions.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	8 Audit Log Management Collect, alert, review, and retain audit logs of events that could help detect, understand, or recover from an attack.			
v7	6 Maintenance, Monitoring and Analysis of Audit Logs Maintenance, Monitoring and Analysis of Audit Logs			

Internal Only - General

3.10 Ensure to Enable Backup and Recovery (Manual)

Profile Applicability:

- Level 1

Description:

The individual logs into their AWS account and chooses their Amazon relational database that they want to backup. To have the database being backed up automatically the individual is encouraged to enable backup. This ensures that the file is being saved automatically and can prevent it from accidental loss. This ensures that the individual can restore their files quickly in the event of a data loss.

Rationale:

Impact:

It would result in having the files protected and being able to retrieve those files in the event of an accidental loss.

Audit:

1. Sign into the AWS Management Console

- Sign into the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon RDS Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/rds/.

3. Select the RDS Instance

- Choose the Amazon RDS instance you want to implement backup and recovery.
- Click on the instance name to access its details page.
- In the instance details page, navigate to the `Backup & Restore` or `Backup` section.

4. Configure Automated Backups

- Under the `Backup` section.
- Click the `Modify` or `Edit` option to configure automated backups.
- Enable automated backups by selecting the desired backup retention period.
- Specify the preferred backup window during which automated backups can occur.

- Choose whether to enable Multi-AZ backups for enhanced durability and availability.
- Click `Continue` or `Save` to apply the changes.

5. Restore from Backups

- In the Amazon RDS console, click on `Snapshots` or `Instances` in the left-side menu.
- Select the snapshot or instance from which you want to perform a restore.
- Click `Restore snapshot` or `Restore to point in time` to initiate restoration.
- Configure the parameters for the restored instance, such as instance identifier, instance class, storage type, and VPC settings.
- Specify the desired option for creating a new DB instance or restoring to an existing DB instance.
- Configure additional settings, such as enabling Multi-AZ deployment or enabling encryption.
- Click "Restore" or "Create" to initiate the restore process.

6. Test and Validate the Restored Instance

- After completing the restore process, test the restored RDS instance to ensure it functions as expected.
- Verify the data, configuration, and connectivity of the restored instance.

7. Monitor and Manage Backups

- Regularly monitor the status and health of your automated backups and manual snapshots.
- Review the backup retention policy and adjust it to align with your business requirements.
- Manage and delete older backups or snapshots to free up storage and reduce costs.

8. Perform Point-in-Time Recovery (Optional)

- In the Amazon RDS console, click on "Snapshots" or `Instances` in the left-side menu.
- Select the instance for which you want to perform point-in-time recovery.
- Click on `Restore to point in time` to initiate the point-in-time recovery process.
- Specify the desired timestamp or time range to restore to.
- Configure the parameters for the restored instance, similar to the restore from the backup process.
- Click `Restore` or "Create" to initiate the point-in-time recovery process.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	11 Data Recovery Establish and maintain data recovery practices sufficient to restore in-scope enterprise assets to a pre-incident and trusted state.			
v7	10 Data Recovery Capabilities Data Recovery Capabilities			

Internal Only - General

3.11 Ensure to Regularly Review Security Configuration (Manual)

Profile Applicability:

- Level 1

Description:

This helps by reviewing the database factors from database engine, review instance details, security networks, encryption settings, audit logging, and authentication. By updating or removing a few things from these lists it helps tighten security and ensures that the users do not have excessive permissions.

Rationale:

Impact:

Updating the system and being updated with security configurations keeps everything secure and prevents it from an attack.

Audit:

1. Sign into the AWS Management Console

- Sign into the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon RDS Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/rds/.

3. Select the RDS Instance

- Choose the Amazon RDS instance you want to review the security configuration.
- Click on the instance name to access its details page.

4. Review the Database Engine Documentation

- Refer to the documentation provided by the database engine vendor (e.g., MySQL, PostgreSQL, Oracle, SQL Server) to understand the security best practices and configuration options specific to the database engine you use on Amazon RDS.
- Review the vendor's guidelines for securing the database engine and associated components.

5. Review the Instance Details

- In the instance details page, review the configuration settings related to security.
- Security group associations: Ensure the appropriate security groups are assigned to the RDS instance to control inbound and outbound traffic.
- IAM database authentication: Verify if IAM database authentication is enabled for enhanced security.
- Encryption at rest: Confirm if encryption at rest is enabled using either AWS-managed keys or customer-managed keys.
- Encryption in transit: Check if SSL/TLS encryption is enabled for secure data transmission.

  Backup and retention: Review the automated backup settings and retention period to ensure data recovery capability.

6. Review Database User Privileges

- Click `Users` in the Amazon RDS console menu.
- Review the privileges assigned to database users.
- Ensure that the least privileged access is implemented, granting only necessary privileges to each user or role.

7. Review Audit and Logging Configuration

- In the Amazon RDS console, navigate to the `Configuration` or `Monitoring & Logs` section.
- Review the settings related to database audit logging and logging.
- Ensure appropriate logs are enabled and configured to capture necessary information for security analysis and monitoring.

8. Review Network Security

- In the Amazon RDS console, navigate to the `Connectivity & Security` or `Security` section.
- Review the network security settings, including the associated security groups and their rules.
- Verify that only necessary ports are open, and access is restricted to trusted sources.

9. Review and Address Security Recommendations

- Periodically review the security recommendations provided by AWS through the Amazon RDS console or the AWS Trusted Advisor service.
- Address any security recommendations promptly to ensure a secure configuration.

10. Document and Update

- Document the security configuration settings and any changes made during the review process.
- Maintain an up-to-date inventory of the security controls and configurations implemented for your RDS instances.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	5 Account Management Use processes and tools to assign and manage authorization to credentials for user accounts, including administrator accounts, as well as service accounts, to enterprise assets and software.			
v7	5 Secure Configuration for Hardware and Software on Mobile Devices, Laptops, Workstations and Servers Secure Configuration for Hardware and Software on Mobile Devices, Laptops, Workstations and Servers			

3.12 Ensure Database is not Publicly accessible (Manual)

Profile Applicability:

- Level 1

Description:

RDS databases must not be publicly accessible. This means the database's network configuration should prevent assignment of public IP addresses or exposure to the public internet, ensuring that connections are only permitted from trusted internal networks.

Rationale:

Restricting public access to databases greatly reduces the attack surface for malicious actors. Publicly accessible databases are highly vulnerable to unauthorized login attempts, exploitation of software vulnerabilities and data breaches. Enforcing private access restricts connectivity and enforces the principle of least privilege and network segmentation.

Impact:

If public access is not properly restricted on databases, data stored in the database is at risk of exposure to the internet, increasing the likelihood of data loss and service disruption.

Audit:

1. Sign in to the AWS Management Console where the RDS database cluster you are auditing resides.
2. Navigate to the Amazon Aurora and RDS Dashboard.

- You can find this under the Database category.

3. Select the DB instance name you wish to audit.

- This opens the details page for your specific RDS DB instance.

4. Under the Connectivity & security tab, check the value of Publicly accessible:

- If Set to No, the instance is not publicly accessible; no further network verification is needed.
- If Set to Yes, continue with additional steps to fully assess exposure.

5. In the Networking section under Connectivity & security, locate the Subnets for the database:

- Right-click on the subnet link and open it in a new tab for further inspection.

6. With the subnet selected, review the attached Route Table:

- Check for routes with Destination: 0.0.0.0/0 and Target: an Internet Gateway (ID starts with igw-).
- If such a route exists, it enables access to the database from the public internet.

If the database is marked as "Publicly accessible: Yes" and the subnets contain a route to 0.0.0.0/0 via an Internet Gateway, the instance is exposed to the public internet.

Remediation:

1. List your RDS cluster's DB instances with:

```
aws rds describe-db-instances --query
"DBInstances[?DBClusterIdentifier=='<your-cluster-
identifier>'].DBInstanceIdentifier"
```

- Replace with your actual RDS cluster identifier.

2. For each DB instance, run the following command:

```
aws rds modify-db-instance --db-instance-identifier <db-instance-identifier>
--no-publicly-accessible --apply-immediately
```

- Replace with the name of your DB instance. The --apply-immediately flag ensures the change is applied right away.

3. Verify changes - confirm that "Publicly Accessible" is now set to "No" for each DB instance:

```
aws rds describe-db-instances --db-instance-identifier <db-instance-
identifier> --query "DBInstances[0].PubliclyAccessible"
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.12 Segment Data Processing and Storage Based on Sensitivity Segment data processing and storage based on the sensitivity of the data. Do not process sensitive data on enterprise assets intended for lower sensitivity data.		●	●

Controls Version	Control	IG 1	IG 2	IG 3
v8	4 Secure Configuration of Enterprise Assets and Software Establish and maintain the secure configuration of enterprise assets (end-user devices, including portable and mobile; network devices; non-computing/IoT devices; and servers) and software (operating systems and applications).			
v8	12.2 Establish and Maintain a Secure Network Architecture Establish and maintain a secure network architecture. A secure network architecture must address segmentation, least privilege, and availability, at a minimum.		●	●

3.13 Ensure Database has IAM Auth is Enabled (Manual)

Profile Applicability:

- Level 2

Description:

RDS clusters should be configured to leverage AWS IAM authentication for database connections. This ensures that users authenticate using temporary IAM-based tokens rather than static long-lived passwords.

Rationale:

Enabling IAM authentication for RDS centralizes and strengthens access control by integrating database authentication with broader AWS IAM identity management. This approach eliminates the risks associated with hard-coded credentials, reduces administrative overhead for password rotation, and allows precise access management using IAM identities and policies.

Impact:

With the usage of IAM database authentication instead of static passwords, RDS clusters are protected against credential leakage and weak password practices, making unauthorized access significantly more difficult. This reduces the attack surface, supports audit and compliance, and ensures that database access is tightly aligned with enterprise identity governance and cloud security standards.

Audit:

1. List RDS clusters and check IAM authentication status:

```
aws rds describe-db-clusters \
  --query
"DBClusters[*].{DBClusterIdentifier:DBClusterIdentifier,IAMDatabaseAuthentica
tionEnabled:IAMDatabaseAuthenticationEnabled}" \
  --output table
```

2. List database users enabled for IAM authentication (RDS MySQL):

- For MySQL, connect to the cluster and run below command to ensure that required database users exist for IAM token logins.

```
SELECT user, plugin FROM mysql.user WHERE plugin='AWSAuthenticationPlugin';
```

- For Postgres, connect to the cluster and run below command to verify that necessary users are granted the rds_iam role.

```
SELECT r.rolname
FROM pg_roles AS r
JOIN pg_auth_members AS m ON r.oid = m.member
JOIN pg_roles AS g ON m.roleid = g.oid
WHERE g.rolname = 'rds_iam';
```

The cluster must have IAM database authentication enabled and users intended for IAM authentication must exist in the database engine with appropriate privileges.

Remediation:

1. Enable IAM Database Authentication on the RDS Cluster

- Use the AWS CLI to enable IAM authentication for an existing RDS cluster:

```
aws rds modify-db-cluster \
  --db-cluster-identifier <your-cluster-name> \
  --enable-iam-database-authentication \
  --apply-immediately
```

2. Create Local Database Users Configured for IAM Authentication

- For RDS MySQL: Connect to the database and create or alter users as follows:

```
CREATE USER 'jane_doe' IDENTIFIED WITH AWSAuthenticationPlugin as 'RDS';
```

or

```
ALTER USER 'jane_doe' IDENTIFIED WITH AWSAuthenticationPlugin as 'RDS';
```

- For RDS PostgreSQL:

  Grant the rds_iam role to eligible users:

```
GRANT rds_iam TO jane_doe;
```

3. Grant Necessary Privileges to the Database Users

- Assign required permissions and roles to the users within your DB engine via standard SQL GRANT commands.

4. Ensure IAM Users/Roles Have Required AWS Permissions

- The IAM principal that connects needs the following AWS permission in their policy:

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "rds-db:connect"
            ],
            "Resource": [
                "arn:aws:rds-db:region:account-
id:dbuser:DbClusterResourceId/db-user-name"
            ]
        }
    ]
}
```

- region is the AWS Region for the DB cluster
- account-id is the AWS account number for the DB cluster.
- DbClusterResourceId is the identifier for the DB cluster. This identifier is unique to an AWS Region and never changes. To find a DB cluster resource ID in the AWS Management Console for Amazon RDS, choose the DB cluster to see its details. Then choose the Configuration tab. The Resource ID is shown in the Configuration section.

5. Test the Configuration

- Use the AWS CLI to generate an authentication token:

```
aws rds generate-db-auth-token \
  --hostname <cluster-endpoint> \
  --port 3306 \
  --region <region> \
  --username <db_user>
```

- Use the generated token to authenticate to the database, confirming successful login.

For mysql:

```
mysql --host= <cluster-endpoint> \
  --port=3306 \
  --ssl-mode=REQUIRED \
  --enable-cleartext-plugin \
  --user= <db_user_name> \
  --password= '<generated_db_auth_token>'
```

For postgres:

```
psql "host=<cluster-endpoint> port=5432 dbname=<database_name>
user=<>db_user_name password='<generated_db_auth_token>' sslmode=require"
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	6.7 Centralize Access Control Centralize access control for all enterprise assets through a directory service or SSO provider, where supported.		●	●

3.14 Ensure Database has delete protection enabled (Manual)

Profile Applicability:

- Level 1

Description:

Ensure that delete protection is enabled on database instances to prevent accidental or unauthorized deletion. This setting safeguards critical databases by requiring explicit disabling of delete protection before deletion, reducing the risk of data loss through human error or malicious activity.

Rationale:

Delete protection provides a safeguard against inadvertent or malicious deletion of critical databases. By requiring deliberate action to disable deletion protection, organizations mitigate risks associated with accidental data deletion and enhance the overall resilience of their data storage platform.

Impact:

Failure to enable delete protection increases the risk of irreversible data loss, potential service disruption, and operational downtime.

Audit:

Run the following command to check if deletion protection is enabled on your RDS DB cluster(s):

```
aws rds describe-db-clusters \
  --query
"DBClusters[*].{DBClusterIdentifier:DBClusterIdentifier,DeletionProtection:De
letionProtection}" \
  --output table
```

- Review each cluster's DeletionProtection status.
- Clusters marked as true have deletion protection enabled.
- Identify any clusters with deletion protection disabled for remediation.

Remediation:

To enable deletion protection on an existing RDS DB cluster:

```
aws rds modify-db-cluster \
  --db-cluster-identifier <your-cluster-name> \
  --deletion-protection \
  --apply-immediately
```

- Replace with your RDS cluster ID.
- This change is applied immediately without downtime.

- Once enabled, the cluster cannot be deleted without first disabling deletion protection.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4 Secure Configuration of Enterprise Assets and Software Establish and maintain the secure configuration of enterprise assets (end-user devices, including portable and mobile; network devices; non-computing/IoT devices; and servers) and software (operating systems and applications).			

4 Amazon DynamoDB

Amazon DynamoDB is a fully managed NoSQL database service offered by Amazon Web Services (AWS). It is designed to provide high-performance, scalable, and reliable storage for applications that require seamless and low-latency access to data. DynamoDB is particularly well-suited for applications that need to handle large amounts of data and require quick and predictable response times.

Internal Only - General

4.1 Ensure AWS Identity and Access Management (IAM) is in use (Manual)

Profile Applicability:

- Level 1

Description:

AWS Identity and Access Management (IAM) lets you securely control your users' access to AWS services and resources. To manage access control for Amazon DynamoDB, you can create IAM policies that control access to tables and data.

Rationale:

IAM policies help you control and maintain access to Amazon DynamoDB as needed.

Audit:

1. Open IAM Console

- Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.

2. Navigate to Policies

- In the IAM console, in the navigation pane, choose `Policies`.

3. Create Policy

- Choose `Create policy`.
- You will be taken to the `Create policy` page.

4. Choose Service

- Click on `Choose a service`.
- Type `DynamoDB` in the search box and select it.

5. Configure Actions

- Under the `Actions` section, select the actions you want to allow the user to perform.
- For instance, you can select `Read` to allow read actions like GetItem, Scan, Query, etc.

6. Set Resources

- Under the `Resources` section, you can specify which tables this policy applies to.
- You can choose "All resources" or specify the ARN (Amazon Resource Name) of specific tables.

7. Review Policy

- Click on `Review policy`.
- Give your policy a name and description.
- Then click `Create policy`.
- Now, you have an IAM policy.

8. Attach Policy

- Navigate to the `Users`, `Groups`, or `Roles` section in the IAM console.
- Choose an existing user, group, or role, or create a new one.
- Once you've selected a user, group, or role, click `Add permissions`.
- Choose `Attach existing policies directly`.
- Search for your created policy, select it, and click `Attach policy`.
- With these steps, you have attached an IAM policy that controls access to DynamoDB resources.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user’s need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

4.2 Ensure Fine-Grained Access Control is implemented (Manual)

Profile Applicability:

- Level 1

Description:

Fine-Grained Access Control (FGAC) on Amazon DynamoDB allows you to control access to data at the row level. Using IAM policies, you can restrict access based on the content within the request. Here is how you can implement FGAC:

Rationale:

Fine-Grained access control helps users to create and allow specific permission within that DB.

Audit:

1. Create an IAM Role

- Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.
- In the navigation pane, choose `Roles` and select `Create role`.
- Choose `AWS service` as the type of trusted entity.
- Choose `DynamoDB` as the service that will use this role, then click `Next: Permissions`.
- On the `Attach permissions policies` page, choose `Next: Tags`. You do not need to attach a policy to this role yet.
- On the `Add tags` page, choose `Next: Review`.
- On the `Review` page, for `Role name`, enter a name for your role, such as DynamoDBFineGrainedAccessRole.
- Choose `Create role`.

2. Create an IAM Policy for Fine-Grained Access Control

- In the navigation pane, choose `Policies` and select `Create policy`.
- Choose the `JSON` tab.
- Paste the following policy into the policy document field, replacing *`us-west-2`*, *`123456789012`*, *`myddbtable`*, *`HK`*, and *`RANGEK`* with your own values:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "dynamodb:GetItem",
                "dynamodb:BatchGetItem",
                "dynamodb:Query",
                "dynamodb:PutItem",
                "dynamodb:UpdateItem",
                "dynamodb:DeleteItem"
            ],
            "Resource": "arn:aws:dynamodb:<us-west-
2:123456789012:table/myddbtable>",
            "Condition": {
                "ForAllValues:StringEquals": {
                    "dynamodb:LeadingKeys": ["${www.amazon.com:user_id}"],
                    "dynamodb:Attributes": [
                        "<HK>",
                        "<RANGEK>"
                    ]
                },
                "StringEqualsIfExists": {
                    "dynamodb:Select": "SPECIFIC_ATTRIBUTES"
                }
            }
        }
    ]
}
```

In this policy:

- `dynamodb:LeadingKeys` restrict access to only the items where the hash key value is the same as the user's ID.
- `dynamodb:Attributes` restrict access to only the "HK" and "RANGEK" attributes of the items.
- `dynamodb:Select` only allows the `SPECIFIC_ATTRIBUTES` operator.
- Choose `Next: Tags`, add any tags if needed, and then choose `Next: Review`.
- For `Name`, enter a name for your policy, such as DynamoDBFineGrainedAccessPolicy.
- Choose `Create policy`.

3. Attach the Policy to the Role

- In the navigation pane, choose `Roles`.
- Choose the role that you created in the previous step.
- On the `Permissions` tab, choose `Attach policies`.
- In the `Filter policies` search box, enter the policy name you created before.
- Select the check box for your policy, then choose `Attach policy`.

Internal Only - General

Note: Fine-grained access control is a powerful feature but can be complex to configure. Be sure to test your setup to ensure it works as expected thoroughly.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

4.3 Ensure DynamoDB Encryption at Rest (Manual)

Profile Applicability:

- Level 1

Description:

Encryption at rest in Amazon DynamoDB enhances the security of your data by encrypting it using AWS Key Management Service (AWS KMS) keys. Here is how to enable encryption at rest while creating a DynamoDB table.

Rationale:

Once the user is in their AWS account, they should open the DynamoDB to create the table and enable encryption. A key would be required to be created to enable encryption. Only the authorized user would always have access to this key. Enabling encryption would keep the user's data private and stored securely, which would only allow them to access it with their key.

Impact:

Add an additional layer of security by preventing any unauthorized personnel from accessing the data since both IAM access to the data and access to the encryption key would be required.

Audit:

1. Open DynamoDB Console

- Sign in to the AWS Management Console and open the DynamoDB console at https://console.aws.amazon.com/dynamodb/.

2. Create DynamoDB Table

- Click `Create table`. This will bring you to the `Create DynamoDB table` page.

3. Specify Table Details

- Enter a `Table name` and `Primary key`.
- The primary key consists of a partition key and, optionally, a sort key.
- Fill in these details according to your requirements.

4. Enable Encryption

- Under the `Settings` section, check the `Enable encryption at rest`.
- By default, DynamoDB uses an AWS-owned CMK to encrypt your data.

Internal Only - General

- To use an AWS-managed CMK or a customer-managed CMK instead, select `AWS-managed CMK` or `Customer-managed CMK` from the dropdown menu, then choose the desired CMK.

5. Create a Table

- Click `Create`.
- This will create your DynamoDB table with encryption at rest enabled.

Note:

1. The setting for encryption at rest applies to all DynamoDB data associated with the table, including primary key data and indexes.
2. If you need to apply encryption at rest to an existing table, you can modify the table settings. However, modifying settings on large tables could take time and impact performance during the transition.
3. Ensure you have the necessary permissions in AWS KMS when choosing an AWS-managed CMK or a customer-managed CMK.

Remediation:

References:

1. https://aws.amazon.com/products/databases/
2. https://docs.aws.amazon.com/kms/latest/developerguide/concepts.html#kms_keys

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.11 Encrypt Sensitive Data at Rest Encrypt sensitive data at rest on servers, applications, and databases containing sensitive data. Storage-layer encryption, also known as server-side encryption, meets the minimum requirement of this Safeguard. Additional encryption methods may include application-layer encryption, also known as client-side encryption, where access to the data storage device(s) does not permit access to the plain-text data.		●	●
v7	14.8 Encrypt Sensitive Information at Rest Encrypt all sensitive information at rest using a tool that requires a secondary authentication mechanism not integrated into the operating system, in order to access the information.			●

4.4 Ensure DynamoDB Encryption in Transit (Manual)

Profile Applicability:

- Level 1

Description:

Use the SSL/TLS protocol to encrypt data in transit between your applications and DynamoDB. Amazon DynamoDB encrypts data in transit by default using Transport Layer Security (TLS) encryption. Here is a step-by-step guide on how to ensure encryption in transit for your DynamoDB:

Rationale:

Amazon DynamoDB uses TLS to encrypt data during transit. To secure your data in transit the individual should identify their client application and what is supported by TLS to configure it correctly.

Impact:

If the user does not have the code configured correctly it would not be able to connect to the DynamoDB.

Audit:

1. Access the DynamoDB Console

- Sign in to the AWS Management Console and open the DynamoDB console at https://console.aws.amazon.com/dynamodb/.

2. Create or Select a DynamoDB Table

- You can create a new DynamoDB table or select an existing one to configure encryption in transit.

3. Verify Encryption Settings

- By default, DynamoDB encrypts data in transit using TLS. To ensure that encryption in transit is enabled:
- In the DynamoDB console, select your table.
- In the table details, navigate to the `Overview` tab.
- Under the `Encryption` section, verify that "Encryption at rest" is enabled. This indicates that data is encrypted at rest.
- Confirm that `Encryption in transit` is enabled. It should be enabled by default.

4. Use SSL/TLS Endpoints for API Calls

- To ensure that your API calls to DynamoDB are encrypted in transit, use SSL/TLS endpoints:
- Use the appropriate SDK or AWS CLI in your application or code that interacts with DynamoDB.
- By default, the SDKs and AWS CLI use the SSL/TLS endpoints provided by DynamoDB.
- Verify that your code is configured to connect to DynamoDB using the appropriate SSL/TLS endpoint.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.10 Encrypt Sensitive Data in Transit Encrypt sensitive data in transit. Example implementations can include: Transport Layer Security (TLS) and Open Secure Shell (OpenSSH).		●	●
v7	14.4 Encrypt All Sensitive Information in Transit Encrypt all sensitive information in transit.		●	●

Internal Only - General

4.5 Ensure VPC Endpoints are configured (Manual)

Profile Applicability:

- Level 1

Description:

Using VPC endpoints with Amazon DynamoDB allows you to securely access DynamoDB resources within your Amazon Virtual Private Cloud (VPC). This keeps your traffic off the public internet.

Rationale:

Using VPC endpoint in the DynamoDB helps ensure that the data is secured and that no external networks would have access to the network. It is a private network where the user has access to their desired availability zones and subnets.

Audit:

1. Open Amazon VPC Console

- Sign in to the AWS Management Console and open the Amazon VPC console at https://console.aws.amazon.com/vpc/.

2. Create a VPC Endpoint

- In the Amazon VPC console, navigate to the `Endpoints` section in the left-side menu.
- Click `Create Endpoint`.
- Select your desired VPC in the `VPC` dropdown menu.
- In the `Service category` section, choose `AWS services`.
- In the `Filter Services` search box, enter `DynamoDB` and select `DynamoDB` from the results.
- Choose your desired availability zone(s) and subnet(s).
- Leave the default settings for other options or customize them according to your requirements.
- Click `Create endpoint`.

3. Update Route Tables

- In the Amazon VPC console, navigate to the `Route Tables` section in the left-side menu.
- Find the route table associated with your VPC or subnet from which you want to access DynamoDB.

1. Edit the route table and add a route for the DynamoDB VPC endpoint.

Internal Only - General

    - Destination: Enter the CIDR block of the DynamoDB VPC endpoint, typically in the form of `vpce-xxxxxx-xxxxxxx-xxxxxxx-xxxxxxx.vpce.amazonaws.com/32`.
    - Target: Select the VPC endpoint ID from the dropdown menu.
2. Save the changes to update the route table.
3. Verify Connectivity

   To ensure that your VPC endpoint for DynamoDB is functioning correctly:

- Launch an Amazon EC2 instance within your VPC or use an existing one.
- Connect to the EC2 instance using SSH or other remote access methods.
- From the EC2 instance, try to access DynamoDB using the SDK or CLI.
- Ensure that the access to DynamoDB is successful and that data can be retrieved or modified.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

Additional Information:

Amazon DynamoDB uses Gateway VPC Endpoints, unlike other services that may offer Interface VPC Endpoints. There are some differences such as Gateway VPC Endpoints do not permit cross-region communication. See AWS's Documentation for more information.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	12.2 Establish and Maintain a Secure Network Architecture Establish and maintain a secure network architecture. A secure network architecture must address segmentation, least privilege, and availability, at a minimum.		●	●
v7	11.7 Manage Network Infrastructure Through a Dedicated Network Manage the network infrastructure across network connections that are separated from the business use of that network, relying on separate VLANs or, preferably, on entirely different physical connectivity for management sessions for network devices.		●	●

4.6 Ensure DynamoDB Streams and AWS Lambda for Automated Compliance Checking is Enabled (Manual)

Profile Applicability:

- Level 1

Description:

Enabling DynamoDB Streams and integrating AWS Lambda allows you to automate compliance checking and perform actions based on changes made to your DynamoDB data.

Rationale:

Enabling the DynamoDB with AWS Lambda allows the individual to either use an existing or create a new execution role that allows Lambda to access DynamoDB and write logs.

Audit:

1. Open DynamoDB Console

- Sign in to the AWS Management Console and open the DynamoDB console at https://console.aws.amazon.com/dynamodb/.

2. Create or Select a DynamoDB Table

- You can create a new DynamoDB table or select an existing one to enable DynamoDB Streams.

3. Enable DynamoDB Streams

- In the DynamoDB console, select your table.
- Click on the `Overview` tab.
- Under the `DynamoDB Streams` section, click on `Manage stream`.
- In the `Manage stream` dialog, choose `Enable` and select the desired view type (e.g., `New and old images`).
- Click `Enable`.

4. Create an AWS Lambda Function

- Open the AWS Management Console and navigate to the Lambda service at https://console.aws.amazon.com/lambda/.
- Click `Create function` to create a new Lambda function.
- Choose a function name, runtime (e.g., Node.js, Python), and other basic settings.

Internal Only - General

- Under `Permissions`, choose an existing or create a new execution role that allows Lambda to access DynamoDB and write logs.
- Click `Create function` to create the Lambda function.

5. Configure AWS Lambda with DynamoDB Stream

- Scroll down to the `Designer` section in the Lambda function editor.
- Click on `Add trigger`.
- Select `DynamoDB` from the trigger list.
- In the `Configure triggers` dialog, choose the DynamoDB table and the stream that you enabled in the previous step.
- Define the batch size and starting position, if applicable.
- Click "Add".

6. Write Lambda Function Code for Compliance Checking

- In the Lambda function editor, scroll up to the code editor section.
- Write your compliance-checking logic in the selected runtime language (e.g., Node.js, Python).
- The code should handle the incoming DynamoDB stream records and perform the necessary compliance checks.
- If needed, you can use the AWS SDKs or other libraries to interact with DynamoDB or other AWS services.

7. Configure Lambda Function Settings

- Scroll down to the `Function overview` section.
- Configure the memory, timeout, and other settings as per your requirements.
- Click `Save` to save the Lambda function.

8. Test the Compliance Checking

- You can test the compliance checking by changing the DynamoDB table and observing the Lambda function's behavior through the CloudWatch logs or other desired actions performed by the function.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

Internal Only - General

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3 Data Protection Develop processes and technical controls to identify, classify, securely handle, retain, and dispose of data.			
v7	13 Data Protection Data Protection			

4.7 Ensure Monitor and Audit Activity is enabled (Manual)

Profile Applicability:

- Level 1

Description:

Regular monitoring and auditing of activity in Amazon DynamoDB help ensure your database's security, performance, and compliance.

Rationale:

This keeps track and ensures who has recently modified a document and monitors all activity within the database. This information allows the individual to use the details provided for auditing purposes and to address any compliance requirements.

Audit:

1. Enable CloudTrail Logging for DynamoDB

- Sign in to the AWS Management Console and open the CloudTrail console at https://console.aws.amazon.com/cloudtrail/.
- Choose `Trails` from the left-side menu.
- Click `Create trail` or select an existing trail.
- Specify a trail name, choose an S3 bucket for storing logs, and configure other trail settings.
- Under `Data events`, select the checkbox for `DynamoDB` to enable logging of DynamoDB data events.
- Click `Create trail` or `Save changes` to save the CloudTrail configuration.

2. Enable DynamoDB Streams

- Sign in to the AWS Management Console and open the DynamoDB console at https://console.aws.amazon.com/dynamodb/.
- Select the DynamoDB table you want to monitor.
- Click on the `Overview` tab.
- Under the `DynamoDB Streams` section, click `Manage stream`.
- Enable DynamoDB Streams with the desired view type (e.g., `New and old images`).
- Click `Enable`.

3. Configure Amazon CloudWatch Alarms

- Sign in to the AWS Management Console and open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
- In the left-side menu, click on `Alarms`.

Internal Only - General

- Click `Create alarm`.
- Select a DynamoDB metric to monitor (e.g., Read or Write capacity units).
- Configure the threshold, conditions, and actions for the alarm.
- Choose the actions to take when the alarm state is triggered (e.g., send notifications, auto-scaling actions, etc.).
- Click `Create alarm` to save the configuration.

4. Analyze and Review Logs and Metrics

- Sign in to the AWS Management Console and open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
- In the left-side menu, click `Logs` to access CloudWatch Logs.
- Select the appropriate log group for DynamoDB (e.g., `/aws/dynamodb/TableName`).
- Review the logs to monitor activities, errors, and any unusual behavior.
- Navigate to the CloudWatch console and click `Metrics` in the left-side menu.
- Select the DynamoDB namespace and the desired metrics (e.g., ConsumedReadCapacityUnits, ConsumedWriteCapacityUnits).
- Analyze the metrics to identify trends, capacity needs, and potential issues.

5. Enable AWS Config for DynamoDB

- Sign in to the AWS Management Console and open the AWS Config console at https://console.aws.amazon.com/config/.
- Click on `Rules` in the left-side menu.
- Click `Add rule`.
- Configure a rule for DynamoDB compliance checks, such as checking for unencrypted tables or insecure IAM policies.
- Customize the rule settings and scope based on your requirements.
- Click `Save` to create the AWS Config rule.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	8.1 Establish and Maintain an Audit Log Management Process Establish and maintain an audit log management process that defines the enterprise's logging requirements. At a minimum, address the collection, review, and retention of audit logs for enterprise assets. Review and update documentation annually, or when significant enterprise changes occur that could impact this Safeguard.	●	●	●
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●

Internal Only - General

4.8 Ensure Database has delete protection enabled (Manual)

Profile Applicability:

- Level 1

Description:

Ensure that delete protection is enabled on database instances to prevent accidental or unauthorized deletion. This setting safeguards critical databases by requiring explicit disabling of delete protection before deletion, reducing the risk of data loss through human error or malicious activity.

Rationale:

Delete protection provides a safeguard against inadvertent or malicious deletion of critical databases. By requiring deliberate action to disable deletion protection, organizations mitigate risks associated with accidental data deletion and enhance the overall resilience of their data storage platform.

Impact:

Failure to enable delete protection increases the risk of irreversible data loss, potential service disruption, and operational downtime.

Audit:

To check whether delete protection is enabled on your DynamoDB tables, use the following command for each table:

```
aws dynamodb describe-table --table-name <your-table-name> \
  --query "{TableName: Table.TableName, DeleteProtectionEnabled: 
Table.DeletionProtectionEnabled}" \
  --output table
```

- Replace with your DynamoDB table name.
- This will return True if delete protection is enabled, False otherwise.

Remediation:

To enable delete protection on an existing DynamoDB table, use the following command:

```
aws dynamodb update-table \
    --table-name my-table \
    --deletion-protection-enabled
```

- Replace with your DynamoDB table name.
- Delete protection prevents the table from being deleted until the protection is disabled explicitly.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4 Secure Configuration of Enterprise Assets and Software Establish and maintain the secure configuration of enterprise assets (end-user devices, including portable and mobile; network devices; non-computing/IoT devices; and servers) and software (operating systems and applications).			

Internal Only - General

4.9 Ensure Database has Backup enabled (Manual)

Profile Applicability:

- Level 1

Description:

Ensure your DynamoDB tables have backups enabled to protect against accidental data loss or corruption. This can be achieved in two ways: by enabling Point-in-Time Recovery (PITR), which provides continuous backups for up to 35 days, and by configuring on-demand backups that can be automated using the AWS Backup service.

Rationale:

Backups are essential for restoring data after accidental deletions, application errors or malicious events. Enabling both continuous and scheduled backups maximizes data resilience while meeting recovery point objectives (RPO) and compliance mandates.

Impact:

Without backups, data loss can be permanent which can lead to business disruption and potential regulatory penalties. Enabling backup provides assurance of data availability and accelerates recovery from incidents, supporting overall system reliability.

Audit:

1. Check if Point-in-Time Recovery (PITR) is enabled

Use the following steps on the AWS console to verify if PITR is enabled for your DynamoDB table and also conform the number of days its retained for:

1.1 Log in to the AWS Management Console.

1.2 Navigate to DynamoDB in the AWS Services menu.

- Select "Tables" from the left sidebar.

1.3 Click on the specific table name that you want to audit.

1.4 Go to the “Backups” tab in the table’s navigation bar.

1.5 Under the “Point-in-time recovery (PITR)” section:

- Verify that “Status” is “On”, which means PITR is enabled.
- Check the “Backup recovery period” value, this displays the configured retention period in days.
- Optionally, review the Earliest restore point and Latest restore point timestamps to confirm the exact window for recovery.

2. Check if automated backups are enabled via AWS Backup Service

AWS Backup allows scheduled, automated backups of DynamoDB tables (if integrated). To confirm backup plan settings, list backup plans and recovery points:

2.1 Check if Table is Assigned to a Backup Plan

- List backup plans:

```
aws backup list-backup-plans --query "BackupPlansList[].BackupPlanName" --
output table
```

- For each backup plan, list all backup selections (resource assignments):

```
aws backup list-backup-selections --backup-plan-id <your-backup-plan-id> --
query "BackupSelections[].SelectionId" --output text
```

- For each selection, list assigned resources and search for your table:

```
aws backup get-backup-selection --backup-plan-id <your-backup-plan-id> --
selection-id <selection-id> --query "BackupSelection.Resources" --output text
```

- If your table ARN appears in any selection, it is protected by the backup plan.

2.2 Verify Backup Plan Configuration

```
aws backup get-backup-plan --backup-plan-id <your-backup-plan-id>
```

Look for the "Lifecycle" fields in each backup rule:

- "DeleteAfterDays" is the retention period.
- "ScheduleExpression" sets the backup schedule (cron format).
- "BackupVaultName" is the name of the vault (where backups are stored).

Summary:

1. Confirm PITR is enabled for the table and note it's retention period (up to 35 days).
2. Confirm if AWS Backup plans exist and are actively creating recovery points for your DynamoDB tables.

Remediation:

1. Create Backup Plan with 2 Rules (Continuous and Scheduled Snapshots)

```
aws backup create-backup-plan --backup-plan '{
  "BackupPlanName": "<BackupPlanName>",
  "Rules": [
    {
      "RuleName": "Continuous-PITR",
      "TargetBackupVaultName": "Default",
      "ScheduleExpression": "cron(0 * * * ? *)",
      "StartWindowMinutes": 60,
      "CompletionWindowMinutes": 180,
      "Lifecycle": { "DeleteAfterDays": 35 },
      "RecoveryPointTags": { "BackupType": "Continuous" },
      "EnableContinuousBackup": true
    },
    {
      "RuleName": "Scheduled-OnDemand-Snapshots",
      "TargetBackupVaultName": "Default",
      "ScheduleExpression": "cron(0 3 ? * SUN *)",
      "StartWindowMinutes": 120,
      "CompletionWindowMinutes": 360,
      "Lifecycle": { "DeleteAfterDays": 90 },
      "RecoveryPointTags": { "BackupType": "OnDemand" }
    }
  ]
}'
```

- Replace "BackupPlanName" with your backup plan name
- Replace "Default" with your actual backup vault name if different
- Update the "ScheduleExpression", "StartWindowMinutes" based on your unique backup schedule. Note, this schedule is ignored for PITR as that is continuous backup.
- Replace "35" with your retention time for PITR if shorter
- Replace "90" with your retention time for Snapshots if different

This command outputs the BackupPlanId necessary for the next step.

2. Assign DynamoDB Table to the Backup Plan

Replace <BackupPlanId> with the ID from Step 1 and <TableArn> with your DynamoDB table ARN.

```
aws backup create-backup-selection \
  --backup-plan-id <BackupPlanId> \
  --backup-selection '{
    "SelectionName": "DynamoDBTableSelection",
    "IamRoleArn": "arn:aws:iam::<account-
id>:role/AWSBackupServiceRolePolicyForBackup",
    "Resources": ["<TableArn>"]
  }'
```

- Make sure the IAM role has the necessary AWS Backup permissions.

These commands create a centralized backup plan with continuous and scheduled snapshot backups, then assign your DynamoDB table as a resource for backup.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	11.2 Perform Automated Backups Perform automated backups of in-scope enterprise assets. Run backups weekly, or more frequently, based on the sensitivity of the data.	●	●	●

5 Amazon ElastiCache

Amazon ElastiCache is a managed in-memory caching service provided by Amazon Web Services (AWS). It is designed to help improve the performance and scalability of applications by allowing them to quickly access and retrieve data that is frequently accessed. ElastiCache is compatible with popular in-memory data stores like Redis and Memcached.

Internal Only - General

5.1 Ensure Secure Access to ElastiCache (Manual)

Profile Applicability:

- Level 1

Description:

Securing access to Amazon ElastiCache involves implementing appropriate authentication and authorization mechanisms.

Rationale:

Audit:

1. Use AWS Identity and Access Management (IAM)

- Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.
- Create IAM users or roles for individuals or applications needing ElastiCache access.
- Define fine-grained permissions using IAM policies to allow only necessary actions on ElastiCache resources.
- Assign IAM policies to the IAM users or roles to grant access.

2. Implement Secure Network Access

- Place your ElastiCache cluster within a Virtual Private Cloud (VPC) to control network access.
- Create and configure security groups to allow access only from trusted networks or specific IP ranges.
- Ensure your VPC's network ACLs (Access Control Lists) are properly configured to restrict inbound and outbound traffic.

3. Enable Encryption in Transit

- Configure your ElastiCache cluster to use SSL/TLS encryption for client connections.
- Use the `--transit-encryption-enabled` parameter when creating or modifying the cluster to enable encryption in transit.
- Update your client applications to connect to the ElastiCache cluster using SSL/TLS.

4. Protect ElastiCache Credentials

- Avoid sharing access keys, secret keys, or IAM user credentials between individuals.

Internal Only - General

- Use IAM roles for Amazon EC2 instances or other AWS services to securely access ElastiCache without needing credentials.
- Rotate your access keys regularly and disable or remove unnecessary IAM users or roles.

5. Enable Event Logging and Monitoring

- Enable CloudWatch Logs for your ElastiCache clusters to capture logs and monitor activities.
- Configure CloudWatch Alarms to be notified of any unusual or suspicious behavior.
- Set up CloudTrail to log API calls made to ElastiCache for auditing and compliance purposes.

6. Regularly Review and Update Access Controls

- Perform regular reviews of IAM policies, security groups, and network ACLs to ensure they align with your security requirements.
- Remove any unnecessary or excessive privileges from IAM policies.
- Stay updated with AWS security best practices and recommendations to improve access controls.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

5.2 Ensure Network Security is Enabled (Manual)

Profile Applicability:

- Level 1

Description:

Implementing network security for Amazon ElastiCache involves configuring your Virtual Private Cloud (VPC), security groups, and network access controls to control access to your ElastiCache clusters.

Rationale:

This helps ensure that the data is safe and protected from any threats and or misconfigurations within the network. This helps to keep a potential hacker getting into the system and compromising the data.

Audit:

1. Create or Select a VPC

- Sign in to the AWS Management Console and open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
- Create a new VPC or select an existing VPC where you want to deploy your ElastiCache cluster.

2. Create Subnets

- In the VPC console, navigate to `Subnets` in the left-side menu.
- Create or select the desired subnets within your VPC where you want to deploy your ElastiCache cluster.

3. Configure Security Groups

- In the VPC console, navigate to `Security Groups` in the left-side menu.
- Create a new security group.

  Or select an existing one to configure the security settings for your ElastiCache cluster.

- Define inbound and outbound rules to control the traffic flow to and from your ElastiCache cluster.
  - Allow inbound traffic from trusted sources (e.g., specific IP ranges or security groups) on the necessary ports used by your ElastiCache cluster.
  - Define outbound rules based on your requirements, such as allowing outbound traffic to specific destinations or ports.

Internal Only - General

- Associate the security group with the ElastiCache cluster when creating or modifying it.

4. Set up Network Access Control Lists (ACLs)

- In the VPC console, navigate to `Network ACLs` in the left-side menu.
- Create or select the appropriate network ACL associated with the subnets used by your ElastiCache cluster.
- Configure inbound and outbound rules in the network ACL to allow or deny traffic to and from your ElastiCache cluster.
  - Define rules based on your security requirements, allowing only necessary protocols, ports, and IP ranges.
- Associate the network ACL with the subnets used by your ElastiCache cluster.

5. Configure Route Tables

- In the VPC console, navigate to `Route Tables` in the left-side menu.
- Create or select the route table associated with the subnets used by your ElastiCache cluster.
- Add or modify routes to ensure traffic to and from your ElastiCache cluster flows correctly.
  - Ensure that the route table has an appropriate route to the internet gateway or virtual private gateway if external connectivity is required.
- Associate the route table with the subnets used by your ElastiCache cluster.

6. Verify Connectivity and Test

- Launch an Amazon EC2 instance within the same VPC and subnet as your ElastiCache cluster or use an existing one.
- Connect to the EC2 instance using SSH or other remote access methods.
- Test the connectivity to your ElastiCache cluster by trying to connect to it using the appropriate client or utility.
- Verify that the network security settings allow the necessary traffic and deny unauthorized access.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	12.2 Establish and Maintain a Secure Network Architecture Establish and maintain a secure network architecture. A secure network architecture must address segmentation, least privilege, and availability, at a minimum.		●	●
v7	11.7 Manage Network Infrastructure Through a Dedicated Network Manage the network infrastructure across network connections that are separated from the business use of that network, relying on separate VLANs or, preferably, on entirely different physical connectivity for management sessions for network devices.		●	●

5.3 Ensure Encryption at Rest and in Transit is configured (Manual)

Profile Applicability:

- Level 1

Description:

Enabling encryption at rest and in transit for Amazon ElastiCache helps protect your data when it is stored and transmitted.

Rationale:

Enabling encryption at rest secured the users data where it is stored. Enabling encryption in transit helps that the data is protected when it is moving from one location to another.

Impact:

If the user didn't enable encryption and rest and during transit, there is a possibility of the data being vulnerable to a ransomware attack.

Audit:

1. Enable Encryption at Rest

- Sign in to the AWS Management Console and open the Amazon ElastiCache console at https://console.aws.amazon.com/elasticache/.
- Create a new ElastiCache cluster or select an existing cluster.
- On the cluster details page, click the `Encryption` tab.
- Select the option to enable encryption Under the `Encryption at Rest` section.
- Choose the desired encryption type:
  - list text hereDefault Encryption: Select this option to use the default AWS-managed key for encryption.
  - list text hereCustomer Managed Key (CMK): Select this option to use your own AWS Key Management Service (KMS) customer-managed key for encryption.
- If you selected `Customer Managed Key (CMK)`, choose the appropriate KMS key from the dropdown menu.
- Click "Save changes" to enable encryption at rest for the ElastiCache cluster.

2. Enable Encryption in Transit

- On the ElastiCache cluster details page, click the `Encryption` tab.
- Select the option to enable encryption Under the "Encryption in Transit" section.
- Choose the desired encryption type:

  - list text hereTransit encryption enabled with SSL/TLS: Select this option to enable encryption in transit using SSL/TLS encryption.
  - list text hereTransit encryption disabled: Select this option if you do not require encryption in transit.
- Click `Save changes` to enable encryption in transit for the ElastiCache cluster.

3. Verify the Encryption Status

- Wait a few minutes for the changes to propagate and the encryption to take effect.
- Refresh the ElastiCache console and navigate to the cluster details page.
- Verify that the encryption status is now enabled for both encryptions at rest and in transit.

Remediation:

The user has two options when it comes to encryption at rest and in transit to choose from. Depending on what actions the user selects from it determines how their data is going to be protected.

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.10 Encrypt Sensitive Data in Transit Encrypt sensitive data in transit. Example implementations can include: Transport Layer Security (TLS) and Open Secure Shell (OpenSSH).		●	●
v8	3.11 Encrypt Sensitive Data at Rest Encrypt sensitive data at rest on servers, applications, and databases containing sensitive data. Storage-layer encryption, also known as server-side encryption, meets the minimum requirement of this Safeguard. Additional encryption methods may include application-layer encryption, also known as client-side encryption, where access to the data storage device(s) does not permit access to the plain-text data.		●	●
v7	14.4 Encrypt All Sensitive Information in Transit Encrypt all sensitive information in transit.		●	●
v7	14.8 Encrypt Sensitive Information at Rest Encrypt all sensitive information at rest using a tool that requires a secondary authentication mechanism not integrated into the operating system, in order to access the information.			●

5.4 Ensure Automatic Updates and Patching are Enabled (Manual)

Profile Applicability:

- Level 1

Description:

Enabling automatic updates and patching for Amazon ElastiCache ensures that your ElastiCache clusters run the latest software versions with important security fixes and enhancements.

Rationale:

Automatic updates help the software be updated and address any vulnerabilities within the software that can help business with any potential exists that can impact the business and prevent any unauthorized access.

Audit:

1. Sign in to the AWS Management Console

- Sign in to the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the ElastiCache Console

- Open the Amazon ElastiCache console by navigating to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/elasticache/.

3. Select the ElastiCache Cluster

- Choose the ElastiCache cluster you want to enable automatic updates and patching.
- Click on the cluster name to access its details page.

4. Enable Automatic Updates

- Click on the `Configuration` tab on the cluster details page.
- Scroll down to the `Cluster details` section.
- Under `Cluster maintenance and updates`, click `Modify`.
- In the `Maintenance and updates` dialog, find the `Auto minor version upgrade` option and select `Enable`.
- Leave other settings unchanged or adjust them according to your requirements.
- Click `Save` to apply the changes.

5. Verify Automatic Updates Status

- Wait for a few moments for the changes to take effect.
- Refresh the cluster details page to see the updated configuration.
- Verify that the "Auto minor version upgrade" setting is now enabled for the ElastiCache cluster.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	7 Continuous Vulnerability Management Develop a plan to continuously assess and track vulnerabilities on all enterprise assets within the enterprise's infrastructure, in order to remediate, and minimize, the window of opportunity for attackers. Monitor public and private industry sources for new threat and vulnerability information.			
v7	3 Continuous Vulnerability Management Continuous Vulnerability Management			

5.5 Ensure Virtual Private Cloud (VPC) is Enabled (Manual)

Profile Applicability:

- Level 1

Description:

Implementing VPC security best practices for Amazon ElastiCache involves configuring your Virtual Private Cloud (VPC) and associated resources to enhance the security of your ElastiCache clusters.

Rationale:

This ensures that only authorized users can access their platforms and prevents any mistakes that can lead to a data breach due to the level of security.

Audit:

1. Create or Select a VPC

- Sign in to the AWS Management Console and open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
- Create a new VPC or select an existing VPC to host your ElastiCache clusters.

2. Configure Subnets

- In the VPC console, navigate to `Subnets` in the left-side menu.
- Create or select the subnets within your VPC where you want to deploy your ElastiCache clusters.
- Ensure you have private subnets for your ElastiCache clusters to avoid exposing them to the public internet.

3. Define Security Groups

- In the VPC console, navigate to `Security Groups` in the left-side menu.
- Create a new security group or select an existing one for your ElastiCache clusters.
- Configure inbound and outbound rules in the security group to control traffic access.
  - Allow inbound access only from trusted sources or specific IP ranges required for your applications.
  - Restrict outbound access to necessary destinations and protocols.
- Associate the security group with your ElastiCache clusters.

4. Configure Network Access Control Lists (ACLs)

- In the VPC console, navigate to `Network ACLs` in the left-side menu.

- Create or select the network ACLs associated with the subnets used by your ElastiCache clusters.
- Configure inbound and outbound rules in the network ACLs to control traffic access.
    - Define rules based on your security requirements, allowing only necessary protocols, ports, and IP ranges.
    - Deny unnecessary or unwanted traffic.
- Associate the network ACLs with the subnets used by your ElastiCache clusters.

5. Configure Routing

- In the VPC console, navigate to `Route Tables` in the left-side menu.
- Create or select the route table associated with the subnets used by your ElastiCache clusters.
- Add or modify routes to ensure traffic flows correctly to and from your ElastiCache clusters.
- Ensure that the route table has appropriate routes to the internet gateway or virtual private gateway if external connectivity is required.
- Associate the route table with the subnets used by your ElastiCache clusters.

6. Review and Update Network Security Settings

- Regularly review and update your VPC security configurations, including security groups, network ACLs, and routing, to align with your security requirements.
- Remove any unnecessary or excessive permissions from security groups and tighten inbound and outbound access as needed.
- Stay informed about AWS security best practices and recommendations to enhance your network security.

Remediation:

The individual is required to create a subnet and configure their inbound and outbound access. Individuals are supposed to configure their ACL and routing ensuring the traffic is flowing smoothly without any interference. This control is important because it only allows authorized user to access their resources as they prefer.

References:

1. https://aws.amazon.com/products/databases/

Internal Only - General

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	12.2 Establish and Maintain a Secure Network Architecture Establish and maintain a secure network architecture. A secure network architecture must address segmentation, least privilege, and availability, at a minimum.		●	●
v7	11.7 Manage Network Infrastructure Through a Dedicated Network Manage the network infrastructure across network connections that are separated from the business use of that network, relying on separate VLANs or, preferably, on entirely different physical connectivity for management sessions for network devices.		●	●

5.6 Ensure Monitoring and Logging is Enabled (Manual)

Profile Applicability:

- Level 1

Description:

Implementing monitoring and logging for Amazon ElastiCache allows you to gain visibility into the performance, health, and behavior of your ElastiCache clusters.

Rationale:

This helps the individual know what is being logged within the activity and determine what next step they should take to address any suspicious activity.

Impact:

If the individual is not monitoring and logging their activity it allows the attacker to attack the system and extract or destroy data.

Audit:

1. Sign in to the AWS Management Console

- Sign in to the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the ElastiCache Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/elasticache/.

3. Select the ElastiCache Cluster

- Choose the ElastiCache cluster for which you want to implement monitoring and logging.
- Click on the cluster name to access its details page.

4. Enable Enhanced Monitoring

- Click on the `Monitoring` tab on the cluster details page.
- Under the `Monitoring` section, click on the `Enable Enhanced Monitoring` button.
- Select the desired monitoring granularity (1 minute, 5 minutes, or 60 minutes) to capture detailed metrics.
- Choose the desired CloudWatch namespace to store the metrics.
- Click `Save changes` to enable enhanced monitoring for the ElastiCache cluster.

5. Set Up CloudWatch Alarms

- In the CloudWatch console, navigate to `Alarms` in the left-side menu.
- Click `Create alarm` to create a new alarm.
- Select the appropriate ElastiCache metrics from the available options.
- Configure the threshold, conditions, and actions for the alarm.
- Choose the actions to take when the alarm state is triggered (e.g., send notifications, auto-scaling actions, etc.).
- Click `Create alarm` to save the alarm configuration.

6. Configure CloudWatch Logs

- In the CloudWatch console, navigate to `Logs` in the left-side menu.
- Click `Create log group` to create a new one.
- Provide a unique name for the log group and optionally specify a retention period for log data.
- Click `Create log group` to create the log group.
- On the ElastiCache cluster details page, click the `Logging` tab.
- Enable the `CloudWatch Logs` option and select the desired log group from the dropdown menu.
- Click `Save changes` to enable CloudWatch Logs for the ElastiCache cluster.

7. Verify Monitoring and Logging

- Wait a few minutes for the monitoring and logging configurations to take effect.
- Refresh the cluster details page for the updated monitoring and logging status.
- Navigate to the CloudWatch console to view metrics, alarms, and logs related to your ElastiCache cluster.

Remediation:

The individual can understand the health, performance, and behavior of their clusters which allows them to address any unusual activity that takes place.

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	8.1 Establish and Maintain an Audit Log Management Process Establish and maintain an audit log management process that defines the enterprise’s logging requirements. At a minimum, address the collection, review, and retention of audit logs for enterprise assets. Review and update documentation annually, or when significant enterprise changes occur that could impact this Safeguard.	●	●	●
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●

5.7 Ensure Security Configurations are Reviewed Regularly (Manual)

Profile Applicability:

- Level 1

Description:

Regularly updating and reviewing the security configuration of your Amazon ElastiCache clusters helps ensure that your clusters are protected against potential vulnerabilities and aligned with your security requirements.

Rationale:

This ensures that the clusters are being regularly updated and protected from any potential vulnerabilities as well as meeting the security requirements.

Impact:

Updating the system and being updated with security configurations keeps everything secure and prevents it from an attack.

Audit:

1. Sign in to the AWS Management Console

- Sign in to the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the ElastiCache Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/elasticache/.

3. Select the ElastiCache Cluster

- Choose the ElastiCache cluster you want to update and review the security configuration. Click on the cluster name to access its details page.

4. Review IAM Policies

- Navigate to the `Configuration` tab on the cluster details page.
- Click on the `IAM Access` tab.
- Review the IAM policies associated with the ElastiCache cluster and its resources.
- Ensure that the IAM policies provide the least privileged access, granting only the necessary permissions to users and roles.

- Update the IAM policies as required based on changes in access requirements or security best practices.

5. Review Security Groups

- Navigate to the `Configuration` tab on the cluster details page.
- Click on the `Security Groups` tab.
- Review the security groups associated with the ElastiCache cluster.
- Ensure that the inbound and outbound rules of the security groups are configured correctly and restrict access to necessary ports and IP ranges.
- Update the security group rules as needed to align with your security requirements.

6. Review Encryption Settings

- Navigate to the `Configuration` tab on the cluster details page.
- Click on the `Encryption at Rest` tab.
- Verify the encryption settings for the ElastiCache cluster.
- Ensure that encryption at rest is enabled and using the appropriate encryption type (default AWS-managed key or customer-managed key).
- Update the encryption settings if necessary to comply with your security policies.

7. Review Network Security

- Navigate to the `Configuration` tab on the cluster details page.
- Click on the `Network & Security` tab.
- Review the VPC, subnets, security groups, and network ACLs associated with the ElastiCache cluster.
- Ensure that the VPC and subnet configurations align with your security requirements.
- Update the network security settings as needed to maintain a secure network architecture.

8. Review Access Control

- Navigate to the `Configuration` tab on the cluster details page.
- Click on the `Security` tab.
- Review the authentication and access control settings for the ElastiCache cluster.
- Ensure that the authentication method (no password, transit encryption, or encryption in transit) meets your security standards.
- Update the access control settings as required to align with your security policies.

9. Regularly Monitor Security Bulletins

- Stay updated with AWS security bulletins, advisories, and best practices.

- Regularly review security-related announcements from AWS.
- Take necessary actions based on security recommendations, such as applying patches or configuration changes.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	5 Account Management Use processes and tools to assign and manage authorization to credentials for user accounts, including administrator accounts, as well as service accounts, to enterprise assets and software.			
v7	5 Secure Configuration for Hardware and Software on Mobile Devices, Laptops, Workstations and Servers Secure Configuration for Hardware and Software on Mobile Devices, Laptops, Workstations and Servers			

Internal Only - General

5.8 Ensure Authentication and Access Control is Enabled (Manual)

Profile Applicability:

- Level 1

Description:

Individual creates IAM roles that would give specific permission to what the user can and cannot do within that database. The Access Control List (ACLs) allows only specific individuals to access the resources.

Rationale:

Impact:

Use specific client's applications or tools that allow the authorized personnel to connect to the database.

Audit:

1. Sign in to the AWS Management Console

- Sign in to the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon Keyspaces Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/keyspaces/.

3. Select the Keyspace

- Choose the Keyspace (database) for which you want to implement authentication and access control.
- Click on the Keyspace name to access its details page.

4. Enable IAM for Cassandra

- In the Keyspace details page, click on the `Configuration` tab.
- Under the `Authentication and access control` section, locate the "IAM for Cassandra" option.
- Click on `Edit`.
- Select the `Enable` option to enable IAM for Cassandra authentication and authorization.
- Choose the IAM role(s) that can access the Keyspace

- Click `Save` to enable IAM for Cassandra.

5. Define IAM Roles and Permissions

- Open the IAM console by navigating to `Identity and Access Management (IAM)` in the AWS Management Console.
- Create IAM roles with appropriate policies defining the desired access level to your Amazon Keyspaces resources.
- You may create different roles for different user groups or applications.
- Ensure that the IAM policies associated with these roles allow the necessary permissions for interacting with Keyspaces.
- Attach the IAM roles to the appropriate AWS identities, such as IAM users or AWS Identity and Access Management roles.

6. Review and Update Access Control

- In the Keyspace details page, click on the `Configuration` tab.
- Under the `Authentication and access control` section, click on `Access Control Lists` (ACLs).
- Review the ACLs to define fine-grained access control at the table and row level.
  - Define rules that allow or deny access based on specific conditions, such as IP addresses or IAM roles.
  - Ensure that the ACL rules align with your security requirements and restrict access to sensitive data if necessary.
- Update the ACLs as needed to accommodate changes.

7. Verify Authentication and Access Control

- Test the authentication and access control mechanisms using client applications or tools that connect to your Amazon Keyspaces resources.
- Verify that only authorized users or applications can access the Keyspaces resources based on the defined IAM roles and ACL rules.
- Monitor the access logs and perform periodic reviews to ensure the authentication and access control measures function as intended.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user’s need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

5.9 Ensure Audit Logging is Enabled (Manual)

Profile Applicability:

- Level 1

Description:

To manage your enterprise caching solution, it is important that you know how your clusters are performing and the resources they are consuming. It's also important that you know the events that are being generated and the costs of your deployment.

Amazon CloudWatch provides metrics for monitoring your cache performance. In addition, cost allocation tags help you monitor and manage costs.

Rationale:

Impact:

Reduce the risk of any fraud or inconsistency within the database because only authorized user has access to it.

Audit:

1. Sign in to the AWS Management Console

- Sign in to the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon Keyspaces Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/keyspaces/.

3. Select the Keyspace

- Choose the Keyspace (database) for which you want to enable audit logging.
- Click on the Keyspace name to access its details page.

4. Enable Amazon CloudWatch Logs

- In the Keyspace details page, click on the `Configuration` tab.
- Under the `Logging` section, locate the `CloudWatch Logs` option.
- Click on `Edit`.
- Select the `Enable` option to enable logging for the Keyspace.
- Choose an existing CloudWatch Logs log group or create a new one to store the logs generated by the Keyspace activities.
- Click `Save` to enable CloudWatch Logs for the Keyspace.

5. Configure CloudWatch Logs

- Open the CloudWatch console by navigating to `CloudWatch` in the AWS Management Console.
- In the left-side menu, click on `Logs`.
- Create a new log group or select an existing log group that will store the Keyspaces logs.
- Configure log retention settings based on your retention requirements. Logs can be stored for a specific number of days or indefinitely.
- Define any necessary log group permissions to control access to the logs.
- Optionally, set up log exports or alarms for specific log events or patterns if needed.

6. Verify the Logging Status

- Wait a few minutes for the changes to propagate and the logging configuration to take effect.
- Refresh the Keyspace details page to see the updated logging status.
- Verify that CloudWatch Logs is enabled for the Keyspace.

7. Monitor and Analyze Logs

- Navigate to the CloudWatch console and select the log group that stores the Keyspaces logs.
- Monitor the logs to gain insights into the activities and operations performed on your Keyspace.
- Use CloudWatch Logs features, such as log searching, filtering, and visualization, to analyze the logs and identify any security or operational issues.
- Establish appropriate log monitoring and alerting mechanisms to proactively identify and respond to potential security incidents or operational anomalies.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	8.1 Establish and Maintain an Audit Log Management Process Establish and maintain an audit log management process that defines the enterprise's logging requirements. At a minimum, address the collection, review, and retention of audit logs for enterprise assets. Review and update documentation annually, or when significant enterprise changes occur that could impact this Safeguard.	●	●	●
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●

5.10 Ensure Security Configurations are Reviewed Regularly (Manual)

Profile Applicability:

- Level 1

Description:

Regularly updating and reviewing the security configuration of your Amazon Keyspaces environment helps ensure that your database is protected against potential vulnerabilities and aligned with your security requirements.

Rationale:

Impact:

If you are not updating these regularly, your database would most likely become susceptible to a vulnerable attack. Not updating your IAM permission, network, and encryption setting, and controlling audit logging, would lead to the attacker getting into the system which would result in data loss.

Audit:

1. Sign in to the AWS Management Console

- Sign in to the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon Keyspaces Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/keyspaces/.

3. Select the Keyspace

- Choose the Keyspace (database) for which you want to update and review the security configuration.
- Click on the Keyspace name to access its details page.

4. Review IAM Roles and Permissions

- In the Keyspace details page, click on the `Configuration` tab.
- Under the `Authentication and access control` section, review the IAM roles and permissions associated with the Keyspace.
- Ensure that the IAM roles have appropriate permissions and follow the principle of least privilege.

- Review the IAM policies and make any necessary updates to align with your security requirements.

5. Review Network Security

- In the Keyspace details page, click on the `Configuration` tab.
- Under the `Network & Security` section, review the VPC, subnets, security groups, and network ACLs associated with the Keyspace.
- Ensure that the VPC and subnet configurations align with your security requirements.
- Review the security group rules and network ACL rules to ensure they restrict access to necessary ports, IP ranges, and protocols.
- Make any necessary updates to tighten the network security settings.

6. Review Encryption Settings

- In the Keyspace details page, click on the `Configuration` tab.
- Under the `Encryption` section, review the encryption settings for the Keyspace.
- Ensure that encryption at rest and in transit are enabled and the appropriate encryption options are chosen.
- Review any customer-managed keys used for encryption and verify their configurations.

7. Review Access Control

- In the Keyspace details page, click on the `Configuration` tab.
- Under the `Authentication and access control` section, review the Access Control Lists (ACLs) for the Keyspace.
- Ensure the ACLs define appropriate access permissions at the table and row levels.
- Review the ACL rules and make any necessary updates to align with your security policies and access requirements.

8. Review Audit Logging

- In the Keyspace details page, click on the `Configuration` tab.
- Review the Keyspace's logging configuration under the `Logging` section.
- Ensure the logs are captured and stored in CloudWatch Logs as expected.
- Please review the log retention settings and y that they comply with your retention policies.

9. Regularly Monitor Security Bulletins

- Stay updated with AWS security bulletins, advisories, and best practices.
- Monitor AWS security announcements and subscribe to relevant security notifications.

- Regularly review and apply security patches, updates, and recommended configuration changes for Amazon Keyspaces.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	5 Account Management Use processes and tools to assign and manage authorization to credentials for user accounts, including administrator accounts, as well as service accounts, to enterprise assets and software.			
v7	5 Secure Configuration for Hardware and Software on Mobile Devices, Laptops, Workstations and Servers Secure Configuration for Hardware and Software on Mobile Devices, Laptops, Workstations and Servers			

Internal Only - General

5.11 Ensure ElastiCache has Cluster Mode Enabled (Manual)

Profile Applicability:

- Level 2

Description:

Cluster Mode Enabled for ElastiCache distributes data across multiple shards, enabling horizontal scaling, higher availability, and isolating potential failures or resource exhaustion to a subset of the data set, rather than the entire cluster.

Rationale:

Enabling Cluster Mode reduces the risk of service outage from node failures, hardware limits, or scaling bottlenecks. Data partitioning across shards allows zero-downtime horizontal scaling, automated failover, and better resource utilization in production workloads, especially under high load or large data sets.

Impact:

Enabling Cluster Mode transforms ElastiCache from a single fault domain into a distributed, highly available system.

Audit:

List Cluster Mode Status for All Replication Groups

```
aws elasticache describe-replication-groups \
  --query "ReplicationGroups[*].{ReplicationGroupId:ReplicationGroupId,
ClusterModeEnabled:ClusterEnabled}"
```

- This will output a concise list showing each cluster's ID and whether Cluster Mode is enabled (true) or not (false).
- Review and flag any replication group entries where "ClusterModeEnabled": false.

Remediation:

Migration from Cluster Mode Disabled (CMD) to Cluster Mode Enabled (CME) is possible via the cluster mode compatible feature provided by AWS.

1. Pre-requisites:

- The cluster may only have keys in database 0 only.
- Applications must use a Valkey or Redis OSS client that is capable of using Cluster protocol and use a configuration endpoint.
- Auto-failover must be enabled on the cluster with a minimum of 1 replica.
- The minimum engine version required for migration is Valkey 7.2 and above, or Redis OSS 7.0 and above.

2. Modify Cluster Mode to Compatible

- Change the existing replication group cluster mode from disabled to Compatible:

```
aws elasticache modify-replication-group \
  --replication-group-id <your-replication-group-id> \
  --cluster-mode compatible \
  --apply-immediately
```

- In this mode, ElastiCache behaves as a single shard cluster but supports both cluster mode enabled and disabled client connections.

3. Migrate All Clients to Cluster Mode Enabled

- Update your application clients to support the cluster protocol using the cluster configuration endpoint.
- Validate application behavior in this intermediate compatible mode.
- This allows your client applications to start transitioning to the cluster-aware mode while maintaining backward compatibility.

4. Complete Cluster Mode Configuration

- Once all client applications have been migrated and validated, finalize the cluster mode by switching from Compatible to Enabled:

```
aws elasticache modify-replication-group \
  --replication-group-id <your-replication-group-id> \
  --cluster-mode enabled \
  --apply-immediately
```

- This enforces cluster mode fully, allowing scaling and other cluster features to be enabled.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4 Secure Configuration of Enterprise Assets and Software Establish and maintain the secure configuration of enterprise assets (end-user devices, including portable and mobile; network devices; non-computing/IoT devices; and servers) and software (operating systems and applications).			

5.12 Ensure ElastiCache is deployed across multiple Availability Zones (AZs) (Manual)

Profile Applicability:

- Level 2

Description:

Deploying Amazon ElastiCache across multiple Availability Zones means configuring the cache cluster nodes (primary and replicas) to be distributed in different AZs within the same AWS region. This multi-AZ deployment improves fault tolerance and availability by mitigating risks associated with failure or degradation in a single Availability Zone. If the primary node or an AZ becomes unavailable, ElastiCache can automatically fail over to a replica in a different AZ, minimizing downtime and data unavailability.

Rationale:

Distributing ElastiCache nodes across multiple AZs protects the caching layer from localized infrastructure failures, such as power outages, networking disruptions, or hardware faults in a single AZ.

Impact:

Enabling multi-AZ deployment for ElastiCache clusters enhances system availability and resiliency, significantly reducing the risk of cache service interruptions due to an AZ failure or node disruption.

Audit:

Audit All Replication Groups for Multi-AZ Status:

```
aws elasticache describe-replication-groups --query
"ReplicationGroups[*].{ID:ReplicationGroupId,MultiAZ:MultiAZ}"
```

This returns true if Multi-AZ with automatic failover is enabled, otherwise false.

Remediation:

1. Prerequisites for Enabling Multi-AZ on ElastiCache

- VPC with Subnets in Multiple Availability Zones: The VPC associated with your ElastiCache replication group must have at least two subnets in different Availability Zones within the same AWS Region.
- Cache Subnet Group Configuration: The cache subnet group used by the replication group must include multiple subnets spanning the desired AZs to support node placement.

- Replication Group with At Least One Replica: Multi-AZ requires a primary node and at least one read replica that can be deployed in a different AZ to support automatic failover.
- Automatic Failover Enabled: Failover between primary and replicas is automatic with Multi-AZ, so automatic failover must be enabled on the replication group (can be set at creation or modified later).

2. Modify the Replication Group to Enable Multi-AZ:

```
aws elasticache modify-replication-group \
  --replication-group-id <your-replication-group-id> \
  --multi-az-enabled \
  --apply-immediately
```

- This command enables Multi-AZ with automatic failover for the specified replication group.
- The --apply-immediately flag ensures the change happens without waiting for the next maintenance window. Use with caution in production.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4 Secure Configuration of Enterprise Assets and Software Establish and maintain the secure configuration of enterprise assets (end-user devices, including portable and mobile; network devices; non-computing/IoT devices; and servers) and software (operating systems and applications).			

5.13 Ensure ElastiCache has automatic backups enabled (Manual)

Profile Applicability:

- Level 1

Description:

Ensure that Amazon ElastiCache clusters that store critical or stateful data have automatic backups enabled with a non-zero retention period. This setting configures ElastiCache to take daily snapshots of caches and retain them for a defined number of days, allowing restoration of data in case of corruption, accidental deletion, or infrastructure failure.

Rationale:

Automatic backups provide a simple and reliable way to recover ElastiCache data without relying solely on application-level safeguards. In the event of node failure, misconfiguration, or data corruption, a recent backup snapshot can be used to create a new cache or replication group, significantly reducing recovery time and impact on dependent applications.

Impact:

Enabling automatic backups for ElastiCache ensures that cache data is regularly captured in snapshots, keeping it protected and readily recoverable in case of accidental deletion, corruption, or node failure. As a result, organizations can quickly recreate cache clusters from recent backups and restore access to the cached data, minimizing downtime and business impact from unexpected data loss events.

Audit:

1. List backup settings for all cache clusters (node-based):

```
aws elasticache describe-cache-clusters \
  --show-cache-node-info \
  --query
"CacheClusters[*].{Id:CacheClusterId,Engine:Engine,SnapshotRetentionLimit:Sna
pshotRetentionLimit}"
```

- SnapshotRetentionLimit > 0 ⇒ automatic backups enabled.
- SnapshotRetentionLimit = 0 or null ⇒ automatic backups disabled.

2. List backup settings for all replication groups (Redis/Valkey):

```
aws elasticache describe-replication-groups \
  --query
"ReplicationGroups[*].{Id:ReplicationGroupId,Engine:Engine,SnapshotRetentionL
imit:SnapshotRetentionLimit}"
```

- Again, treat SnapshotRetentionLimit = 0 as non-compliant for this control.

Remediation:

Enable backups on a replication group (Redis/Valkey):

```
aws elasticache modify-replication-group \
  --replication-group-id <replication-group-id> \
  --snapshot-retention-limit 7 \
  --snapshotting-cluster-id <primary-cache-cluster-id> \
  --apply-immediately
```

- replication-group-id is your replication group (e.g., my-redis-rg).
- primary-cache-cluster-id is the cache cluster ID that should be used as the daily snapshot source (often the primary node, like my-redis-rg-001).
- snapshot-retention-limit 7 sets a 7-day retention; choose a value (1–35 days) per your policy.
- Optionally set or adjust --preferred-maintenance-window or a specific --snapshot-window if supported for your engine/version.
- Use --apply-immediately for immediate effect; omit it to apply in the next maintenance window.
- You can find the cluster IDs with:

```
aws elasticache describe-cache-clusters \
  --show-cache-node-info \
  --query
"CacheClusters[*].{Id:CacheClusterId,ReplicationGroupId:ReplicationGroupId}"
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	11 Data Recovery Establish and maintain data recovery practices sufficient to restore in-scope enterprise assets to a pre-incident and trusted state.			

6 Amazon MemoryDB for Redis

Amazon MemoryDB for Redis is a managed, highly available, and durable Redis-compatible in-memory database service provided by Amazon Web Services (AWS). It is designed to offer a fully managed Redis experience with the additional benefits of high availability, durability, and ease of use.

Internal Only - General

6.1 Ensure Network Security is Enabled (Manual)

Profile Applicability:

- Level 1

Description:

Rationale:

Audit:

1. Create or Select a Virtual Private Cloud (VPC)

- Sign in to the AWS Management Console and open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
- Create a new VPC or select an existing VPC where you want to deploy your Amazon MemoryDB clusters.

2. Configure Subnets

- In the VPC console, navigate to `Subnets` in the left-side menu.
- Create or select the subnets within your VPC where you want to deploy your Amazon MemoryDB clusters.
- Ensure you have private subnets to isolate your MemoryDB clusters from the public internet.

3. Define Security Groups

- In the VPC console, navigate to `Security Groups` in the left-side menu.
- Create a new security group or select an existing one for your Amazon MemoryDB clusters.
- Configure inbound and outbound rules in the security group to control traffic access.
  - Allow inbound access only from trusted sources, such as specific IP ranges or security groups, on the necessary ports used by MemoryDB.
  - Define outbound rules based on your requirements, allowing outbound traffic to necessary destinations or ports.
- Associate the security group with your Amazon MemoryDB clusters.

4. Configure Network Access Control Lists (ACLs)

- In the VPC console, navigate to `Network ACLs` in the left-side menu.
- Create or select the network ACLs associated with the subnets used by your Amazon MemoryDB clusters.
- Configure inbound and outbound rules in the network ACLs to control traffic access.

    - Define rules based on your security requirements, allowing only necessary protocols, ports, and IP ranges.
    - Deny unnecessary or unwanted traffic.
- Associate the network ACLs with the subnets used by your Amazon MemoryDB clusters.

5. Configure VPC Endpoints

- In the VPC console, navigate to `Endpoints` in the left-side menu.
- Create or select the VPC endpoints required for Amazon MemoryDB.
    - If you need to access MemoryDB from within your VPC, create a VPC endpoint for Amazon MemoryDB to connect your applications securely.
    - If you need to access MemoryDB from another VPC or on-premises network, set up VPC peering or a transit gateway to establish a secure connection.

6. Verify Connectivity and Test

- Launch an Amazon EC2 instance within the same VPC and subnet as your Amazon MemoryDB clusters or use an existing one.
- Connect to the EC2 instance using SSH or other remote access methods.
- Test the connectivity to your Amazon MemoryDB clusters by trying to connect to them using the appropriate client or utility.
- Verify that the network security settings allow the necessary traffic and deny unauthorized access.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	12.2 Establish and Maintain a Secure Network Architecture Establish and maintain a secure network architecture. A secure network architecture must address segmentation, least privilege, and availability, at a minimum.		●	●

Controls Version	Control	IG 1	IG 2	IG 3
v7	11.7 Manage Network Infrastructure Through a Dedicated Network Manage the network infrastructure across network connections that are separated from the business use of that network, relying on separate VLANs or, preferably, on entirely different physical connectivity for management sessions for network devices.		●	●

6.2 Ensure Data at Rest and in Transit is Encrypted (Manual)

Profile Applicability:

- Level 1

Description:

Rationale:

Audit:

1. Sign in to the AWS Management Console

- Sign in to the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon MemoryDB Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/memorydb/.

3. Select the Cluster

- Choose the MemoryDB cluster for which you want to enable encryption at rest and in transit.
- Click on the cluster name to access its details page.

4. Enable Encryption at Rest

- In the cluster details page, navigate to the `Encryption at Rest` section.
- Click on `Modify` to edit the encryption settings.
- Select the desired encryption option:
  - AWS Managed Key (Default): Choose this option to use the default AWS managed key for encryption at rest. Amazon MemoryDB automatically encrypts your data using this key.
  - Customer Managed Key (CMK): Choose this option if you want to use your own AWS Key Management Service (KMS) customer-managed key for encryption. Select the appropriate CMK from the dropdown menu.
- Click "Apply Changes" to enable encryption at rest for the MemoryDB cluster.

5. Enable Encryption in Transit

- In the cluster details page, navigate to the `Encryption in Transit` section.
- Click on `Modify` to edit the encryption settings.
- Select the desired encryption option:

  - Encryption in Transit Enabled: Choose this option to enable encryption in transit for data transmitted between your client applications and MemoryDB. MemoryDB uses SSL/TLS encryption to secure the communication channel.
  - Encryption in Transit Disabled: Choose this option if you do not require encryption in transit.
- Click `Apply Changes` to enable encryption in transit for the MemoryDB cluster.

6. Verify Encryption Status

- Wait a few minutes for the changes to propagate and the encryption settings to take effect.
- Refresh the cluster details page to see the updated encryption status.
- Verify that encryption at rest and in transit are enabled for the MemoryDB cluster.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.10 Encrypt Sensitive Data in Transit Encrypt sensitive data in transit. Example implementations can include: Transport Layer Security (TLS) and Open Secure Shell (OpenSSH).		●	●
v8	3.11 Encrypt Sensitive Data at Rest Encrypt sensitive data at rest on servers, applications, and databases containing sensitive data. Storage-layer encryption, also known as server-side encryption, meets the minimum requirement of this Safeguard. Additional encryption methods may include application-layer encryption, also known as client-side encryption, where access to the data storage device(s) does not permit access to the plain-text data.		●	●
v7	14.4 Encrypt All Sensitive Information in Transit Encrypt all sensitive information in transit.		●	●
v7	14.8 Encrypt Sensitive Information at Rest Encrypt all sensitive information at rest using a tool that requires a secondary authentication mechanism not integrated into the operating system, in order to access the information.			●

6.3 Ensure Authentication and Access Control is Enabled (Manual)

Profile Applicability:

- Level 1

Description:

Rationale:

Users should select whether they like to enable authentication. If they want to authenticate a password would be required, which would only allow the authorized person to access the cluster. Defining access control allows specific workers in a business access to the database.

Impact:

Allowing authentication verifies the identity of the person and who has appropriate access to a company's data.

Audit:

1. Sign into the AWS Management Console

- Sign into the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon MemoryDB Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/memorydb/.

3. Select the Cluster

- Choose the Amazon MemoryDB cluster on which you want to implement authentication and access control.
- Click on the cluster name to access its details page.

4. Enable Authentication

- In the cluster details page, navigate to the `Authentication` section.
- Click on `Modify` to edit the authentication settings.
- Select the desired authentication option:
  - No Authentication: This option allows unauthenticated access to your MemoryDB cluster.

  - Password Authentication: Choose this option to enable password-based authentication. Enter the desired password for the cluster.
- Click `Apply Changes` to enable authentication for the MemoryDB cluster.

5. Define Access Control Policies

- In the cluster details page, navigate to the "Access Control" section.
- Click on `Modify` to edit the access control settings.
- Define the access control policies based on your requirements:
  - For Redis-based clusters, you can use Redis Access Control Lists (ACLs) to control access at the Redis command level.
  - Use the Redis commands to create, modify, or delete ACL rules as needed.
  - You can define rules based on IP addresses, users, or patterns to allow or deny specific commands or operations.
- Click `Apply Changes` to save the access control policies for the MemoryDB cluster.

6. Test Authentication and Access Control

- Use a Redis client or utility to connect to your Amazon MemoryDB cluster.
- Provide the necessary authentication credentials, such as the password, if password-based authentication is enabled.
- Test the connection and verify that you can access the MemoryDB cluster based on the defined access control policies.

7. Regularly Review and Update Access Control

- Periodically review the access control policies to ensure they align with your security requirements.
- Update the ACL rules, passwords, or other authentication mechanisms to adapt to changing access requirements or security policies.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

Internal Only - General

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

6.4 Ensure Audit Logging is Enabled (Manual)

Profile Applicability:

- Level 1

Description:

Enabling audit logging on Amazon MemoryDB allows you to capture and store logs of activities performed on your clusters.

Rationale:

It captures and saves logs of activities that took place in the cluster.

Impact:

Reduces risks of any fraud since worker activity is being monitored and tracked.

Audit:

1. Sign into the AWS Management Console

- Sign into the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon MemoryDB Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/memorydb/.

3. Select the Cluster

- Choose the MemoryDB cluster for which you want to enable audit logging. Click on the cluster name to access its details page.

4. Enable Amazon CloudWatch Logs

- In the cluster details page, navigate to the `Logging` section.
- Click on `Modify` to edit the logging settings.
- Select the option to enable CloudWatch Logs.
- Choose an existing CloudWatch log group or create a new one to store the logs generated by the MemoryDB cluster activities.
- Optionally, you can specify a log retention period to define how long the logs will be stored.
- Click `Apply Changes` to enable CloudWatch Logs for the MemoryDB cluster.

5. Configure CloudWatch Logs

Internal Only - General

- Open the CloudWatch console by navigating to `CloudWatch` in the AWS Management Console.
- In the left-side menu, click on `Logs`.
- Create a new log group or select an existing log group that will store the MemoryDB logs.
- Configure log retention settings based on your retention requirements. Logs can be stored for a specific number of days or indefinitely.
- Define any necessary log group permissions to control access to the logs.
- Optionally, set up log exports or alarms for specific log events or patterns if needed.

6. Verify Logging Status

- Wait a few minutes for the changes to propagate and the logging configuration to take effect.
- Refresh the cluster details page to see the updated logging status.
- Verify that CloudWatch Logs is enabled for the MemoryDB cluster.

7. Monitor and Analyze Logs

- Navigate to the CloudWatch console and select the log group that stores the MemoryDB logs.
- Monitor the logs to gain insights into the activities and operations performed on your MemoryDB cluster.
- Use CloudWatch Logs features, such as log searching, filtering, and visualization, to analyze the logs and identify any security or operational issues.
- Establish appropriate log monitoring and alerting mechanisms to proactively identify and respond to potential security incidents or operational anomalies.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	8.1 Establish and Maintain an Audit Log Management Process Establish and maintain an audit log management process that defines the enterprise's logging requirements. At a minimum, address the collection, review, and retention of audit logs for enterprise assets. Review and update documentation annually, or when significant enterprise changes occur that could impact this Safeguard.	●	●	●

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●

Internal Only - General

6.5 Ensure Security Configurations are Reviewed Regularly (Manual)

Profile Applicability:

- Level 1

Description:

This helps by removing or updating any IAM roles, security networks, encryption settings, audit logging, and authentication. By updating or removing a few things from these lists it helps tighten security and ensures that the users do not have excessive permissions.

Rationale:

Impact:

By regularly checking these settings in the database the user is preventing the database from a cyber threat.

Audit:

1. Sign into the AWS Management Console

- Sign into the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon MemoryDB Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/memorydb/.

3. Select the Cluster

- Choose the Amazon MemoryDB cluster for which you want to update and review the security configuration.
- Click on the cluster name to access its details page.

4. Review IAM Roles and Permissions

- In the cluster details page, navigate to the `Security` or `Access Control` section.
- Review the IAM roles and permissions associated with the cluster.
- Ensure that the IAM roles have appropriate permissions and follow the principle of least privilege.

- Review the IAM policies and make any necessary updates to align with your security requirements.

5. Review Network Security

- In the cluster details page, navigate to the `Security` or `Network & Security` section.
- Review the Virtual Private Cloud (VPC), subnets, security groups, and network ACLs associated with the cluster.
- Ensure that the VPC and subnet configurations align with your security requirements.
- Review the security group rules and network ACL rules to ensure they restrict access to necessary ports, IP ranges, and protocols.
- Make any necessary updates to tighten the network security settings.

6. Review Encryption Settings

- In the cluster details page, navigate to the `Security` or `Encryption` section.
- Review the encryption settings for the cluster.
- Ensure that encryption at rest and in transit are enabled and the appropriate encryption options are chosen.
- Review any customer-managed keys used for encryption and verify their configurations.

7. Review Authentication and Access Control

- In the cluster details page, navigate to the `Security` or `Access Control` section.
- Review the authentication options and access control policies in place for the cluster.
- Ensure that the authentication mechanisms and access control policies align with your security requirements.
- Make any necessary updates to adapt to changing access requirements or security policies.

8. Review Audit Logging

- In the cluster details page, navigate to the `Monitoring` or `Logging` section.
- Review the logging configuration for the cluster.
- Ensure that the logs are captured and stored as expected.
- Please review the log retention settings and verify that they comply with your retention policies.

9. Regularly Monitor Security Bulletins

- Stay updated with AWS security bulletins, advisories, and best practices.

- Monitor AWS security announcements and subscribe to relevant security notifications.
- Regularly review and apply security patches, updates, and recommended configuration changes for Amazon MemoryDB.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	5 Account Management Use processes and tools to assign and manage authorization to credentials for user accounts, including administrator accounts, as well as service accounts, to enterprise assets and software.			
v7	5 Secure Configuration for Hardware and Software on Mobile Devices, Laptops, Workstations and Servers Secure Configuration for Hardware and Software on Mobile Devices, Laptops, Workstations and Servers			

6.6 Ensure Monitoring and Alerting is Enabled (Manual)

Profile Applicability:

- Level 1

Description:

Implementing monitoring and alerting on Amazon MemoryDB allows you to proactively detect and respond to any performance issues, security events, or operational anomalies.

Rationale:

This helps in ensuring that everything in the system is secure and if there is an unusual activity that takes place it addresses the issues quickly and efficiently.

Impact:

Enabling monitoring and alerting has a positive impact in the business operations when the issue is identified and addressed accordingly.

Audit:

1. Sign into the AWS Management Console

- Sign into the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon MemoryDB Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/memorydb/.

3. Select the Cluster

- Choose the Amazon MemoryDB cluster for which you want to implement monitoring and alerting. Click on the cluster name to access its details page.

4. Enable Amazon CloudWatch

- In the cluster details page, navigate to the `Monitoring` or `CloudWatch` section.
- Click on `Enable` to enable CloudWatch monitoring for the cluster.
- Select the appropriate CloudWatch metric categories to monitor, such as CPU utilization, memory utilization, network traffic, and storage capacity.
- Configure the desired granularity and period for metric collection.
- Click `Enable` or `Save` to enable CloudWatch monitoring for the cluster.

5. Set Up CloudWatch Alarms

- In the CloudWatch console, navigate to `Alarms` in the left-side menu.
- Click on `Create Alarm` to set up a new alarm.
- Select the CloudWatch metric related to the aspect you want to monitor, such as CPU utilization or memory utilization.
- Configure the alarm threshold based on your desired criteria, such as setting CPU utilization above a certain percentage.
- Define the actions to be taken when the alarm is triggered.
- Click `Create Alarm` to create the CloudWatch alarm.

6. Configure Amazon EventBridge Rules (Optional)

- In the Amazon EventBridge console, navigate to `Rules` in the left-side menu.
- Click on `Create rule` to set up a new rule.
- Define the event pattern or source that should trigger the rule, such as specific MemoryDB events or errors.
- Configure the target actions, such as sending notifications, executing AWS Lambda functions, or invoking AWS Step Functions.
- Click `Create` to create the EventBridge rule.

7. Configure Auto Scaling (Optional)

- In the MemoryDB cluster details page, navigate to the `Auto Scaling` section.
- Configure auto-scaling settings based on your workload and performance requirements.
- Define the scaling policies, such as increasing or decreasing the number of replica nodes based on CPU utilization or other metrics.
- Set the desired minimum and maximum number of replica nodes for the cluster.
- Click `Save` or `Apply Changes` to apply the auto-scaling configuration.

8. Regularly Review and Adjust Monitoring and Alarms

- Periodically review the CloudWatch metrics and alarms to ensure they align with your monitoring needs and performance expectations.
- Adjust the thresholds and actions based on changing workload patterns or performance requirements.
- Stay informed about new CloudWatch features and best practices to optimize your monitoring setup.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	8.1 Establish and Maintain an Audit Log Management Process Establish and maintain an audit log management process that defines the enterprise’s logging requirements. At a minimum, address the collection, review, and retention of audit logs for enterprise assets. Review and update documentation annually, or when significant enterprise changes occur that could impact this Safeguard.	●	●	●
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●

Internal Only - General

6.7 Ensure MemoryDB has automatic backups enabled (Manual)

Profile Applicability:

- Level 1

Description:

Ensure that Amazon MemoryDB clusters that store critical or stateful data have automatic backups enabled with a non-zero retention period. This setting configures MemoryDB to take snapshots of database and retain them for a defined number of days, allowing restoration of data in case of corruption, accidental deletion, or infrastructure failure.

Rationale:

Automatic backups provide a simple and reliable way to recover MemoryDB data without relying solely on application-level safeguards. In the event of node failure, misconfiguration, or data corruption, a recent backup snapshot can be used to create a new database, significantly reducing recovery time and impact on dependent applications.

Impact:

Enabling automatic backups for MemoryDB ensures that cluster data is regularly captured in snapshots, keeping it protected and readily recoverable in case of accidental deletion, corruption, or node failure. As a result, organizations can quickly recreate clusters from recent backups and restore access to the data, minimizing downtime and business impact from unexpected data loss events.

Audit:

List snapshot settings for all memorydb clusters

```
aws memorydb describe-clusters \
  --query
"Clusters[*].{Name:Name,SnapshotRetentionLimit:SnapshotRetentionLimit}"
```

- SnapshotRetentionLimit > 0 ⇒ automatic snapshots enabled.
- SnapshotRetentionLimit = 0 or missing/null ⇒ automatic snapshots disabled.

Remediation:

Enable automatic snapshots on a specific cluster

```
aws memorydb update-cluster \
  --cluster-name <cluster-name> \
  --snapshot-retention-limit 7 \
  --snapshot-window "03:00-04:00"
```

- --snapshot-retention-limit 7 configures a 7-day retention; use a value aligned with your policy (for example 7, 14, or 30 days).
- --snapshot-window "03:00-04:00" (optional) sets the daily snapshot window in UTC. Choose an off-peak period to minimize performance impact.
- MemoryDB applies these changes immediately for the cluster configuration.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	11 Data Recovery Establish and maintain data recovery practices sufficient to restore in-scope enterprise assets to a pre-incident and trusted state.			

Internal Only - General

7 Amazon DocumentDB

Internal Only - General

7.1 Ensure Network Architecture Planning (Manual)

Profile Applicability:

- Level 1

Description:

Plan the network architecture to isolate your DocumentDB instances within a secure Virtual Private Cloud (VPC). Configure appropriate security groups and network access control lists (ACLs) to control inbound and outbound traffic to your DocumentDB instances.

Rationale:

Depending on how the network is established between devices, which then helps secure data when transferring it from one server to another.

Impact:

The way the users design their network sets the performance for the system and how it would interact with servers.

Audit:

1. Understand Amazon VPC Basics

- Familiarize yourself with Amazon Virtual Private Cloud (VPC) and its concepts.
- Learn about VPC components, including subnets, route tables, and security groups.

2. Determine VPC Requirements for DocumentDB

- Identify the specific networking requirements for your Amazon DocumentDB deployment.
- Consider factors such as network availability, fault tolerance, and security requirements.

3. Create a New VPC or Use an Existing VPC

- Decide whether to create a new VPC dedicated to Amazon DocumentDB or use an existing VPC.
- If creating a new VPC, follow the steps to create a VPC in the AWS Management Console.

4. Configure Subnets

Internal Only - General

- Determine the number and size of subnets needed for your DocumentDB deployment.
- Create the required subnets within your VPC, ensuring proper availability zone distribution.

5. Set Up Routing

- Configure the route tables associated with your subnets.
- Ensure that the route tables have the necessary routes for proper network connectivity.

6. Configure Security Groups

- Create or configure security groups to control inbound and outbound traffic to your DocumentDB instances.
- Define the necessary inbound rules to allow access from authorized sources.

7. Plan Connectivity Options

- Decide how your DocumentDB instances will connect to your VPC and other resources.
- Determine if you need to set up VPC peering, VPN connections, or AWS Direct Connect for connectivity.

8. Consider High Availability and Fault Tolerance

- Evaluate your requirements for high availability and fault tolerance.
- Design your network architecture to ensure that DocumentDB instances are deployed across multiple availability zones for resilience.

9. Implement Network Access Control

- Consider using network access control lists (ACLs) to provide an additional layer of security.
- Configure the ACLs to allow only necessary traffic and block unauthorized access.

10. Test and Validate the Network Architecture

- Ensure that your network architecture is correctly configured and meets your requirements.
- Test connectivity and verify that DocumentDB instances can be accessed securely.

Internal Only - General

Remediation:

To establish connection, the users would need to factor in their virtual private cloud (VPC), create subnet, configure routing, and implement ACLs.

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	12.2 Establish and Maintain a Secure Network Architecture Establish and maintain a secure network architecture. A secure network architecture must address segmentation, least privilege, and availability, at a minimum.		●	●
v7	11.7 Manage Network Infrastructure Through a Dedicated Network Manage the network infrastructure across network connections that are separated from the business use of that network, relying on separate VLANs or, preferably, on entirely different physical connectivity for management sessions for network devices.		●	●

7.2 Ensure VPC Security is Configured (Manual)

Profile Applicability:

- Level 1

Description:

Creating a VPC, configuring subnets, and creating security groups help isolate your DocumentDB instances within your virtual network and control inbound and outbound traffic.

Rationale:

Setting up a Virtual Private Cloud (VPC) protects the private network that has been established from any external networks from interfering. It allows internal networks to communicate with one another with the network that has been established.

Impact:

Builds a strong connection between internal networks, has a strong connection with the internet, and it secures your data from getting into the hands of an unauthorized party.

Audit:

1. Sign into the AWS Management Console

- Sign into the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon VPC Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/vpc/.

3. Create a VPC (Virtual Private Cloud)

- Click on the `Create VPC` button to create a new VPC.
- Provide the necessary details, such as VPC name, CIDR block, and additional configuration options.
- Click on `Create` to create the VPC.

4. Configure VPC Subnets

- Once the VPC is created, navigate to the `Subnets` section in the VPC console.
- Click on the `Create subnet` button to create a new subnet.
- Provide the necessary details, such as subnet name, VPC selection, and subnet CIDR block.

- Repeat this step to create multiple subnets within your VPC, if required.

5. Create Security Groups

- Navigate to the `Security Groups` section in the VPC console.
- Click the `Create security group` button to create a new security group.
- Provide a name and description for the security group.
- Configure inbound and outbound rules to allow the necessary traffic to and from the DocumentDB instances.
- Repeat this step to create additional security groups if needed.

6. Launch Amazon DocumentDB Cluster in VPC

- Navigate to the service using the "Find Services" search bar or by directly accessing the console at https://console.aws.amazon.com/docdb/.
- Click on `Create database` to create a new DocumentDB cluster.
- Configure the necessary parameters, such as cluster name, instance specifications, and storage options.
- In the `Network & Security` section, select the VPC and subnets you created earlier.
- Choose the appropriate security group(s) to apply to the DocumentDB cluster.
- Click `Create` to launch the DocumentDB cluster in the configured VPC.

7. Test Connectivity

- Once the DocumentDB cluster is launched, validate that you can connect to it from authorized sources.
- Use the appropriate database client or tools to establish a connection and verify connectivity.

8. Monitor and Update Security Groups

- Regularly monitor and update the security groups associated with the DocumentDB cluster.
- Modify the inbound and outbound rules to ensure appropriate access control and security.

Remediation:

The individual is required to create a subnet and configure their inbound and outbound access. Individuals are supposed to configure and route, ensuring the traffic is flowing smoothly without any interference. This control is important because it only allows authorized users to access their resources as they prefer.

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	12.2 Establish and Maintain a Secure Network Architecture Establish and maintain a secure network architecture. A secure network architecture must address segmentation, least privilege, and availability, at a minimum.		●	●
v7	11.7 Manage Network Infrastructure Through a Dedicated Network Manage the network infrastructure across network connections that are separated from the business use of that network, relying on separate VLANs or, preferably, on entirely different physical connectivity for management sessions for network devices.		●	●

7.3 Ensure Encryption at Rest is Enabled (Manual)

Profile Applicability:

- Level 1

Description:

Rationale:

This helps ensure that the data is kept secure and protected when at rest. The user must choose from two key options which then determine when the data is encrypted at rest.

Impact:

If an unauthorized user steals the data, it would be unreadable for them because a key would be required to decrypt the message into plaintext.

Audit:

1. Sign into the AWS Management Console

- Sign into the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon DocumentDB Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/docdb/.

3. Select the DocumentDB Cluster

- Choose the Amazon DocumentDB cluster for which you want to enable encryption at rest.
- Click on the cluster name to access its details page.
- In the cluster details page, navigate to the "Configuration" section.

4. Enable Encryption at Rest

- Under the `Storage` section.
- Click on the "Edit" button or "Modify" option to configure the encryption settings.
- Choose the option to enable encryption at rest for the cluster.

5. Choose the Encryption Key

- Select the AWS Key Management Service (KMS) key that you want to use for encrypting your DocumentDB data.

- You can choose an existing KMS key or create a new one.
- Ensure that the KMS key you select has appropriate permissions for DocumentDB to use it.

6. Save the Configuration

- Click the `Save` button to apply the encryption at rest configuration.
- DocumentDB will start the process of encrypting the existing data and all new data written to the cluster.

7. Verify Encryption Status

- Monitor the cluster status to ensure that the encryption process is completed successfully.
- Once the encryption is enabled, the cluster status will reflect the updated encryption status.

8. Test Connectivity

- Validate that you can still connect to the DocumentDB cluster after enabling encryption at rest.
- Ensure that your applications and authorized users can access the encrypted data.

9. Monitor and Manage Encryption

- Regularly monitor the encryption status of your DocumentDB cluster.
- Ensure that the encryption remains enabled and that no unauthorized modifications are made.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

Internal Only - General

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.11 Encrypt Sensitive Data at Rest Encrypt sensitive data at rest on servers, applications, and databases containing sensitive data. Storage-layer encryption, also known as server-side encryption, meets the minimum requirement of this Safeguard. Additional encryption methods may include application-layer encryption, also known as client-side encryption, where access to the data storage device(s) does not permit access to the plain-text data.		●	●
v7	14.8 Encrypt Sensitive Information at Rest Encrypt all sensitive information at rest using a tool that requires a secondary authentication mechanism not integrated into the operating system, in order to access the information.			●

Internal Only - General

7.4 Ensure Encryption in Transit is Enabled (Manual)

Profile Applicability:

- Level 1

Description:

Rationale:

Amazon Database DB uses SSL/TLS to encrypt data during transit. To secure your data in transit the individual should identify their client application and what is supported by TLS to configure it correctly.

Audit:

1. Sign into the AWS Management Console

- Sign into the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon DocumentDB Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/docdb/.

3. Select the DocumentDB Cluster

- Choose the Amazon DocumentDB cluster for which you want to enable encryption in transit.
- Click on the cluster name to access its details page.
- In the cluster details page, navigate to the "Configuration" section.

4. Enable Encryption in Transit

- Under the `Network & Security` section.
- Click on the `Edit` button or `Modify` option to configure the encryption settings.
- Enable the option for encryption in transit by choosing the appropriate setting.
- Note that encryption in transit uses SSL/TLS to secure communications between your applications and the DocumentDB cluster.

5. Save the Configuration

- Click on the "Save" button to apply the encryption in transit configuration.
- DocumentDB will automatically handle the SSL/TLS encryption for network traffic between clients and the cluster.

Internal Only - General

6. Validate Encryption in Transit

- Test the connectivity to your DocumentDB cluster from your applications or clients.
- Ensure that the communication is established securely using SSL/TLS encryption.

7. Monitor and Maintain Encryption in Transit

- Regularly monitor the encryption in transit configuration for your DocumentDB cluster.
- Stay informed about updates or changes in SSL/TLS protocols and encryption standards.
- Keep your client applications current to ensure they support the latest encryption protocols.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.10 Encrypt Sensitive Data in Transit Encrypt sensitive data in transit. Example implementations can include: Transport Layer Security (TLS) and Open Secure Shell (OpenSSH).		●	●
v7	14.4 Encrypt All Sensitive Information in Transit Encrypt all sensitive information in transit.		●	●

7.5 Ensure to Implement Access Control and Authentication (Manual)

Profile Applicability:

- Level 1

Description:

Configure authentication mechanisms for your DocumentDB instances, such as using AWS Identity and Access Management (IAM) users or database users. Define appropriate user roles and permissions to control access to the DocumentDB instances and databases.

Rationale:

Audit:

1. Sign into the AWS Management Console

- Sign into the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon DocumentDB Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/docdb/.

3. Select the DocumentDB Cluster

- Choose the Amazon DocumentDB cluster for which you want to implement access control and authentication.
- Click on the cluster name to access its details page.
- In the cluster details page, navigate to the "Configuration" section.

4. Enable Authentication

- Under the `Network & Security` section.
- Click on the `Edit` button or `Modify` option to configure the authentication settings.
- Enable the option for authentication by choosing the appropriate setting.
- DocumentDB supports authentication through username and password or through AWS Identity and Access Management (IAM) roles.

5. Configure Database Users

- In the cluster details page, navigate to the `Users` or `Database users` section.

- Click the `Add user` button to create a new database user.
- Enter the username and password for the database user.
- Assign appropriate permissions to the user, such as read-only or read-write access to specific databases or collections.

6. Save the Configuration

- Click on the `Save` button to apply the authentication and access control configuration.
- DocumentDB will enforce authentication for connections to the cluster.

7. Test Authentication

- Validate that your client applications or tools can connect to the DocumentDB cluster using the configured authentication credentials.
- Ensure that the authentication process is successfully completed.

8. Monitor and Manage Access Control

- Regularly monitor and manage the access control configuration for your DocumentDB cluster.
- Review and update the permissions assigned to database users as needed.
- Remove any unnecessary or unused database users to minimize security risks.

9. Consider IAM Authentication (Optional)

- If desired, you can also configure IAM authentication for your DocumentDB cluster.
- Follow the AWS documentation to set up IAM authentication for DocumentDB, if applicable.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●

Controls Version	Control	IG 1	IG 2	IG 3
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

7.6 Ensure Audit Logging is Enabled (Manual)

Profile Applicability:

- Level 1

Description:

Enable audit logging to capture database activities, including login attempts, queries, and modifications. Send the logs to Amazon CloudWatch or a centralized log management system for analysis and monitoring.

Rationale:

It captures and saves logs of activities that took place in the cluster, by recording login attempts, queries, and any changes within the database.

Audit:

1. Sign into the AWS Management Console

- Sign into the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon DocumentDB Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/docdb/.

3. Select the DocumentDB Cluster

- Choose the Amazon DocumentDB cluster for which you want to enable audit logging.
- Click on the cluster name to access its details page.
- In the cluster details page, navigate to the "Configuration" section.

4. Enable Audit Logging

- Under the `Database options` or `Database features` section.
- Click on the `Edit` button or `Modify` option to configure the audit logging settings.
- Enable the option for audit logging by choosing the appropriate setting.
- Specify the destination for the audit logs, which can be an Amazon CloudWatch Logs group or an Amazon S3 bucket.

5. Configure Audit Log Destination

- If you choose to send audit logs to an Amazon CloudWatch Logs group, select the existing group or create a new one.

Internal Only - General

- If you choose to send audit logs to an Amazon S3 bucket, select the existing bucket or create a new one. Provide the necessary permissions for DocumentDB to write logs to the bucket.

6. Set Audit Log Retention Period

- Specify the retention period for the audit logs, indicating how long the logs should be retained in the selected destination.
- Consider your compliance and regulatory requirements when determining the retention period.

7. Save the Configuration

   Click on the `Save` button to apply the audit logging configuration. DocumentDB will start recording audit logs according to the configured settings.

8. Validate Audit Logging

- Perform operations on your DocumentDB cluster to generate audit log events.
- Verify that the audit logs are recorded and sent to the specified destination.
- Review the logs to ensure they contain the expected information and events.

9. Monitor and Analyze Audit Logs

- Use Amazon CloudWatch Logs or other log analysis tools to monitor and analyze the audit logs generated by DocumentDB.
- Set up log metrics, alarms, and notifications to detect unusual activities or security incidents.
- Review audit logs regularly to identify potential security threats, compliance violations, or unauthorized access attempts.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	8.1 Establish and Maintain an Audit Log Management Process Establish and maintain an audit log management process that defines the enterprise’s logging requirements. At a minimum, address the collection, review, and retention of audit logs for enterprise assets. Review and update documentation annually, or when significant enterprise changes occur that could impact this Safeguard.	●	●	●
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●

7.7 Ensure Regular Updates and Patches (Manual)

Profile Applicability:

- Level 1

Description:

Stay informed about the latest security updates and patches released by Amazon for DocumentDB. Regularly apply updates and patches to your DocumentDB instances to protect against known vulnerabilities.

Rationale:

Impact:

Helps the organization reduce their security risk by regularly updating and patching their database and database engine. Regularly updating and scanning for any weaknesses in the company can bring up possible vulnerabilities that could have led to potential cyber-attack.

Audit:

1. Stay Informed

- Stay updated with Amazon DocumentDB announcements, release notes, and security bulletins.
- Subscribe to AWS newsletters, forums, and notifications to receive timely updates regarding updates and patches.

2. Plan for Maintenance Windows

- Determine a suitable maintenance window to apply updates and patches to your DocumentDB cluster.
- Consider the impact on your applications and users when scheduling the maintenance window.

3. Monitor the AWS Management Console

- Regularly check the AWS Management Console for notifications related to available updates and patches for your DocumentDB cluster.
- The console will provide information on new versions and available patches.

4. Review the Release Notes and Changelog

- Before applying any updates or patches, review the release notes and changelog for the new version or patch.

Internal Only - General

- Pay attention to any compatibility or breaking changes that may require application adjustments.

5. Create a Test Environment (Optional)

- If feasible, create a separate test environment that closely resembles your production environment.
- Deploy a copy of your DocumentDB cluster in the test environment to test the updates and patches before applying them to production.

6. Apply Updates and Patches

- During the scheduled maintenance window, initiate the process to apply updates and patches to your DocumentDB cluster.
- Follow the recommended procedure provided by AWS, which may involve a few simple clicks in the AWS Management Console.
- Ensure that you select the appropriate version or patch to apply.

7. Monitor the Update Process

- Monitor the progress of the update or patch application for your DocumentDB cluster.
- AWS will provide status updates during the process to keep you informed.

8. Verify Post-Update Functionality

- After the update or patch is applied, test the functionality of your applications that rely on the DocumentDB cluster.
- Verify that your applications are working as expected and that any integration or dependencies are intact.

9. Review and Update Documentation

- Update your documentation, including standard operating procedures (SOPs), to reflect the new version or patch applied to the DocumentDB cluster.
- Document any changes or considerations specific to the update or patch.

10. Monitor for New Updates

- Continuously monitor for new updates and patches released by AWS for DocumentDB.
- Repeat the update process regularly to ensure your DocumentDB cluster remains up to date with the latest security enhancements and bug fixes.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	7 Continuous Vulnerability Management Develop a plan to continuously assess and track vulnerabilities on all enterprise assets within the enterprise's infrastructure, in order to remediate, and minimize, the window of opportunity for attackers. Monitor public and private industry sources for new threat and vulnerability information.			
v7	3 Continuous Vulnerability Management Continuous Vulnerability Management			

7.8 Ensure to Implement Monitoring and Alerting (Manual)

Profile Applicability:

- Level 1

Description:

This helps by alerting the system if any unusual event has occurred or if a particular threshold has been achieved because the user is able to set a desired interval or the cluster. This then allows system administrators to swiftly correct the situation and avoid subsequent complications if something unusual is happening.

Rationale:

Impact:

Has a positive impact in the business operations when the issue is identified and addressed accordingly.

Audit:

1. Sign into the AWS Management Console

- Sign into the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon DocumentDB Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/docdb/.

3. Select the DocumentDB Cluster

- Choose the Amazon DocumentDB cluster for which you want to implement monitoring and alerting.
- Click on the cluster name to access its details page.
- In the cluster details page, navigate to the "Monitoring" section.

4. Enable Enhanced Monitoring

- Under the `Enhanced Monitoring` section.
- Click on the `Edit` button or `Modify` option to configure enhanced monitoring settings.
- Enable the desired metrics and set the desired monitoring interval for the cluster.
- Enhanced monitoring provides additional insights into the performance and health of your DocumentDB cluster.

Internal Only - General

5. Set Up CloudWatch Alarms

- Scroll down to the `CloudWatch Alarms` section.
- Click on the `Create alarm` button.
- Configure the CloudWatch alarm based on the metrics you want to monitor and the thresholds you want to set.
- Specify the actions to be taken when the alarm is triggered, such as sending notifications or executing automated actions.

6. Customize Metrics and Dashboards (Optional)

- If desired, you can customize the metrics and dashboards in Amazon CloudWatch to suit your specific monitoring requirements.
- Create custom metrics, build personalized dashboards, and set up alarms based on your application's unique needs.

7. Test Monitoring and Alerting

- Perform operations on your DocumentDB cluster to generate metric data and trigger the configured alarms.
- Verify that CloudWatch is capturing the metrics and triggering the appropriate actions based on your alarm settings.

8. Regularly Review and Fine-Tune

- Regularly review the monitoring metrics, CloudWatch alarms, and any event-driven actions triggered by DocumentDB events.
- Fine-tune the monitoring settings, alarms, and notifications based on the observed patterns and requirements of your application.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	8.1 Establish and Maintain an Audit Log Management Process Establish and maintain an audit log management process that defines the enterprise's logging requirements. At a minimum, address the collection, review, and retention of audit logs for enterprise assets. Review and update documentation annually, or when significant enterprise changes occur that could impact this Safeguard.	●	●	●
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●

7.9 Ensure to Implement Backup and Disaster Recovery (Manual)

Profile Applicability:

- Level 1

Description:

Set up automated backups for your DocumentDB instances to ensure data durability and recoverability. Consider implementing a disaster recovery plan that includes data replication across different availability zones or regions.

Rationale:

Having the data backed up ensures that all the crucial information is stored securely it defends against any human errors and system errors that resulted in data loss. An organization that has a disaster recovery plan is prepared for any disruption that would impact business operations.

Impact:

If a business does not have a backup and recovery plan it would have a negative impact on the business, which would result in less productivity, suffer data loss that cannot be restored, and loss of revenue.

Audit:

1. Sign into the AWS Management Console

- Sign into the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon DocumentDB Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/docdb/.

3. Select the DocumentDB Cluster

- Choose the Amazon DocumentDB cluster for which you want to implement backup and disaster recovery.
- Click on the cluster name to access its details page.
- In the cluster details page, navigate to the "Backup" section.

4. Enable Automated Backups

- Under the `Automated backups` section.

- Click on the `Edit` button or `Modify` option to configure automated backup settings.
- Enable automated backups by choosing the desired backup retention period.
- Specify the number of days for which automated backups should be retained.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	11 Data Recovery Establish and maintain data recovery practices sufficient to restore in-scope enterprise assets to a pre-incident and trusted state.			
v7	10 Data Recovery Capabilities Data Recovery Capabilities			

Internal Only - General

7.10 Ensure to Configure Backup Window (Manual)

Profile Applicability:

- Level 1

Description:

Rationale:

Audit:

1. Perform Manual Backups (Optional)

- If desired, you can also create manual backups of your DocumentDB cluster.
- In the cluster details page, navigate to the `Backup` section.
- Click on the `Create backup` button.
- Provide a name for the backup and confirm the action.

2. Restore from Backups (Optional)

- If a disaster occurs or you need to restore your DocumentDB cluster to a previous state, you can restore it from the available backups.
- In the cluster details page, navigate to the `Backup` section.
- Choose the backup from which you want to restore.
- Follow the prompts and provide the necessary information to initiate the restore process.

3. Test Backup and Restore Procedures

- Periodically test the backup and restore procedures to ensure they work as expected.
- Perform test restores on non-production environments to validate the integrity and completeness of the backup data.

4. Regularly Monitor and Validate Backups

- Regularly monitor the backup status and validate that the backups are completed successfully.
- Monitor backup storage usage to ensure it is within the desired limits and plan for additional storage as needed.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

Internal Only - General

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	11.1 Establish and Maintain a Data Recovery Process Establish and maintain a data recovery process. In the process, address the scope of data recovery activities, recovery prioritization, and the security of backup data. Review and update documentation annually, or when significant enterprise changes occur that could impact this Safeguard.	●	●	●
v7	10.2 Perform Complete System Backups Ensure that each of the organization's key systems are backed up as a complete system, through processes such as imaging, to enable the quick recovery of an entire system.	●	●	●

7.11 Ensure to Conduct Security Assessments (Manual)

Profile Applicability:

- Level 1

Description:

Periodically perform security assessments, including vulnerability assessments and penetration testing, to identify and address any security weaknesses. Review your security configuration against best practices and industry standards.

Rationale:

This helps ensure that any vulnerabilities that might lie dormant be addressed promptly, which would reduce the risk of a malicious attack. Reviewing and making sure the security policies are authentic ensures the safety of the organization data.

Audit:

1. Define the Scope of the Security Assessment

- Clearly define the scope of the security assessment for your Amazon DocumentDB cluster.
- Determine the objectives, areas of focus, and any specific compliance or security standards you must adhere to.

2. Review Security Documentation

- Familiarize yourself with the AWS security best practices and documentation related to Amazon DocumentDB.
- Review the AWS Shared Responsibility Model and understand the security controls provided by AWS.

3. Assess Network Security

- Review the network security configuration of your Amazon DocumentDB cluster.
- Ensure it is deployed within a secure Virtual Private Cloud (VPC) with appropriate security groups and network access control lists (ACLs).
- Validate that the network traffic to and from the cluster is appropriately restricted based on your security requirements.

4. Evaluate Encryption Configuration

- Assess the encryption settings for your Amazon DocumentDB cluster.
- Verify that encryption at rest is enabled and that the data stored in the cluster is encrypted.

Internal Only - General

- Validate that encryption in transit is enforced, ensuring that all client connections to the cluster are encrypted using SSL/TLS.

5. Review Access Control Mechanisms

- Evaluate the access control mechanisms implemented for your Amazon DocumentDB cluster.
- Ensure that appropriate Identity and Access Management (IAM) policies and roles are in place to control access to the cluster.
- Review user accounts and their privileges, and validate that multi-factor authentication (MFA) is enforced for administrative access.

6. Examine Audit Logging and Monitoring

- Review the audit logging and monitoring configuration for your Amazon DocumentDB cluster.
- Verify that audit logging is enabled, capturing relevant database activities and events.
- Evaluate the monitoring setup using Amazon CloudWatch or other tools to detect unusual or suspicious activities.

7. Assess Backup and Disaster Recovery

- Evaluate the backup and disaster recovery mechanisms in place for your Amazon DocumentDB cluster.
- Verify that automated backups are enabled and configured with an appropriate retention period.
- Validate that manual backups can be performed and restored successfully.

8. Perform Vulnerability Scanning and Penetration Testing (If Applicable)

- If allowed and within the terms of service, perform vulnerability scanning and penetration testing on your Amazon DocumentDB cluster.
- Conduct security assessments to identify any vulnerabilities or weaknesses that could be exploited.

9. Document Findings and Remediation Plan

- Document the findings of your security assessment, including any identified vulnerabilities or areas of improvement.
- Develop a remediation plan that addresses the identified issues and outlines the necessary actions to enhance the security posture of your DocumentDB cluster.

10. Implement Remediation Measures

- Implement the necessary remediation measures based on your remediation plan.

Internal Only - General

- Apply security patches, adjust configuration settings, and strengthen access controls as required.

11. Regularly Repeat the Security Assessment

- Conduct regular security assessments on your Amazon DocumentDB cluster to ensure ongoing compliance and identify new risks or vulnerabilities.
- Stay updated with security best practices and apply any relevant updates or patches to your cluster.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	18.1 Establish and Maintain a Penetration Testing Program Establish and maintain a penetration testing program appropriate to the size, complexity, and maturity of the enterprise. Penetration testing program characteristics include scope, such as network, web application, Application Programming Interface (API), hosted services, and physical premise controls; frequency; limitations, such as acceptable hours, and excluded attack types; point of contact information; remediation, such as how findings will be routed internally; and retrospective requirements.		●	●
v7	20.1 Establish a Penetration Testing Program Establish a program for penetration tests that includes a full scope of blended attacks, such as wireless, client-based, and web application attacks.		●	●

Internal Only - General

7.12 Ensure DocumentDB has delete protection enabled (Manual)

Profile Applicability:

- Level 1

Description:

Ensure that delete protection is enabled on database instances to prevent accidental or unauthorized deletion. This setting safeguards critical databases by requiring explicit disabling of delete protection before deletion, reducing the risk of data loss through human error or malicious activity.

Rationale:

Delete protection provides a safeguard against inadvertent or malicious deletion of critical databases. By requiring deliberate action to disable deletion protection, organizations mitigate risks associated with accidental data deletion and enhance the overall resilience of their data storage platform.

Impact:

Failure to enable delete protection increases the risk of irreversible data loss, potential service disruption, and operational downtime.

Audit:

List deletion protection status for all DocumentDB clusters:

```
aws docdb describe-db-clusters \
  --query
"DBClusters[*].{DBClusterIdentifier:DBClusterIdentifier,DeletionProtection:De
letionProtection}"
```

- DeletionProtection: true ⇒ Deletion protection is enabled.
- DeletionProtection: false ⇒ Deletion protection is not enabled (non-compliant).

Remediation:

Enable deletion protection on a specific cluster:

```
aws docdb modify-db-cluster \
  --db-cluster-identifier <cluster-identifier> \
  --deletion-protection \
  --apply-immediately
```

- Replace with your DocDB cluster ID.
- This change is applied immediately without downtime.
- Once enabled, the cluster cannot be deleted without first disabling deletion protection.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4 Secure Configuration of Enterprise Assets and Software Establish and maintain the secure configuration of enterprise assets (end-user devices, including portable and mobile; network devices; non-computing/IoT devices; and servers) and software (operating systems and applications).			

Internal Only - General

8 Amazon Keyspaces (formerly Amazon Managed Apache Cassandra Service)

Amazon Keyspaces, formerly known as Amazon Managed Apache Cassandra Service (MCS), is a fully managed, serverless, and scalable database service offered by Amazon Web Services (AWS). It is designed to provide developers with a highly available, globally distributed, and fully managed Apache Cassandra database compatible service. Cassandra is a popular NoSQL database known for its ability to handle large volumes of data across multiple regions and provide high availability and fault tolerance.

Internal Only - General

8.1 Ensure Keyspace Security is Configured (Manual)

Profile Applicability:

- Level 1

Description:

To access Amazon Keyspaces, the user would be required to log in with their AWS credentials. Once logged in the user can access the AWS resources and can explore the resources that Amazon Keyspaces offers. Amazon Keyspaces offers a lot of security that can mitigate a potential attack.

Rationale:

Audit:

1. Sign in to the AWS Management Console

- Sign in to the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon Keyspaces Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/keyspaces/.

3. Explore Amazon Keyspaces Security Features

- In the Amazon Keyspaces console, navigate to the `Features` or `Security` section to explore the available security features.
- Take note of the following critical security features:
  - Encryption at Rest: Understand how Amazon Keyspaces provides encryption at rest for your data. It uses server-side encryption by default, ensuring that data stored in Keyspaces is encrypted.
  - Encryption in Transit: Learn how to configure encryption in transit for data transmitted between your client applications and Amazon Keyspaces. Amazon Keyspaces supports Transport Layer Security (TLS) encryption to secure the communication channel.
  - Virtual Private Cloud (VPC) Support: Explore the VPC support options Amazon Keyspaces provides. It allows you to deploy your Keyspaces resources within your VPC for enhanced network isolation and control.
  - Authentication Options: Understand the authentication mechanisms available in Amazon Keyspaces. IAM for Cassandra allows you to use AWS Identity and Access Management (IAM) to authenticate and authorize client connections to Keyspaces.

    - Access Control: Learn about access control options in Amazon Keyspaces. It supports fine-grained access control using Access Control Lists (ACLs) at the table and row level to manage access permissions for different users or roles.
    - Audit Logging: Explore how to enable audit logging for Amazon Keyspaces. Amazon CloudWatch Logs can capture and store logs from your Keyspaces resources, providing visibility into activities for security and compliance purposes.

4. Documentation and Resources

- Access the official Amazon Keyspaces documentation by navigating to the `Documentation` or `Learn` section in the Amazon Keyspaces console.
- Review the comprehensive documentation to gain in-depth knowledge about each security feature, including best practices, configuration options, and implementation details.
- Utilize other AWS resources such as whitepapers, blogs, and security-related documentation further to enhance your understanding of Amazon Keyspaces security features.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

8.2 Ensure Network Security is Enabled (Manual)

Profile Applicability:

- Level 1

Description:

In order to access Amazon Keyspaces the user is required to set specific networking parameters and security measurements without these extra steps they will not be able to access it. Users are required to create or select a virtual private cloud (VPC) and define their inbound and outbound rules accordingly.

Rationale:

Impact:

Only authorized users have access to the database which limits and controls any risk of an attack. This ensures better performance of the system to a private network and better security.

Audit:

1. Create or Select a Virtual Private Cloud (VPC)

- Sign in to the AWS Management Console and open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
- Create a new VPC or select an existing VPC where you want to deploy your Amazon Keyspaces resources.

2. Configure Subnets

- In the VPC console, navigate to `Subnets` in the left-side menu.
- Create or select the subnets within your VPC where you want to deploy your Amazon Keyspaces resources.
- Ensure you have private subnets to isolate your Keyspaces resources from the public internet.

3. Define Security Groups

- In the VPC console, navigate to `Security Groups` in the left-side menu.
- Create a new security group or select an existing one for your Amazon Keyspaces resources.
- Configure inbound and outbound rules in the security group to control traffic access.
  - Allow inbound access only from trusted sources, such as specific IP ranges or security groups, on the necessary ports used by Amazon Keyspaces.

  - Define outbound rules based on your requirements, allowing outbound traffic to necessary destinations or ports.
- Associate the security group with your Amazon Keyspaces resources.

4. Configure Network Access Control Lists (ACLs)

- In the VPC console, navigate to `Network ACLs` in the left-side menu.
- Create or select the network ACLs associated with the subnets used by your Amazon Keyspaces resources.
- Configure inbound and outbound rules in the network ACLs to control traffic access.
  - Define rules based on your security requirements, allowing only necessary protocols, ports, and IP ranges.
  - list text hereDeny unnecessary or unwanted traffic.
- Associate the network ACLs with the subnets used by your Amazon Keyspaces resources.

5. Configure VPC Endpoints

- In the VPC console, navigate to `Endpoints` in the left-side menu.
- Create or select the VPC endpoints required for Amazon Keyspaces.
- If you need to access Keyspaces from within your VPC, create a VPC endpoint for Amazon Keyspaces to connect your applications securely.
- If you need to access Keyspaces from another VPC or on-premises network, set up VPC peering or a transit gateway to establish a secure connection.

6. Verify Connectivity and Test

- Launch an Amazon EC2 instance within the same VPC and subnet as your Amazon Keyspaces resources or use an existing one.
- Connect to the EC2 instance using SSH or other remote access methods.
- Test the connectivity to your Amazon Keyspaces resources by trying to connect to them using the appropriate client or utility.
- Verify that the network security settings allow the necessary traffic and deny unauthorized access.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	12.2 Establish and Maintain a Secure Network Architecture Establish and maintain a secure network architecture. A secure network architecture must address segmentation, least privilege, and availability, at a minimum.		●	●
v7	11.7 Manage Network Infrastructure Through a Dedicated Network Manage the network infrastructure across network connections that are separated from the business use of that network, relying on separate VLANs or, preferably, on entirely different physical connectivity for management sessions for network devices.		●	●

8.3 Ensure Data at Rest and in Transit is Encrypted (Manual)

Profile Applicability:

- Level 1

Description:

Once a user is logged in to their AWS account and has access to their Amazon Keyspaces they are encouraged to choose from the following two options to encrypt their data. Depending on which key they select for encryption at rest would store the data according to their preference. For encryption in transit the user is also encouraged to choose from two options depending on if the data needs to be encrypted during transit.

Rationale:

Impact:

Prevents any unauthorized user from accessing the database and provides security when transferring the data from one location to another.

Audit:

1. Sign in to the AWS Management Console

- Sign in to the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon Keyspaces Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/keyspaces/.

3. Select the Keyspace

- Choose the Keyspace (database) for which you want to enable encryption at rest and in transit.
- Click on the Keyspace name to access its details page.

4. Enable Encryption at Rest

- In the Keyspace details page, click on the `Configuration` tab.
- Under the `Encryption` section, locate the "Encryption at Rest" option.
- Click on `Edit`.
- Select the desired encryption setting:

Internal Only - General

    - Default Encryption: Choose this option to use the default AWS-managed key for encryption at rest. Amazon Keyspaces automatically encrypts your data using this default key.
    - Customer Managed Key (CMK): Choose this option if you want to use your own AWS Key Management Service (KMS) customer-managed key for encryption. Select the appropriate CMK from the dropdown menu.
- Click "Save" to enable encryption at rest for the Keyspace.

5. Enable Encryption in Transit

- In the Keyspace details page, click on the `Configuration` tab.
- Under the `Encryption` section, locate the "Encryption in Transit" option.
- Click on `Edit`.
- Select the desired encryption setting:
    - Encryption in Transit Enabled: Choose this option to enable encryption in transit for data transmitted between your client applications and Amazon Keyspaces. Keyspaces support Transport Layer Security (TLS) encryption for secure communication.
    - Encryption in Transit Disabled: Choose this option if you do not require encryption in transit.
- Click "Save" to enable encryption in transit for the Keyspace.

6. Verify Encryption Status

- Wait a few minutes for the changes to propagate and the encryption settings to take effect.
- Refresh the Keyspace details page to see the updated encryption status.
- Verify that encryption at rest and in transit are enabled for the Keyspace.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.10 Encrypt Sensitive Data in Transit Encrypt sensitive data in transit. Example implementations can include: Transport Layer Security (TLS) and Open Secure Shell (OpenSSH).		●	●

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.11 Encrypt Sensitive Data at Rest Encrypt sensitive data at rest on servers, applications, and databases containing sensitive data. Storage-layer encryption, also known as server-side encryption, meets the minimum requirement of this Safeguard. Additional encryption methods may include application-layer encryption, also known as client-side encryption, where access to the data storage device(s) does not permit access to the plain-text data.		●	●
v7	14.4 Encrypt All Sensitive Information in Transit Encrypt all sensitive information in transit.		●	●
v7	14.8 Encrypt Sensitive Information at Rest Encrypt all sensitive information at rest using a tool that requires a secondary authentication mechanism not integrated into the operating system, in order to access the information.			●

8.4 Ensure Amazon Keyspaces tables have Point-in-Time Recovery (PITR) enabled (Manual)

Profile Applicability:

- Level 1

Description:

Ensure that Amazon Keyspaces tables have Point-in-Time Recovery (PITR) enabled. When PITR is enabled, Amazon Keyspaces automatically creates continuous backups of table data, allowing tables to be restored to any point in time within the last 35 days. This protection is applied at the table level and provides defense against accidental writes, deletions, and other data loss scenarios.

Rationale:

Enabling PITR on Amazon Keyspaces tables provides continuous, automatic backup protection without requiring manual snapshot management or impacting table performance or availability. In the event of accidental data corruption, malicious writes, or system failures, PITR allows rapid recovery to any second within the last 35 days, significantly reducing data loss exposure and recovery time.

Impact:

Enabling PITR for Amazon Keyspaces ensures that table data is continuously protected and recoverable to any point within 35 days, providing strong defense against accidental loss and corruption while maintaining full table performance and availability.

Audit:

List PITR status for all tables in a keyspace:

```
aws keyspaces get-table \
  --keyspace-name <keyspace-name> \
  --table-name <table-name>
```

- If the value of pointInTimeRecovery = DISABLED, this means PITR is turned off

Remediation:

Enable PITR on a specific table:

```
aws keyspaces update-table \
  --keyspace-name <keyspace-name> \
  --table-name <table-name> \
  --point-in-time-recovery status=ENABLED
```

- This command enables PITR on the specified table in the keyspace.
- The change takes effect immediately with no performance impact.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	11 Data Recovery Establish and maintain data recovery practices sufficient to restore in-scope enterprise assets to a pre-incident and trusted state.			

9 Amazon Neptune

Amazon Neptune is a fully managed graph database service provided by Amazon Web Services (AWS). It is designed to store, query, and analyze highly connected data with complex relationships, making it particularly well-suited for applications that require deep and rich data modeling, such as social networking, recommendation engines, fraud detection, and knowledge graphs.

Internal Only - General

9.1 Ensure Network Security is Enabled (Manual)

Profile Applicability:

- Level 1

Description:

This helps ensure that all the necessary security measurements are taken to prevent a cyber-attack. Such as utilizing VPC, creating certain inbound and outbound rules, and ACLs.

Rationale:

Impact:

Provides privacy and lets the user customize their security preferences. Prevents private network from interfering with public networks.

Audit:

1. Sign in to the AWS Management Console

- Sign in to the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon Neptune Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/neptune/.

3. Select the Neptune Cluster

- Choose the Amazon Neptune cluster for which you want to configure network security.
- Click on the cluster name to access its details page.

4. Configure Security Groups

- In the cluster details page, navigate to the `Connectivity & Security` or `Network & Security` section.
- Under `Security Groups`, click on `Manage security groups`.
- Click on `Create new security group` or select an existing security group associated with your Neptune cluster.
- Configure inbound and outbound rules within the security group to control network traffic.
    - For inbound rules, specify the allowed source IP addresses or security groups and the necessary ports for accessing the Neptune cluster.

Internal Only - General

    - For outbound rules, define the allowed destination IP addresses or security groups and the required ports for outbound connections from the Neptune cluster.
- Save the security group settings.

5. Configure Network Access Control Lists (ACLs)

- In the cluster details page, navigate to the `Connectivity & Security` or `Network & Security` section.
- Under `Network Access Control Lists (ACLs)`, click on `Manage network ACLs`.
- Create a new network ACL or select an existing one associated with your Amazon Neptune cluster.
- Configure inbound and outbound rules within the network ACL to control network traffic at the subnet level.
- Define rules based on IP address ranges, protocols, and ports to allow or deny specific traffic.
- Consider security best practices and compliance requirements when configuring the network ACL rules.
- Save the network ACL settings.

6. Verify Network Security Configuration

- Review the security group and network ACL settings to ensure they align with your security requirements.
- Confirm that the inbound and outbound rules only allow necessary traffic and deny unauthorized access.
- Verify that your Neptune cluster's security groups and network ACLs are correctly configured.

7. Test Network Connectivity

- Launch an Amazon EC2 instance within the same VPC and subnet as your Neptune cluster, or use an existing one.
- Connect to the EC2 instance using SSH or other remote access methods.
- Test the network connectivity to your Neptune cluster by attempting to connect to it using the appropriate client or utility.
- Ensure that the network security settings allow the necessary traffic and deny unauthorized access.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	12.2 Establish and Maintain a Secure Network Architecture Establish and maintain a secure network architecture. A secure network architecture must address segmentation, least privilege, and availability, at a minimum.		●	●
v7	11.7 Manage Network Infrastructure Through a Dedicated Network Manage the network infrastructure across network connections that are separated from the business use of that network, relying on separate VLANs or, preferably, on entirely different physical connectivity for management sessions for network devices.		●	●

Internal Only - General

9.2 Ensure Data at Rest is Encrypted (Manual)

Profile Applicability:

- Level 1

Description:

This helps ensure that the data is kept secure and protected when at rest. The user must choose from two key options which then determine when the data is encrypted at rest.

Rationale:

Impact:

If an unauthorized user steals the data, it would be unreadable for them because a key would be required to decrypt the message into plaintext.

Audit:

1. Sign into the AWS Management Console

- Sign into the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon Neptune Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/neptune/.

3. Select the Neptune Cluster

- Choose the Amazon Neptune cluster for which you want to enable encryption at rest.
- Click on the cluster name to access its details page.

4. Enable Encryption at Rest

- In the cluster details page, navigate to the `Configuration` or `Encryption at Rest` section.
- Under `Encryption at Rest`, click on `Modify`.
- In the `Encryption at Rest` dialog box, select the encryption option you prefer:
  - AWS managed key (default): Choose this option to use the default AWS managed key for encryption.
  - Customer-managed key (CMK): Choose this option if you want to use your own AWS Key Management Service (KMS) customer-managed key for encryption. Select the appropriate CMK from the dropdown menu.

Internal Only - General

- Click `Apply Changes` to enable encryption at rest for the Neptune cluster.

5. Verify Encryption Status

- Wait a few minutes for the changes and configuration to take effect.
- Refresh the cluster details page to see the updated encryption status.
- Verify that encryption at rest is enabled for the Neptune cluster.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.11 Encrypt Sensitive Data at Rest Encrypt sensitive data at rest on servers, applications, and databases containing sensitive data. Storage-layer encryption, also known as server-side encryption, meets the minimum requirement of this Safeguard. Additional encryption methods may include application-layer encryption, also known as client-side encryption, where access to the data storage device(s) does not permit access to the plain-text data.		●	●
v7	14.8 Encrypt Sensitive Information at Rest Encrypt all sensitive information at rest using a tool that requires a secondary authentication mechanism not integrated into the operating system, in order to access the information.			●

9.3 Ensure Data in Transit is Encrypted (Manual)

Profile Applicability:

- Level 1

Description:

Enabling encryption in transit helps that the data is protected when it is moving from one location to another.

Rationale:

Impact:

If an unauthorized user steals the data, it would be unreadable for them because a key would be required to decrypt the message into plaintext.

Audit:

1. Sign into the AWS Management Console

- Sign into the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon Neptune Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/neptune/.

3. Select the Neptune Cluster

- Choose the Amazon Neptune cluster for which you want to implement encryption in transit.
- Click on the cluster name to access its details page.

4. Enable SSL/TLS Encryption

- In the cluster details page, navigate to the `Configuration` or `Encryption in Transit` section.
- Under `Encryption in Transit`, ensure that the `Enable` option is selected.
- Optionally, you can also select the `Enforce` option to require SSL/TLS encryption for all client connections to the Neptune cluster.
- Click `Apply Changes` to enable SSL/TLS encryption for the Neptune cluster.

5. Update Client Applications

Internal Only - General

- When connecting to the Neptune cluster, update your client applications to establish an SSL/TLS-encrypted connection.
- Consult your client drivers or libraries documentation or configuration settings to enable SSL/TLS encryption.
- Configure the necessary SSL/TLS settings, such as specifying the SSL/TLS certificate to use.

6. Verify Encryption in Transit

- Test the connection to the Neptune cluster from your client application.
- Ensure that the connection is established using SSL/TLS encryption.
- Verify that all data transmitted between your client applications and the Neptune cluster is encrypted in transit.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.10 Encrypt Sensitive Data in Transit Encrypt sensitive data in transit. Example implementations can include: Transport Layer Security (TLS) and Open Secure Shell (OpenSSH).		●	●
v7	14.4 Encrypt All Sensitive Information in Transit Encrypt all sensitive information in transit.		●	●

9.4 Ensure Authentication and Access Control is Enabled (Manual)

Profile Applicability:

- Level 1

Description:

This helps ensure that there are specific IAM roles and policies that are given the necessary information within a Neptune DB cluster to operate as needed.

Rationale:

Impact:

Allowing authentication verifies the identity of the person and who has appropriate access to a company's data.

Audit:

1. Sign into the AWS Management Console

- Sign into the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon Neptune Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/neptune/.

3. Select the Neptune Cluster

- Choose the Amazon Neptune cluster on which you want to implement authentication and access control.
- Click on the cluster name to access its details page.

4. Enable IAM Database Authentication

- In the cluster details page, navigate to the `Configuration` or `Database Authentication` section.
- Under `Database Authentication`, select the option to enable IAM database authentication.
- Click `Apply Changes` to enable IAM database authentication for the Neptune cluster.

5. Configure IAM Roles and Policies

- Open the AWS Identity and Access Management (IAM) console by navigating to `IAM` in the AWS Management Console.
- Create IAM roles and policies that define the desired access control for your Neptune resources.
- Assign the necessary permissions to the IAM roles to allow specific actions on the Neptune cluster, such as read, write, or manage operations.
- Associate the IAM roles with the appropriate users, groups, or AWS services that need access to the Neptune cluster.

6. Test IAM Database Authentication

- Update your client applications or tools to use IAM database authentication when connecting to the Neptune cluster.
- Configure your applications to assume the necessary IAM roles before establishing a connection to Neptune.
- Test the connection from your client application to the Neptune cluster to verify that IAM database authentication is working as expected.
- Ensure that users or services are authenticated and authorized based on the IAM roles and policies defined.

7. Regularly Review and Update IAM Roles and Policies

- Periodically review your IAM roles and policies to ensure they align with your security requirements and access control needs.
- Make necessary updates to IAM roles and policies to adapt to changes in user access requirements or organizational security policies.
- Follow the principle of least privilege and ensure that users or services have only the necessary permissions to perform their required actions on the Neptune cluster.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user’s need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●

Controls Version	Control	IG 1	IG 2	IG 3
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

Internal Only - General

9.5 Ensure Audit Logging is Enabled (Manual)

Profile Applicability:

- Level 1

Description:

This control is important because it helps ensure activity within the cluster and identifies who has last modified the document and who has access to it, in case of breaches. It also ensures compliance with regulation requirements.

Rationale:

Impact:

Reduces risks of any fraud since worker activity is being monitored and tracked.

Audit:

1. Sign into the AWS Management Console

- Sign into the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon Neptune Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/neptune/.

3. Select the Neptune Cluster

- Choose the Amazon Neptune cluster on which you want to enable audit logging. Click on the cluster name to access its details page.

4. Enable Amazon CloudWatch Logs

- In the cluster details page, navigate to the `Monitoring` or `Logging` section.
- Under `CloudWatch Logs`, click `Enable` to enable logging for the Neptune cluster.
- Select an existing CloudWatch Logs group or create a new one to store the logs.
- Choose the appropriate retention period for the logs, considering your compliance and retention requirements.
- Click `Save` or `Apply Changes` to enable CloudWatch Logs for the Neptune cluster.

5. Configure Log Levels (Optional)

- In the cluster details page, navigate to the `Configuration` or `Logging` section.

Internal Only - General

- Under `Logging`, you may have the option to configure log levels for different components of Neptune, such as query logs or error logs.
- Adjust the log levels according to your logging and troubleshooting needs.
- Click `Apply Changes` to save the log level configuration.

6. Review and Analyze Logs

- Access the Amazon CloudWatch console by navigating to `CloudWatch` in the AWS Management Console.
- Go to the CloudWatch Logs section and locate the log group associated with your Neptune cluster.
- Select the log group and review the logs generated by Neptune.
- Analyze the logs for troubleshooting, performance monitoring, or auditing purposes.

7. Set Up Log Metric Filters and Alarms (Optional)

- In the CloudWatch console, navigate to `Metric Filters` in the left-side menu.
- Click on `Create metric filter` to set up filters to extract specific information from the Neptune logs.
- Define the filter patterns to capture the desired log events and extract the required metrics.
- Configure alarms based on the extracted metrics to trigger notifications or automated actions when specific conditions are met.

8. Regularly Monitor Logs

- Continuously monitor the logs generated by Neptune using the CloudWatch Logs console or programmatically using the CloudWatch APIs.
- Review the logs regularly to identify any abnormal or suspicious activities.
- Set up appropriate notifications or alerts to proactively respond to critical log events.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	8.1 Establish and Maintain an Audit Log Management Process Establish and maintain an audit log management process that defines the enterprise’s logging requirements. At a minimum, address the collection, review, and retention of audit logs for enterprise assets. Review and update documentation annually, or when significant enterprise changes occur that could impact this Safeguard.	●	●	●
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●

Internal Only - General

9.6 Ensure Security Configurations are Reviewed Regularly (Manual)

Profile Applicability:

- Level 1

Description:

This helps by removing or updating any IAM roles, security networks, encryption settings, audit logging, and authentication. By updating or removing a few things from these lists it helps tighten security and ensures that the users do not have excessive permissions.

Rationale:

Impact:

By updating and revising the control within our Amazon Neptune cluster it would keep the system as secure as possible.

Audit:

1. Establish a Security Review Schedule

- Determine a regular schedule for reviewing and updating the security configuration of your Amazon Neptune environment.
- Consider factors such as the frequency of changes, compliance requirements, and industry best practices to determine the appropriate review interval.

2. Monitor AWS Security Bulletins

- Stay informed about AWS security updates and announcements related to Amazon Neptune.
- Regularly review AWS security bulletins and notifications to identify any security patches, updates, or new features relevant to your Neptune environment.
- Take note of any security recommendations or best practices provided by AWS.

3. Review IAM Roles and Policies

- Access the AWS Identity and Access Management (IAM) console by navigating to `IAM` in the AWS Management Console.
- Review the IAM roles and policies associated with your Neptune resources.
- Ensure that the assigned permissions align with the principle of least privilege and reflect the current access requirements.
- Update the IAM roles and policies as needed to adapt to changes in user access or security requirements.

Internal Only - General

4. Review Security Groups and Network ACLs

- Access the Amazon Neptune console by navigating to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/neptune/.
- In the Neptune console, navigate to the `Connectivity & Security` or `Network & Security` section.
- Review the security groups and network ACLs associated with your Neptune clusters.
- Ensure that the inbound and outbound rules are up to date and aligned with your security requirements.
- Remove any unnecessary or outdated rules and add new rules if required.

5. Review Encryption Settings

- Navigate to the `Configuration` section or relevant encryption settings in the Neptune console.
- Review the encryption settings for both encryption at rest and encryption in transit.
- Ensure that the appropriate encryption options and key management strategies are in place.
- Consider rotating encryption keys periodically, following best practices and compliance requirements.

6. Review VPC Configuration

- Access the Amazon VPC console by navigating to `VPC` in the AWS Management Console.
- Review the VPC configuration associated with your Neptune clusters.
- Ensure the subnets, routing tables, and VPC peering settings are configured correctly.
- Verify that the network architecture provides your Neptune resources' desired isolation and connectivity.

7. Conduct Security Assessments

- Periodically conduct security assessments and penetration testing on your Neptune environment.
- Engage security experts or use appropriate security tools to identify vulnerabilities, weaknesses, or misconfigurations.
- Analyze the assessment results and take necessary actions to remediate any security issues or risks.

8. Stay Up to Date with Best Practices

- Continuously educate yourself and your team on the latest security best practices for Amazon Neptune.
- Stay informed about emerging security threats and vulnerabilities.

  -Regularly review AWS documentation, security blogs, and other relevant resources to enhance your understanding and implementation of security practices.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	5 Account Management Use processes and tools to assign and manage authorization to credentials for user accounts, including administrator accounts, as well as service accounts, to enterprise assets and software.			
v7	5 Secure Configuration for Hardware and Software on Mobile Devices, Laptops, Workstations and Servers Secure Configuration for Hardware and Software on Mobile Devices, Laptops, Workstations and Servers			

Internal Only - General

9.7 Ensure Monitoring and Alerting is Enabled (Manual)

Profile Applicability:

- Level 1

Description:

Rationale:

Audit:

1. Sign in to the AWS Management Console

- Sign in to the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.

2. Open the Amazon Neptune Console

- Navigate to the service using the `Find Services` search bar or by directly accessing the console at https://console.aws.amazon.com/neptune/.

3. Select the Neptune Cluster

- Choose the Amazon Neptune cluster on which you want to implement monitoring and alerting.
- Click on the cluster name to access its details page.

4. Set Up Amazon CloudWatch Metrics

- In the cluster details page, navigate to the `Monitoring` or `Metrics` section.
- Enable CloudWatch metrics for the Neptune cluster by clicking `Enable` or `Configure`.
- Select the desired metrics to monitor, such as CPU utilization, storage usage, or network throughput.
- Choose the appropriate granularity and sampling intervals for the metrics.
- Click `Save` or `Apply Changes` to enable CloudWatch metrics for the Neptune cluster.

5. Configure CloudWatch Alarms

- In the CloudWatch console, navigate to `Alarms` in the left-side menu.
- Click `Create alarm` to configure alarms based on specific metric thresholds.
- Select the desired metric to monitor and set the threshold values for triggering an alarm.
- Define the actions to be taken when the alarm state changes, such as sending notifications or triggering automated actions.

- Configure the alarm settings, including alarm name, description, and notification recipients.
- Click `Create alarm` to save the alarm configuration.

6. Set Up Amazon EventBridge Rules

- In the Amazon EventBridge console, navigate to `Rules` in the left-side menu.
- Click on `Create rule` to set up rules for specific events or log entries related to Neptune.
- Define the event pattern or log filter to match the desired events.
- Configure the target actions to be taken when the rule matches an event, such as sending notifications or invoking AWS Lambda functions.
- Specify the rule settings, including rule name, description, and event source.
- Click `Create` to save the rule configuration.

7. Review and Customize Metrics and Alarms

- Periodically review the metrics and alarms configured for your Neptune cluster.
- Adjust the metric thresholds and alarm settings based on your performance and alerting requirements.
- Consider adding more metrics or alarms as needed to monitor additional aspects of your Neptune environment.

8. Regularly Monitor and Respond to Alerts

- Continuously monitor the CloudWatch metrics and alarm states for your Neptune cluster.
- Respond promptly to any alarms triggered by critical or abnormal conditions.
- Investigate the root causes of the alerts and take appropriate actions to mitigate issues.

9. Utilize Additional Monitoring Tools

- Explore and leverage additional monitoring and observability tools available in the AWS ecosystem, such as Amazon CloudWatch Logs Insights, AWS X-Ray, or third-party monitoring solutions.
- Configure these tools to gather insights and detect any performance or security issues in your Neptune environment.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	8.1 Establish and Maintain an Audit Log Management Process Establish and maintain an audit log management process that defines the enterprise’s logging requirements. At a minimum, address the collection, review, and retention of audit logs for enterprise assets. Review and update documentation annually, or when significant enterprise changes occur that could impact this Safeguard.	●	●	●
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●

9.8 Ensure Neptune Database is not Publicly accessible (Manual)

Profile Applicability:

- Level 1

Description:

Neptune databases must not be publicly accessible. This means the database's network configuration should prevent assignment of public IP addresses or exposure to the public internet, ensuring that connections are only permitted from trusted internal networks.

Rationale:

Restricting public access to databases greatly reduces the attack surface for malicious actors. Publicly accessible databases are highly vulnerable to unauthorized login attempts, exploitation of software vulnerabilities and data breaches. Enforcing private access restricts connectivity and enforces the principle of least privilege and network segmentation.

Impact:

If public access is not properly restricted on databases, data stored in the database is at risk of exposure to the internet, increasing the likelihood of data loss and service disruption.

Audit:

1. Sign in to the AWS Management Console where the Aurora database cluster you are auditing resides.
2. Navigate to the Neptune Dashboard.

- You can find this under the Database category.

3. Select the DB instance name you wish to audit.

- This opens the details page for your specific Neptune DB instance.

4. Under the Connectivity & Security tab, check the value of Publicly accessible:

- If Set to No, the instance is not publicly accessible; no further network verification is needed.
- If Set to Yes, continue with additional steps to fully assess exposure.

5. In the Networking section under Connectivity & security, locate the Subnets for the database:

Internal Only - General

- Right-click on the subnet link and open it in a new tab for further inspection.

6. With the subnet selected, review the attached Route Table:

- Check for routes with Destination: 0.0.0.0/0 and Target: an Internet Gateway (ID starts with igw-).
- If such a route exists, it enables access to the database from the public internet.

If the database is marked as "Publicly accessible: Yes" and the subnets contain a route to 0.0.0.0/0 via an Internet Gateway, the instance is exposed to the public internet.

Remediation:

1. Disable public accessibility on a specific Neptune instance:

```
aws neptune modify-db-instance \
  --db-instance-identifier <instance-identifier> \
  --no-publicly-accessible \
  --apply-immediately
```

- --no-publicly-accessible disables public accessibility for the instance.
- --apply-immediately applies the change without waiting for the next maintenance window.

2. Verify that public accessibility has been disabled

```
aws neptune describe-db-instances \
  --db-instance-identifier <instance-identifier> \
  --query
"DBInstances[0].{DBInstanceIdentifier:DBInstanceIdentifier,PubliclyAccessible
:PubliclyAccessible}"
```

- Confirm PubliclyAccessible is now false.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.12 Segment Data Processing and Storage Based on Sensitivity Segment data processing and storage based on the sensitivity of the data. Do not process sensitive data on enterprise assets intended for lower sensitivity data.		●	●
v8	4 Secure Configuration of Enterprise Assets and Software Establish and maintain the secure configuration of enterprise assets (end-user devices, including portable and mobile; network devices; non-computing/IoT devices; and servers) and software (operating systems and applications).			

Controls Version	Control	IG 1	IG 2	IG 3
v8	12.2 Establish and Maintain a Secure Network Architecture Establish and maintain a secure network architecture. A secure network architecture must address segmentation, least privilege, and availability, at a minimum.		●	●

9.9 Ensure Neptune Database has automatic backups enabled (Manual)

Profile Applicability:

- Level 1

Description:

Ensure that Amazon Neptune DB clusters have automated backups enabled with a non-zero backup retention period (for example, 7 to 35 days). Neptune automatically creates continuous, incremental backups of cluster data and retains them for the configured retention period, allowing point-in-time recovery to any second within the backup window.

Rationale:

Enabling automated backups with a sufficient retention period ensures that Neptune cluster data can be quickly recovered to any point within the retention window, protecting against accidental deletions, data corruption, application errors, and infrastructure failures. Neptune backups are continuous and incremental, providing robust disaster recovery and business continuity capabilities with minimal storage overhead.

Impact:

Enabling automated backups for Neptune ensures that cluster data is continuously protected and recoverable to any point within the configured retention period (1–35 days), significantly reducing exposure to data loss and enabling rapid recovery from operational incidents.

Audit:

List backup retention status for all Neptune DB clusters:

```
aws neptune describe-db-clusters \
  --query
"DBClusters[*].{DBClusterIdentifier:DBClusterIdentifier,Engine:Engine,BackupR
etentionPeriod:BackupRetentionPeriod}"
```

- BackupRetentionPeriod > 0 ⇒ Automated backups are enabled (compliant).
- BackupRetentionPeriod = 0 or missing/null ⇒ Automated backups are not enabled (non-compliant).

Remediation:

Enable automated backups on a specific Neptune cluster:

```
aws neptune modify-db-cluster \
  --db-cluster-identifier <cluster-identifier> \
  --backup-retention-period 7 \
  --preferred-backup-window "03:00-04:00" \
  --apply-immediately
```

- --backup-retention-period 7 sets a 7-day retention; use a value between 1 and 35 days aligned with your policy.
- --preferred-backup-window "03:00-04:00" (optional) sets the daily UTC backup window during off-peak hours to minimize performance impact.
- --apply-immediately applies the change without waiting for the next maintenance window.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	11 Data Recovery Establish and maintain data recovery practices sufficient to restore in-scope enterprise assets to a pre-incident and trusted state.			

9.10 Ensure Database has delete protection enabled (Manual)

Profile Applicability:

- Level 1

Description:

Ensure that delete protection is enabled on database instances to prevent accidental or unauthorized deletion. This setting safeguards critical databases by requiring explicit disabling of delete protection before deletion, reducing the risk of data loss through human error or malicious activity.

Rationale:

Delete protection provides a safeguard against inadvertent or malicious deletion of critical databases. By requiring deliberate action to disable deletion protection, organizations mitigate risks associated with accidental data deletion and enhance the overall resilience of their data storage platform.

Impact:

Failure to enable delete protection increases the risk of irreversible data loss, potential service disruption, and operational downtime.

Audit:

List deletion protection status for all Neptune DB clusters

```
aws neptune describe-db-clusters \
  --query
"DBClusters[*].{DBClusterIdentifier:DBClusterIdentifier,Engine:Engine,Deletio
nProtection:DeletionProtection}"
```

- DeletionProtection: true ⇒ Deletion protection is enabled (compliant).
- DeletionProtection: false ⇒ Deletion protection is not enabled (non-compliant).

Remediation:

Enable deletion protection on a specific Neptune cluster:

```
aws neptune modify-db-cluster \
  --db-cluster-identifier <cluster-identifier> \
  --deletion-protection \
  --apply-immediately
```

- --deletion-protection enables the deletion protection feature.
- --apply-immediately applies the change without waiting for the next maintenance window.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4 Secure Configuration of Enterprise Assets and Software Establish and maintain the secure configuration of enterprise assets (end-user devices, including portable and mobile; network devices; non-computing/IoT devices; and servers) and software (operating systems and applications).			

Internal Only - General

9.11 Ensure Neptune DB instances are deployed across multiple Availability Zones (AZs) (Manual)

Profile Applicability:

- Level 2

Description:

Deploying Amazon Neptune across multiple Availability Zones means configuring the database cluster nodes (primary and replicas) to be distributed in different AZs within the same AWS region. This multi-AZ deployment improves fault tolerance and availability by mitigating risks associated with failure or degradation in a single Availability Zone. If the primary node or an AZ becomes unavailable, Neptune can automatically fail over to a replica in a different AZ, minimizing downtime and data unavailability.

Rationale:

Distributing Neptune DBs across multiple AZs protects the database from localized infrastructure failures, such as power outages, networking disruptions, or hardware faults in a single AZ.

Impact:

Enabling multi-AZ deployment for Neptune DBs enhances system availability and resiliency, significantly reducing the risk of cache service interruptions due to an AZ failure or node disruption.

Audit:

List Multi-AZ status for all Neptune DB clusters

```
aws neptune describe-db-clusters \
  --query
"DBClusters[*].{DBClusterIdentifier:DBClusterIdentifier,Engine:Engine,MultiAZ
:MultiAZ}"
```

- MultiAZ: true ⇒ Multi-AZ is enabled (compliant).
- MultiAZ: false ⇒ Multi-AZ is not enabled (non-compliant).

Remediation:

Enable Multi-AZ on Neptune DB clusters by adding a reader replica in a different Availability Zone.

1. Sign in to the AWS Management Console where the Aurora database cluster you are auditing resides.
2. Navigate to the Neptune Dashboard.

- You can find this under the Database category.

3. Select the DB cluster where you want to create the reader instance.
4. Choose Actions, and then choose Add reader.
5. Configure the replica DB instance

On the Create replica DB instance page, specify the following options:

- DB instance class: Choose a DB instance class that matches or aligns with your primary instance (e.g., db.r5.large). This defines the processing and memory requirements for the Neptune replica.
- Availability zone: Specify a different Availability Zone than the primary DB instance. This is critical for Multi-AZ deployment. The list shows only AZs that are mapped by the DB subnet group for the cluster.
- Encryption: Enable or disable encryption (recommended: enable if primary has encryption enabled).
- Read replica source: Choose the identifier of the primary instance to create the Neptune replica for.
- DB instance identifier: Enter a unique name for the instance in your region. Consider including the AZ in the name (e.g., neptune-replica-us-east-1b).
- Database port: Specify the port number on which the database accepts connections (default: 8182 for Neptune).
- DB parameter group: Select the parameter group for this instance (typically the same as the primary).
- Log exports: Choose any logs you want to publish (audit, slowquery, etc.).
- Auto Minor Version Upgrade: Choose Yes to enable automatic minor version upgrades for the replica.

5. Choose Create read replica to create the Neptune replica instance.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4 Secure Configuration of Enterprise Assets and Software Establish and maintain the secure configuration of enterprise assets (end-user devices, including portable and mobile; network devices; non-computing/IoT devices; and servers) and software (operating systems and applications).			

10 Amazon Timestream

10.1 Ensure Data Ingestion is Secure (Manual)

Profile Applicability:

- Level 1

Description:

Rationale:

This helps ensure that the system is updated with any potential vulnerabilities that might pose a threat to the organization. Helps authenticate the sources that are coming to the database and ensures that only authorized users have the credential to access the data.

Audit:

1. Secure Data Sources

   Ensure that your data sources are protected with appropriate security measures. Implement secure network configurations, access controls, and authentication mechanisms for your data sources. Apply security patches and updates to your data source systems to prevent vulnerabilities.

2. Use HTTPS or AWS Direct Connect

   When ingesting data into Timestream, use secure communication protocols such as HTTPS. Encrypt data in transit to protect it from unauthorized interception. Consider using AWS Direct Connect for a dedicated private network connection to Timestream, ensuring data privacy.

3. Implement Client-Side Encryption

   Encrypt your data before sending it to Timestream using client-side encryption. Use industry-standard encryption algorithms and strong encryption keys to protect the confidentiality of your data. Store and manage the encryption keys securely using AWS Key Management Service (KMS).

4. Authenticate Data Sources

   Implement authentication mechanisms for your data sources to ensure only authorized sources can ingest data into Timestream. Use mechanisms such as API keys, access tokens, or client certificates to verify the authenticity of the data source. Consider integrating with AWS Identity and Access Management (IAM) for centralized authentication and access control.

5. Validate and Sanitize Data

Internal Only - General

Implement data validation and sanitization mechanisms to prevent injection attacks or malformed data from being ingested into Timestream. Use input validation techniques and enforce data format requirements to ensure the integrity of the ingested data. Implement data quality checks to identify and handle anomalies or outliers.

6. Monitor Data Ingestion

   Implement monitoring and logging for data ingestion processes. Regularly review logs and metrics related to data ingestion to detect anomalies or suspicious activities. Set up alarms and notifications for data ingestion failures or unexpected patterns.

7. Regularly Update Data Ingestion Components

   Keep your data ingestion components, such as APIs, scripts, or connectors, up to date with the latest security patches and updates. Follow safe coding practices and stay informed about security vulnerabilities and fixes specific to your data ingestion tools.

8. Implement Network Security Controls

   Use network security controls such as security groups, network ACLs, and VPC configurations to restrict access to your Timestream resources. Configure inbound and outbound traffic rules to allow only necessary network connections for data ingestion. Follow the principle of least privilege, granting access only to the required IPs or networks.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3 Data Protection Develop processes and technical controls to identify, classify, securely handle, retain, and dispose of data.			
v7	13 Data Protection Data Protection			

Internal Only - General

10.2 Ensure Data at Rest is Encrypted (Manual)

Profile Applicability:

- Level 1

Description:

Enable encryption at rest for Amazon Timestream to protect your data while it is stored. Utilize AWS Key Management Service (KMS) to manage and control the encryption keys used for data encryption. Configure Timestream to encrypt your data using AWS-managed keys or customer-managed keys.

Rationale:

This helps ensure that the data is kept secure and protected when at rest. The user must choose from two key options which then determine when the data is encrypted at rest.

Audit:

1. Understand Encryption at Rest in Timestream

   Familiarize yourself with the concept of encryption at rest and its importance in securing your data in Timestream. Understand that encryption at rest ensures that your data remains protected even if the underlying storage media is compromised.

2. Create an AWS Key Management Service (KMS) Key

   Open the AWS Management Console and navigate to the AWS Key Management Service (KMS) service. Create a new KMS customer master key (CMK) or use an existing one to manage the encryption keys for Timestream. Follow the AWS documentation and best practices for creating and managing KMS keys.

3. Enable Encryption at Rest in Timestream

   Open the Amazon Timestream console. Select the Timestream database or table you want to enable encryption at rest. Click on the "Encryption" tab or section. Choose the option to enable encryption at rest. Select the KMS key that you created earlier to be used for encryption.

4. Verify Encryption at Rest

   Confirm that encryption at rest is enabled for the selected Timestream database or table. Review the encryption settings in the Timestream console to ensure the correct KMS key is associated.

5. Monitor and Audit Encryption at Rest

   Regularly monitor the encryption at rest status in the Timestream console. Leverage AWS CloudTrail and AWS CloudWatch to monitor and track encryption-related activities or events. Set up appropriate alerts and notifications to detect any issues or unauthorized changes to the encryption settings.

6. Test Data Access and Decryption

   Access the Timestream data that is encrypted at rest. Verify that you can retrieve and decrypt the data using the appropriate access controls and KMS key permissions. Perform thorough testing to ensure data access and decryption functions as expected.

7. Review and Update Encryption Configuration

   Regularly review your encryption configuration and settings for Timestream. Ensure that the appropriate KMS key is still associated with the Timestream resources. Update the encryption settings if necessary, such as rotating encryption keys or modifying key policies.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.11 Encrypt Sensitive Data at Rest Encrypt sensitive data at rest on servers, applications, and databases containing sensitive data. Storage-layer encryption, also known as server-side encryption, meets the minimum requirement of this Safeguard. Additional encryption methods may include application-layer encryption, also known as client-side encryption, where access to the data storage device(s) does not permit access to the plain-text data.		●	●
v7	14.8 Encrypt Sensitive Information at Rest Encrypt all sensitive information at rest using a tool that requires a secondary authentication mechanism not integrated into the operating system, in order to access the information.			●

10.3 Ensure Encryption in Transit is Configured (Manual)

Profile Applicability:

- Level 1

Description:

Configure your applications or tools to use secure communication protocols when interacting with Amazon Timestream. Utilize endpoints to establish private and secure connections to Timestream.

Rationale:

The database uses HTTPS/TLS to encrypt data during transit. To secure your data in transit the individual should identify their client application and what is supported by HTTPS/TLS in order to configure it correctly. Also has an option for leverage, which creates a private connection between virtual private code (VPC) without interfering with public networks.

Impact:

If the client does not have the code configured correctly it would not be able to connect to the server.

Audit:

1. Understand Encryption in Transit in Timestream

   Familiarize yourself with the concept of encryption in transit and its importance in securing data communication. Understand that encryption in transit ensures that data transmitted between clients and Timestream remains confidential and protected from interception.

2. Use HTTPS for Communication

   Configure your client applications or tools to communicate with Amazon Timestream over HTTPS. Utilize the HTTPS protocol to establish secure encrypted connections between clients and the Timestream service. Ensure your client applications support the TLS (Transport Layer Security) protocol versions AWS recommends.

3. Leverage AWS PrivateLink (Optional)

   Consider using AWS PrivateLink to establish private and secure connections between your VPC and Timestream. Configure a VPC endpoint for Timestream to securely access the service without traversing the public internet.

Internal Only - General

4. Enable SSL/TLS Certificates

   Obtain and configure valid SSL/TLS certificates for your client applications or tools. Install the SSL/TLS certificates on your client systems or load balancers. Use the configured certificates to establish secure connections with Timestream.

5. Verify Encryption in Transit

   Validate that your client applications or tools are using secure communication channels. Verify that HTTPS is being utilized for communication with Timestream. Confirm that SSL/TLS certificates are properly configured and used in communication.

6. Monitor Encryption in Transit

   Utilize Amazon CloudWatch to monitor the metrics and logs related to your Timestream resources. Set up appropriate alarms and notifications to alert you of any potential security incidents or anomalies in the encryption in transit process. Regularly review the CloudWatch logs and metrics to ensure the integrity and security of the data in transit.

7. Regularly Update Encryption Configuration

   Stay informed about the latest encryption standards, protocols, and best practices. Regularly review and update your encryption configurations and settings to align with industry standards and security recommendations. Apply any necessary updates or patches to client applications or tools to maintain strong encryption in transit.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.10 Encrypt Sensitive Data in Transit Encrypt sensitive data in transit. Example implementations can include: Transport Layer Security (TLS) and Open Secure Shell (OpenSSH).		●	●
v7	14.4 Encrypt All Sensitive Information in Transit Encrypt all sensitive information in transit.		●	●

10.4 Ensure Access Control and Authentication is Enabled (Manual)

Profile Applicability:

- Level 1

Description:

Utilize AWS Identity and Access Management (IAM) to control access to your Amazon Timestream resources. Define IAM policies that grant or deny permissions for specific Timestream actions and resources.

Rationale:

Users should select whether they like to enable authentication. If they want to authenticate the user would be required to implement IAM roles would grant or deny permissions within that database. Users also have an option to enable multi-factor authentication, which adds an extra layer of security restricting access to unauthorized users.

Impact:

Allowing authentication verifies the identity of the person and who has appropriate access to a company’s data.

Audit:

1. Understand AWS Identity and Access Management (IAM)

   Familiarize yourself with IAM, the AWS service used to manage access to AWS resources. Understand IAM users, groups, roles, policies, and permissions, essential for access control in Timestream.

2. Create IAM Users, Groups, and Roles

   Access the AWS Management Console and navigate to the IAM service. Create IAM users, groups, and roles based on your organization's access control requirements for Timestream. Define appropriate permissions for these entities, limiting access to specific Timestream actions and resources.

3. Assign IAM Policies

   Create IAM policies that define the desired level of access to Timestream. Associate these policies with the respective IAM users, groups, and roles created earlier. Ensure that the policies provide the necessary permissions for users to interact with Timestream resources.

Internal Only - General

4. Use IAM Roles for External Applications

   If you have external applications or services accessing Timestream, create IAM roles specific to those applications. Define the necessary permissions in the IAM roles and grant them to the respective applications or services. Configure the applications or services to assume these IAM roles when accessing Timestream.

5. Enable Multi-Factor Authentication (MFA)

   Enable MFA for IAM users who require access to Timestream. Configure MFA devices and enforce MFA usage for these users. MFA adds an extra layer of security by requiring an additional authentication factor during the login process.

6. Implement AWS Identity Federation (Optional)

   Consider implementing AWS Identity Federation if you need to grant access to Timestream to users from external identity providers. Configure the necessary trust relationships and establish a federation between the external identity provider and AWS. Ensure that the federated users have the appropriate IAM policies and permissions for Timestream.

7. Regularly Review and Update Access Controls

   Periodically review and update the IAM policies and permissions for Timestream. Remove unnecessary access permissions and ensure access controls align with your organization's security requirements. Monitor IAM activity logs and AWS CloudTrail to identify unauthorized access attempts or unusual activities.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user’s need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●

Controls Version	Control	IG 1	IG 2	IG 3
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

10.5 Ensure Fine-Grained Access Control is Enabled (Manual)

Profile Applicability:

- Level 1

Description:

Leverage Timestream's fine-grained access control capabilities to control table or row level access. Define access policies that limit access to specific tables, columns, or rows based on user roles or conditions. Implement data filtering and row-level security to restrict access to sensitive information.

Rationale:

This helps by having specific permissions which can be denied due to multiple conditions of the database. This allows the user to control certain aspects of the database.

Impact:

This adds an extra layer for users to sign into with their credentials to the database.

Audit:

1. Understand Fine-Grained Access Control in Timestream

   Familiarize yourself with the concept of fine-grained access control and its benefits in Timestream. Understand that fine-grained access control allows you to control access to specific tables, columns, or rows within Timestream databases.

2. Define Timestream Database and Tables

   Create the necessary Timestream databases and tables that will be used for fine-grained access control. Design your database schema and define the tables, columns, and rows that need granular access control.

3. Create IAM Policies for Fine-Grained Access

   Access the AWS Management Console and navigate to the IAM service. Define IAM policies that grant or deny permissions for specific Timestream actions, databases, tables, columns, or rows. Leverage Timestream's fine-grained access control policy language to specify the conditions and restrictions for access.

4. Assign IAM Policies to IAM Users, Groups, or Roles

Internal Only - General

Associate the IAM policies created earlier with the respective IAM users, groups, or roles. Assign the appropriate policies to grant access to specific Timestream databases, tables, columns, or rows. Follow the principle of least privilege and provide only the necessary permissions to users based on their requirements.

5. Test Fine-Grained Access Control

   Validate the fine-grained access control settings by attempting different actions on Timestream databases, tables, columns, or rows. Verify that the defined policies accurately restrict or allow access based on the specified conditions. Perform thorough testing to enforce the expected granularity and security level.

6. Regularly Review and Update Access Policies

   Periodically review the fine-grained access control policies to ensure they align with your organization's security requirements. Remove any unnecessary or outdated policies. Regularly monitor IAM activity logs and AWS CloudTrail to identify any unauthorized access attempts or unusual activities related to fine-grained access control.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

10.6 Ensure Audit Logging is Enabled (Manual)

Profile Applicability:

- Level 1

Description:

Enable AWS CloudTrail to capture and log API calls and activities related to Amazon Timestream. Configure CloudTrail to store the logs in a secure location and regularly review the logs for any unauthorized or suspicious activities.

Rationale:

This captures and saves logs of activities that took place in the database.

Impact:

This reduces risks of any fraud since worker activity is being monitored and tracked.

Audit:

1. Understand Audit Logging in Timestream

   Familiarize yourself with audit logging and its importance in monitoring and tracking activities in Timestream. Understand that audit logs capture API calls and events related to Timestream actions and resources.

2. Enable AWS CloudTrail

   Access the AWS Management Console and navigate to the AWS CloudTrail service. Create a new CloudTrail trail or use an existing one to capture Timestream audit logs. Configure the trail to include Timestream as a data source for logging.

3. Configure CloudTrail Logging Options

   Specify the desired settings for the CloudTrail trail, such as the S3 bucket to store the audit logs and the log file encryption options. Enable logging of management and data events related to Timestream. Configure the trail to capture the necessary information for your audit and compliance requirements.

4. Set Up CloudTrail Notifications and Alerts

   Configure CloudTrail to send notifications or trigger actions based on specific events or conditions. Set up CloudWatch Alarms to monitor and receive notifications for critical Timestream audit events. Define the appropriate alert thresholds and actions to respond to specific events.

Internal Only - General

5. Access and Review Audit Logs

   Access the configured S3 bucket where the Timestream audit logs are stored. Retrieve and review the logs using AWS Management Console, AWS CLI, or any preferred log analysis tools. Analyze the audit logs to track Timestream activities, detect anomalies, and investigate security incidents.

6. Retention and Compliance Considerations

   Determine the appropriate retention period for your Timestream audit logs based on compliance and regulatory requirements. Implement appropriate data lifecycle management policies for your audit logs stored in the S3 bucket. Ensure compliance with data protection and privacy regulations applicable to your organization.

7. Regularly Review and Monitor Audit Logs

   Establish a regular review process for your Timestream audit logs. Monitor the logs for unauthorized access attempts, unusual activities, or policy violations. Respond promptly to any identified security incidents or anomalies.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	8.1 Establish and Maintain an Audit Log Management Process Establish and maintain an audit log management process that defines the enterprise's logging requirements. At a minimum, address the collection, review, and retention of audit logs for enterprise assets. Review and update documentation annually, or when significant enterprise changes occur that could impact this Safeguard.	●	●	●
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●

10.7 Ensure Regular Updates and Patches are Installed (Manual)

Profile Applicability:

- Level 1

Description:

Stay updated with the latest security patches and updates provided by AWS for Amazon Timestream. Follow AWS security best practices and recommendations to ensure your Timestream implementation remains secure.

Rationale:

Impact:

This helps the organization reduce their security risk by regularly updating and patching their database and database engine. Regularly updating and scanning for any weaknesses in the company can bring up possible vulnerabilities that could have led to potential cyber-attack.

Audit:

1. Stay Informed about Updates

   Stay updated with the latest announcements and releases related to Amazon Timestream. Subscribe to AWS notifications, blogs, and forums to learn about new features, enhancements, and security patches.

2. Review AWS Documentation

   Regularly review the official AWS documentation for Amazon Timestream. Pay attention to any updates or recommendations related to security, performance, and best practices.

3. Implement a Patch Management Process

   Establish a patch management process specific to Amazon Timestream within your organization. Define roles and responsibilities for managing patches, including testing and deployment procedures.

4. Test Patches in a Non-Production Environment

   Before deploying patches in production, create a non-production environment to test the patches. Set up a replica or a sandbox environment that resembles your production environment. Test the patches thoroughly to ensure they do not introduce compatibility issues or adverse effects.

Internal Only - General

5. Schedule Patching Maintenance Windows

   Identify suitable maintenance windows to apply patches to your Timestream resources. Consider the impact on system availability and plan the maintenance window accordingly. Coordinate with relevant teams and stakeholders to ensure minimal disruption during the patching process.

6. Apply Patches

   Once you have successfully tested the patches in the non-production environment and scheduled a maintenance window. Apply the patches to your production Timestream resources. Follow the recommended patching procedures provided by AWS in the documentation. Ensure you follow any specific instructions or requirements for applying patches to Timestream.

7. Verify Patch Deployment

   After applying patches, monitor the Timestream resources to ensure they function as expected. Conduct thorough testing to validate that the patched resources operate correctly and have not introduced any issues.

8. Regularly Monitor for Updates

   Continuously monitor for new updates, patches, and security bulletins related to Amazon Timestream. Stay informed about any vulnerabilities or critical patches that require immediate attention. Adjust your patch management process and schedule to incorporate new updates and releases.

9. Automate Patch Management (Optional)

   Consider automating the patch management process using AWS tools or third-party solutions. Implement automation scripts or systems that handle patch deployments, testing, and monitoring.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	7 Continuous Vulnerability Management Develop a plan to continuously assess and track vulnerabilities on all enterprise assets within the enterprise's infrastructure, in order to remediate, and minimize, the window of opportunity for attackers. Monitor public and private industry sources for new threat and vulnerability information.			
v7	3 Continuous Vulnerability Management Continuous Vulnerability Management			

Internal Only - General

10.8 Ensure Monitoring and Alerting is Enabled (Manual)

Profile Applicability:

- Level 1

Description:

Utilize Amazon CloudWatch to monitor key metrics, events, and logs related to Amazon Timestream. Set up appropriate alarms and notifications to detect security incidents or abnormal behavior proactively.

Rationale:

This helps the individual know what is being logged within the activity and determine what next step should be if they spot any anomalies.

Audit:

1. Define Monitoring Objectives

   Determine the key metrics, events, and logs you want to monitor in Amazon Timestream. Identify the specific monitoring requirements based on your use case, workload, and business needs.

2. Choose Monitoring Tools

   Evaluate the available monitoring tools for Amazon Timestream, such as AWS CloudWatch, third-party monitoring solutions, or custom-built monitoring systems. Select the monitoring tool that best aligns with your monitoring objectives and requirements.

3. Configure CloudWatch Metrics

   Utilize Amazon CloudWatch to monitor key performance metrics of Timestream. Enable and configure CloudWatch metrics such as database CPU utilization, storage usage, query latency, and other relevant metrics. Set appropriate thresholds for these metrics to trigger alarms and notifications.

4. Create CloudWatch Alarms

   Set up CloudWatch alarms based on your defined thresholds and monitoring objectives. Define the conditions that trigger the alarms, such as CPU utilization exceeding a certain percentage or query latency exceeding a specific threshold. Configure the notification actions for the alarms, such as sending notifications via email, SMS, or triggering automated actions.

5. Enable Enhanced Monitoring (Optional)

Internal Only - General

Consider enabling enhanced monitoring for Timestream, which provides more detailed performance metrics. Configure the enhanced monitoring settings to collect additional metrics that provide deeper insights into the health and performance of Timestream.

6. Configure Log Streams and Filters

   Enable Timestream's integration with AWS CloudWatch Logs. Configure log streams and filters to capture and centralize Timestream logs into CloudWatch Logs. Define relevant log filters to extract and track specific log events for monitoring purposes.

7. Regularly Review and Analyze Monitoring Data

   Continuously review the monitoring data and metrics CloudWatch provides or your chosen monitoring tool. Analyze the data to identify performance bottlenecks, anomalies, or issues in your Timestream implementation. Take necessary actions based on the monitoring insights to optimize performance, improve resource utilization, or troubleshoot issues.

8. Periodically Review and Adjust Monitoring Configuration

   Regularly review your monitoring configuration to ensure it aligns with your evolving requirements and workload. Adjust your monitoring setup, such as adding or modifying metrics, updating alarm thresholds, or incorporating new log filters.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	8.1 Establish and Maintain an Audit Log Management Process Establish and maintain an audit log management process that defines the enterprise's logging requirements. At a minimum, address the collection, review, and retention of audit logs for enterprise assets. Review and update documentation annually, or when significant enterprise changes occur that could impact this Safeguard.	●	●	●

Internal Only - General

Controls Version	Control	IG 1	IG 2	IG 3
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●

10.9 Ensure to Review and Update the Security Configuration (Manual)

Profile Applicability:

- Level 1

Description:

Conduct regular security reviews and assessments of your Amazon Timestream implementation. Evaluate access permissions, encryption settings, and security controls to ensure they align with your organization's security requirements.

Rationale:

By regularly reviewing security configuration it helps the businesses to detect any threat they might be hindering and address the threat in a timely manner.

Impact:

This helps by reviewing the database factors from database engine, review instance details, security networks, encryption settings, audit logging, and authentication. By updating or removing a few things from these lists it helps tighten security and ensures that the users do not have excessive permissions.

Audit:

1. Understand Security Best Practices

   Familiarize yourself with the security best practices and recommendations provided by AWS for Timestream. Stay updated with the latest security guidelines and recommendations from AWS.

2. Review IAM Policies

   Regularly review the IAM policies associated with Timestream resources. Ensure that the assigned IAM policies provide the necessary permissions for users and roles while adhering to the principle of least privilege.

3. Audit User Access

   Periodically review the list of users and roles that have access to Timestream. Remove any unnecessary or unused accounts or permissions to minimize the attack surface.

4. Monitor Access Patterns

Internal Only - General

Utilize AWS CloudTrail and Amazon CloudWatch logs to monitor access patterns and activities related to Timestream. Set up alerts and notifications to detect any suspicious or unauthorized access attempts.

5. Implement Security Controls

   Continuously assess and evaluate the security controls in place for Timestream. Implement additional security measures, such as VPC peering, security groups, or network ACLs, to further secure access to Timestream resources.

6. Regularly Review Security Group Rules

   Regularly review the security group rules associated with Timestream instances. Remove any unnecessary open ports or protocols to minimize potential attack vectors.

7. Stay Informed about Security Updates

   Keep track of security updates, patches, and new features released by AWS for Timestream. Stay informed about any security vulnerabilities or fixes related to Timestream.

8. Conduct Security Assessments

   Perform periodic security assessments on your Timestream implementation, including vulnerability and penetration testing. Identify and remediate any security vulnerabilities or weaknesses discovered during the assessments.

9. Stay Compliant

   Regularly review and update your security configurations to meet compliance requirements and industry standards. Stay informed about any changes in compliance regulations that may impact your Timestream environment.

10. Educate and Train

    Provide regular security awareness training to users and administrators working with Timestream. Ensure that everyone involved understands their security responsibilities and follows security best practices.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	5 Account Management Use processes and tools to assign and manage authorization to credentials for user accounts, including administrator accounts, as well as service accounts, to enterprise assets and software.			
v7	5 Secure Configuration for Hardware and Software on Mobile Devices, Laptops, Workstations and Servers Secure Configuration for Hardware and Software on Mobile Devices, Laptops, Workstations and Servers			

10.10 Ensure Database has automated Backups enabled (Manual)

Profile Applicability:

- Level 1

Description:

Ensure that Amazon Timestream tables have automated backups enabled through AWS Backup with a defined backup schedule and retention policy. AWS Backup provides scheduled, automated backup functionality for Timestream tables, creating regular point-in-time snapshots that are retained according to a configurable lifecycle policy.

Rationale:

Amazon Timestream stores critical time-series data that is often mission-critical for monitoring, analytics, and operational intelligence. Automated backups through AWS Backup ensure that Timestream tables are continuously protected without requiring manual intervention, and can be rapidly restored in the event of accidental deletion, data corruption, misconfiguration, or application errors.

Impact:

Enabling automated backups for Timestream ensures that time-series data is regularly captured in durable backups and recoverable to any point within the configured retention window, providing strong protection against accidental loss and data corruption.

Audit:

Important Note: Amazon Timestream does not have a native automated backup feature built into the service. Instead, backups are managed through AWS Backup, which provides scheduled, on-demand, and lifecycle-managed backup functionality for Timestream tables.

Check if automated backups are enabled via AWS Backup Service:

1. Check if Timestream Database is Assigned to a Backup Plan

- List backup plans:

```
aws backup list-backup-plans --query "BackupPlansList[].BackupPlanName" --
output table
```

- For each backup plan, list all backup selections (resource assignments):

```
aws backup list-backup-selections --backup-plan-id <your-backup-plan-id> --
query "BackupSelections[].SelectionId" --output text
```

- For each selection, list assigned resources and search for your database:

```
aws backup get-backup-selection --backup-plan-id <your-backup-plan-id> --
selection-id <selection-id> --query "BackupSelection.Resources" --output text
```

- If your table ARN appears in any selection, it is protected by the backup plan.

2. Verify Backup Plan Configuration

```
aws backup get-backup-plan --backup-plan-id <your-backup-plan-id>
```

Look for the "Lifecycle" fields in each backup rule:

- "DeleteAfterDays" is the retention period.
- "ScheduleExpression" sets the backup schedule (cron format).
- "BackupVaultName" is the name of the vault (where backups are stored).

Remediation:

1. Create Backup Plan for on-demand snapshots:

```
aws backup create-backup-plan --backup-plan '{
  "BackupPlanName": "<BackupPlanName>",
  "Rules": [
    {
      "RuleName": "Scheduled-OnDemand-Snapshots",
      "TargetBackupVaultName": "Default",
      "ScheduleExpression": "cron(0 3 ? * SUN *)",
      "StartWindowMinutes": 120,
      "CompletionWindowMinutes": 360,
      "Lifecycle": { "DeleteAfterDays": 90 },
      "RecoveryPointTags": { "BackupType": "OnDemand" }
    }
  ]
}'
```

- Replace "BackupPlanName" with your backup plan name
- Replace "Default" with your actual backup vault name if different
- Update the "ScheduleExpression", "StartWindowMinutes" based on your unique backup schedule.
- Replace "35" with your retention time for PITR if shorter
- Replace "90" with your retention time for Snapshots if different

This command outputs the BackupPlanId necessary for the next step.

2. Assign Timestream Database to the Backup Plan

Replace <BackupPlanId> with the ID from Step 1 and <TableArn> with your Timestream table ARN.

```
aws backup create-backup-selection --backup-plan-id <BackupPlanId> --backup-selection '{ "SelectionName": "timestream-tables", "IamRoleArn": "arn:aws:iam:::role/AWSBackupServiceRolePolicyForBackup", "Resources": ["<TableArn>"] }'
```

- Make sure the IAM role has the necessary AWS Backup permissions.
- These commands create a centralized backup plan with scheduled snapshot backups, then assign your Timestream database as a resource for backup.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	11 Data Recovery Establish and maintain data recovery practices sufficient to restore in-scope enterprise assets to a pre-incident and trusted state.			

11 Amazon Ledger Database Services (QLDB)

Internal Only - General

11.1 Ensure to Implement Identity and Access Management (IAM) (Manual)

Profile Applicability:

- Level 1

Description:

This control is important because by having IAM roles implemented in the database it only allows certain people who are authenticated into the database to modify the database and would not give access to unauthorized personnel. This ensures that the data is being protected from any threat actor.

Rationale:

Impact:

Only authorized personnel can access the database and configure the applications by using their IAM credentials. If the user credentials are compromised by an unauthorized user, it would limit them to access specific areas within the database due to the leverage IAM roles established.

Audit:

1. Understand IAM and QLDB Integration

- Familiarize yourself with IAM and its role in controlling access to AWS services, including QLDB.
- Understand how IAM policies define permissions and access control rules for QLDB resources.

2. Define IAM Users and Groups

- Identify the users and groups that will need access to QLDB.
- Create IAM user accounts for individuals who require direct access to QLDB.
- Create IAM groups to organize users based on their roles or responsibilities logically.

3. Define IAM Policies

- Determine the permissions and actions users and groups need to perform on QLDB resources.
- Create custom IAM policies or leverage existing IAM-managed policies to define these permissions.
- Consider the principle of least privilege and grant only the necessary permissions for each user or group.

4. Attach IAM Policies to Users and Groups

- Associate the appropriate IAM policies with the IAM users and groups.
- Ensure that each user or group has the necessary permissions to perform their tasks on QLDB.
- Regularly review and update the assigned policies as access requirements evolve.

5. Leverage IAM Roles

- Identify AWS services or applications that require access to QLDB.
- Create IAM roles to provide temporary credentials and permissions for these services.
- Define trust relationships and establish the necessary permissions in the IAM role policies.

6. Enable IAM Database Authentication

- Configure IAM database authentication for QLDB to allow users to authenticate using their IAM credentials.
- Enable the appropriate IAM authentication option in the QLDB configuration.
- Configure your applications or clients to use IAM credentials when connecting to QLDB.

7. Test IAM Access

- Use IAM user credentials to log in and test the access to QLDB.
- Verify that users can perform their intended actions based on their assigned IAM policies.
- Test IAM roles and authentication for applications or services requiring access to QLDB.

8. Monitor and Audit IAM Activity

- Monitor IAM activity logs using AWS CloudTrail.
- Set up appropriate CloudTrail trails to capture IAM-related events and API calls.
- Regularly review IAM logs for any unauthorized access attempts or suspicious activities.

9. Regularly Review and Update IAM Configuration

- Periodically review the IAM policies, users, groups, and roles associated with QLDB.
- Ensure access is granted based on business requirements and follows the principle of least privilege.

Internal Only - General

- Remove or update IAM configurations when users or roles are no longer required.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

Internal Only - General

11.2 Ensure Network Access is Secure (Manual)

Profile Applicability:

- Level 1

Description:

By applying certain network access such as Virtual Private Cloud (VPC) it protects the private network that has been established from any external networks from interfering. It allows internal networks to communicate with one another with the network that has been established. The Access Control List (ACLs) allows only specific individuals to access the resources. Also, by monitoring and logging the activity within the database it helps the individual know what is being logged within the activity and determine what next step they should take to address it.

Rationale:

Impact:

Setting these certain rules in your network provides a strong security and prevents the organization suffering a ransomware attack.

Audit:

1. Deploy QLDB in a VPC

- Create a Virtual Private Cloud (VPC) to isolate your QLDB resources.
- Define the network CIDR blocks, subnets, and routing configurations for the VPC.
- Ensure that the VPC is correctly configured with appropriate network access controls.

2. Configure Security Groups

- Create security groups within your VPC to control inbound and outbound traffic to QLDB.
- Determine the necessary protocols and ports for QLDB access.
- Configure security group rules to allow access from trusted sources, such as specific IP ranges or other security groups within your VPC.

3. Set Up Network ACLs

- Configure Network Access Control Lists (ACLs) within your VPC to provide an additional layer of network security.
- Define inbound and outbound rules in the ACLs to allow or deny specific traffic based on IP addresses, ports, or protocols.

Internal Only - General

- Review and adjust the ACL rules to align with your organization's security policies and requirements.

4. Use VPC Endpoints or PrivateLink

- Consider using VPC endpoints or AWS PrivateLink to securely access QLDB without traversing the public internet.
- Set up a VPC endpoint for QLDB to allow private connectivity within your VPC.
- Configure the routing and security group rules to enable traffic flow through the VPC endpoint or PrivateLink.

5. Secure External Connections

- If external connections to QLDB are required, implement secure access methods such as Virtual Private Network (VPN) or AWS Direct Connect.
- Configure VPN connections or Direct Connect links to establish encrypted and private connectivity between your on-premises network and the VPC hosting QLDB.
- Apply appropriate security measures, such as strong authentication and encryption, to protect data transmitted over external connections.

6. Enable Logging and Monitoring

- Enable logging for QLDB to capture important system events and database activity.
- Utilize services like Amazon CloudWatch Logs to centralize and analyze QLDB logs.
- Set up appropriate alarms and notifications to alert you of any suspicious network activity or potential security incidents.

7. Regularly Review and Update Network Security

- Regularly review your VPC configurations, security groups, and network ACLs.
- Stay informed about AWS security best practices and recommendations.
- Update your network security measures as needed to address emerging threats or changes in your security requirements.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	12.2 Establish and Maintain a Secure Network Architecture Establish and maintain a secure network architecture. A secure network architecture must address segmentation, least privilege, and availability, at a minimum.		●	●
v7	11.7 Manage Network Infrastructure Through a Dedicated Network Manage the network infrastructure across network connections that are separated from the business use of that network, relying on separate VLANs or, preferably, on entirely different physical connectivity for management sessions for network devices.		●	●

11.3 Ensure Data at Rest is Encrypted (Manual)

Profile Applicability:

- Level 1

Description:

This helps ensure that the data is kept secure and protected when at rest. The user must choose from two key options which then determine when the data is encrypted at rest.

Rationale:

Impact:

If an unauthorized user steals the data, it would be unreadable for them because a key would be required to decrypt the message into plaintext.

Audit:

1. Create an AWS Key Management Service (KMS) Key

- Sign in to the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.
- Open the AWS Key Management Service (KMS) console.
- Create a new KMS key or select an existing one to encrypt your QLDB data at rest.
- Configure the key policy to grant the appropriate IAM users or role permissions.

2. Enable Encryption for QLDB

- Open the Amazon QLDB console.
- Choose the QLDB ledger for which you want to enable encryption at rest.
- Click on the `Configuration` tab.
- Under the `Encryption` section.
- Click on the `Edit` button or `Modify` option.
- Enable encryption for the ledger.
- Select the KMS key you created or chose in the first step for encrypting the QLDB data.
- Save the changes to enable encryption at rest for the QLDB ledger.

3. Verify Encryption Status

- Once the encryption at rest is enabled, the QLDB console will indicate the encryption status as `Enabled` for the selected ledger.
- Ensure that the KMS key specified for encryption is the correct key you intended to use.

4. Testing and Verification

- Perform read and write operations on your QLDB ledger to validate that the data is encrypted at rest.
- Verify that you can access and query the encrypted data using appropriate authentication and authorization methods.

5. Key Management and Rotation

- Follow AWS best practices for key management, including securely storing and managing the KMS key used for QLDB encryption.
- Implement a key rotation policy, following AWS recommendations and compliance requirements if required.

6. Backup and Disaster Recovery

- Ensure you have appropriate backup and disaster recovery mechanisms for your QLDB data.
- Consider backing up the KMS key used for encryption to prevent data loss in case of a key compromise or accidental deletion.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.11 Encrypt Sensitive Data at Rest Encrypt sensitive data at rest on servers, applications, and databases containing sensitive data. Storage-layer encryption, also known as server-side encryption, meets the minimum requirement of this Safeguard. Additional encryption methods may include application-layer encryption, also known as client-side encryption, where access to the data storage device(s) does not permit access to the plain-text data.		●	●
v7	14.8 Encrypt Sensitive Information at Rest Encrypt all sensitive information at rest using a tool that requires a secondary authentication mechanism not integrated into the operating system, in order to access the information.			●

11.4 Ensure Data in Transit is Encrypted (Manual)

Profile Applicability:

- Level 1

Description:

Use Transport Layer Security (TLS) to encrypt communications between clients and your QLDB instance. QLDB provides TLS support by default, allowing secure communication over the network. Configure your client applications to use TLS when connecting to QLDB.

Rationale:

Amazon Quantum Ledger Database (QLDB), uses TLS to encrypt data during transit. To secure your data in transit the individual should identify their client application and what is supported by TLS in order to configure it correctly.

Impact:

If the user does not have the code configured correctly it would not be able to connect to the server.

Audit:

1. Understand TLS Encryption for QLDB

- Learn about Transport Layer Security (TLS) and its role in securing data during transit.
- Understand how TLS works to establish secure communication channels between clients and QLDB.

2. Configure Clients for TLS Encryption

- Ensure that your client applications support TLS encryption for communication with QLDB.
- Use the appropriate AWS SDK or QLDB driver that provides TLS encryption support.
- Update your application code or configurations to enable TLS encryption.

3. Obtain the QLDB Endpoint

- Sign in to the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.
- Open the Amazon QLDB console.
- Locate the QLDB ledger for which you want to enable encryption in transit.
- Note down the QLDB endpoint for your ledger.

4. Establish TLS Connection

- Use the QLDB endpoint obtained earlier to establish a TLS connection between your client application and QLDB.
- Configure your client application to connect to QLDB using the secure HTTPS protocol.
- Provide the necessary authentication credentials or tokens required to establish the connection.

5. Verify TLS Encryption

- Once the TLS connection is established, verify that the connection is secured using TLS by checking for a valid TLS certificate.
- Ensure that your client application can communicate securely with QLDB without any errors or warnings related to encryption.

6. Regularly Update Client Applications

- Stay updated with the latest versions of the AWS SDKs or QLDB drivers used by your client applications.
- Regularly update your client applications to leverage the latest TLS encryption features and security enhancements.

7. Monitor and Review TLS Connections

- Utilize AWS CloudTrail and Amazon CloudWatch to monitor and log TLS-related events and errors.
- Review the logs and alerts to identify potential security issues or anomalies related to TLS connections.

8. Secure Other Communication Channels

- Ensure that other communication channels your client applications use, such as APIs or data transfers, also utilize TLS encryption.
- Implement appropriate encryption and security measures to protect sensitive data during transit in all communication channels.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.10 Encrypt Sensitive Data in Transit Encrypt sensitive data in transit. Example implementations can include: Transport Layer Security (TLS) and Open Secure Shell (OpenSSH).		●	●
v7	14.4 Encrypt All Sensitive Information in Transit Encrypt all sensitive information in transit.		●	●

11.5 Ensure to Implement Access Control and Authentication (Manual)

Profile Applicability:

- Level 1

Description:

Utilize QLDB's built-in authentication and access control mechanisms. Define IAM policies to control which users or roles can perform specific actions on QLDB resources. Leverage IAM roles for cross-service access, securely integrating QLDB with other AWS services.

Rationale:

Users should select whether they like to enable authentication. If they want to authenticate the user would be required to implement IAM roles would grant or deny permissions within that database.

Impact:

Allowing authentication verifies the identity of the person and who has appropriate access to a company's data.

Audit:

1. Understand QLDB Authentication and Access Control

- Familiarize yourself with the authentication and access control mechanisms provided by QLDB.
- Understand the concepts of users, permissions, and roles in QLDB's access control model.

2. Configure IAM for QLDB

- Sign in to the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.
- Open the Amazon QLDB console.
- Go to the `Ledgers` section.
- Select the QLDB ledger for which you want to configure access control.
- Under the `Configuration` tab.
- Click on `Edit` or `Modify` to make changes.
- Enable IAM-based authentication by selecting the appropriate option.
- Define the IAM policies that grant or deny permissions for specific QLDB actions.
- Configure fine-grained access control by associating IAM policies with IAM users or roles.

Internal Only - General

3. Create IAM Users or Roles

- Identify the individuals or services that require access to QLDB.
- Create IAM user accounts for individuals or IAM roles for services.
- Assign appropriate IAM policies to these users or roles based on their required access levels.

4. Grant Required Permissions

- Define IAM policies that grant the necessary permissions for QLDB operations.
- Consider the principle of least privilege and only provide the minimum permissions required for each user or role.
- Assign IAM policies to IAM users or roles to allow access to specific QLDB resources.

5. Test Access Control

- Use IAM user credentials or IAM role credentials to test access to QLDB resources.
- Verify that users or services can perform the expected actions based on their assigned IAM policies.
- Test both read and write operations to ensure appropriate access permissions.

6. Monitor and Audit Access

- Enable AWS CloudTrail for QLDB to capture and log all API calls and activities.
- Use Amazon CloudWatch to monitor and analyze the logs for unauthorized access attempts or suspicious activities.
- Implement additional logging and auditing mechanisms as per your organization's security requirements.

7. Regularly Review and Update Access Control

- Conduct periodic reviews of IAM policies, users, and roles associated with QLDB.
- Remove or update access for users or roles that no longer require QLDB access.
- Stay updated with AWS security best practices and IAM and access control recommendations.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

Internal Only - General

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	3.3 Configure Data Access Control Lists Configure data access control lists based on a user's need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.	●	●	●
v7	14.6 Protect Information through Access Control Lists Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.	●	●	●

Internal Only - General

11.6 Ensure Monitoring and Logging is Enabled (Manual)

Profile Applicability:

- Level 1

Description:

Enable QLDB's built-in logging to capture important system events and database activity. Monitor the logs for any suspicious activities or errors. Leverage Amazon CloudWatch to collect and analyze logs, set up alarms, and receive notifications for potential security incidents.

Rationale:

This helps the individual know what is being logged within the activity and determine what next step they should take to address it.

Audit:

1. Enable AWS CloudTrail

- Sign in to the AWS Management Console at https://console.aws.amazon.com/ with your AWS account credentials.
- Open the AWS CloudTrail console.
- Create a new trail or select an existing trail.
- Configure the trail to capture QLDB API calls and relevant events.
- Specify the Amazon S3 bucket where the CloudTrail logs will be stored.
- Enable the trail to start capturing QLDB events.

2. Enable Amazon CloudWatch Logs

- Open the Amazon CloudWatch console.
- Create a new log group or select an existing log group.
- Configure the log group to receive QLDB logs from CloudTrail.
- Define the log retention period to retain the logs for the desired duration.
- Enable CloudWatch Logs to start receiving and storing QLDB logs.

3. Configure Log Metric Filters

- In the CloudWatch console, go to the log group that contains the QLDB logs.
- Define log metric filters to extract specific information or patterns from the logs.
- Create metric filters based on your monitoring and alerting requirements.
- Specify the target metric and define the filter patterns to match the desired log events.

4. Create CloudWatch Dashboards and Alarms

Internal Only - General

- Create CloudWatch dashboards to visualize and monitor important QLDB metrics.
- Customize the dashboard widgets to display relevant log metrics, such as API calls or errors.
- Set up CloudWatch alarms to trigger notifications or automated actions based on specific thresholds or conditions.
- Configure alarm actions, such as sending email notifications or invoking AWS Lambda functions, to respond to critical events.

5. Enable EventBridge Integration (Optional)

- Open the Amazon EventBridge console.
- Create a new rule or select an existing rule.
- Configure the rule to match specific QLDB events or patterns.
- Define targets for the rule, such as invoking Lambda functions or sending notifications to other AWS services.

6. Monitor and Analyze Logs and Metrics

- Regularly review the CloudWatch logs and metrics for QLDB.
- Monitor key metrics and performance indicators to identify any issues or anomalies.
- Use CloudWatch Logs Insights to query and analyze log data for troubleshooting.

7. Integrate with AWS Monitoring and Alerting Tools

- Leverage other AWS monitoring and alerting services like AWS X-Ray or AWS ServiceLens to gain deeper insights into QLDB performance and behavior.
- Configure additional alerts or notifications using AWS services like Amazon SNS or AWS Chatbot.

8. Regularly Review and Update Logging and Monitoring Configuration

- Periodically review and update your CloudTrail, CloudWatch, and EventBridge configurations to align with changes in your monitoring requirements.
- Stay informed about AWS security best practices and new features.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	8.1 Establish and Maintain an Audit Log Management Process Establish and maintain an audit log management process that defines the enterprise’s logging requirements. At a minimum, address the collection, review, and retention of audit logs for enterprise assets. Review and update documentation annually, or when significant enterprise changes occur that could impact this Safeguard.	●	●	●
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●

11.7 Ensure to Enable Backup and Recovery (Manual)

Profile Applicability:

- Level 1

Description:

Having the data backed up ensures that all the crucial information is stored securely it defends against any human errors and system errors that resulted in data loss. An organization that has a disaster recovery plan is prepared for any disruption that would impact business operations.

Rationale:

Impact:

If a business does not have a backup and recovery plan it would have a negative impact on the business, which would result in less productivity, suffer data loss that cannot be restored, and loss of revenue.

Audit:

1. Understand QLDB Backup and Recovery Features

- Familiarize yourself with the built-in backup and recovery capabilities provided by QLDB.
- Understand the concepts of ledgers, revisions, and journal export for backup and restore operations.

2. Determine Backup and Recovery Requirements

- Assess your organization's backup and recovery requirements for QLDB.
- Define the recovery point objective (RPO) and recovery time objective (RTO) that align with your business needs.
- Determine the desired backup frequency and retention period for your QLDB data.

3. Enable Automatic Backups

- Open the Amazon QLDB console.
- Select the QLDB ledger for which you want to enable automatic backups.
- Click on the `Configuration` tab.
- Under the `Backup` section, enable automatic backups.
- Specify the desired backup retention period for the automatic backups.

4. Perform Manual Backups (Optional)

Internal Only - General

- If you need additional backups or want to perform on-demand backups, initiate manual backups.
- Open the Amazon QLDB console.
- Select the QLDB ledger you want to back up.
- Click on the `Backups` tab.
- Choose the `Create Backup` option.
- Provide a meaningful backup name and initiate the backup process.

5. Restore QLDB from Backups

- Open the Amazon QLDB console.
- Go to the `Backups` tab.
- Select the backup from which you want to restore the QLDB ledger.
- Click on the `Restore` option.
- Specify the desired restoration name and initiate the restoration process.

6. Regularly Test Restore Process

- Periodically test the restore process to ensure that backups are working correctly.
- Select a backup and initiate the restoration to a separate QLDB ledger.
- Verify that the restored ledger contains the expected data and is accessible.

7. Implement Data Archiving (Optional)

- If you require long-term data retention or compliance with specific data retention policies, consider implementing data archiving strategies.
- Leverage AWS services like Amazon S3 for long-term storage of QLDB journal exports or backups.

8. Disaster Recovery Planning

- Develop a comprehensive disaster recovery plan for QLDB to mitigate the impact of catastrophic events.
- Consider implementing cross-region replication or multi-region deployments to provide geographic redundancy.
- Test the disaster recovery plan periodically to validate its effectiveness.

9. Monitor Backup and Recovery Operations

- Regularly monitor backup and recovery operations using Amazon CloudWatch and AWS CloudTrail.
- Set up appropriate alarms and notifications to ensure timely identification of any backup or recovery issues.

Remediation:

References:

1. https://aws.amazon.com/products/databases/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	11 Data Recovery Establish and maintain data recovery practices sufficient to restore in-scope enterprise assets to a pre-incident and trusted state.			
v7	10 Data Recovery Capabilities Data Recovery Capabilities			

Internal Only - General

Appendix: Summary Table

CIS Benchmark Recommendation		Set Correctly	
		Yes	No
1	**Introduction**		
2	**Amazon Aurora**		
2.1	Ensure the Use of Security Groups (Manual)	□	□
2.2	Ensure Data at Rest is Encrypted (Manual)	□	□
2.3	Ensure Data in Transit Encryption is Enforced (Manual)	□	□
2.4	Ensure IAM Roles and Policies are Created (Manual)	□	□
2.5	Ensure Database Audit Logging is Enabled (Manual)	□	□
2.6	Ensure Passwords are Regularly Rotated (Manual)	□	□
2.7	Ensure Least Privilege Access (Manual)	□	□
2.8	Ensure Automatic Backups and Retention Policies are configured (Manual)	□	□
2.9	Ensure Database is not Publicly accessible (Manual)	□	□
2.10	Ensure Database has IAM Auth is Enabled (Manual)	□	□
2.11	Ensure Database has delete protection enabled (Manual)	□	□
3	**Amazon RDS**		
3.1	Ensure to Choose the Appropriate Database Engine (Manual)	□	□
3.2	Ensure to Create The Appropriate Deployment Configuration (Manual)	□	□
3.3	Ensure to Create a Virtual Private Cloud (VPC) (Manual)	□	□
3.4	Ensure to Configure Security Groups (Manual)	□	□

CIS Benchmark Recommendation		Set Correctly	
		Yes	No
3.5	Enable Encryption at Rest (Manual)	□	□
3.6	Enable Encryption in Transit (Manual)	□	□
3.7	Ensure to Implement Access Control and Authentication (Manual)	□	□
3.8	Ensure to Regularly Patch Systems (Manual)	□	□
3.9	Ensure Monitoring and Logging is Enabled (Manual)	□	□
3.10	Ensure to Enable Backup and Recovery (Manual)	□	□
3.11	Ensure to Regularly Review Security Configuration (Manual)	□	□
3.12	Ensure Database is not Publicly accessible (Manual)	□	□
3.13	Ensure Database has IAM Auth is Enabled (Manual)	□	□
3.14	Ensure Database has delete protection enabled (Manual)	□	□
4	**Amazon DynamoDB**		
4.1	Ensure AWS Identity and Access Management (IAM) is in use (Manual)	□	□
4.2	Ensure Fine-Grained Access Control is implemented (Manual)	□	□
4.3	Ensure DynamoDB Encryption at Rest (Manual)	□	□
4.4	Ensure DynamoDB Encryption in Transit (Manual)	□	□
4.5	Ensure VPC Endpoints are configured (Manual)	□	□
4.6	Ensure DynamoDB Streams and AWS Lambda for Automated Compliance Checking is Enabled (Manual)	□	□
4.7	Ensure Monitor and Audit Activity is enabled (Manual)	□	□

CIS Benchmark Recommendation		Set Correctly	
		Yes	No
4.8	Ensure Database has delete protection enabled (Manual)	□	□
4.9	Ensure Database has Backup enabled (Manual)	□	□
5	**Amazon ElastiCache**		
5.1	Ensure Secure Access to ElastiCache (Manual)	□	□
5.2	Ensure Network Security is Enabled (Manual)	□	□
5.3	Ensure Encryption at Rest and in Transit is configured (Manual)	□	□
5.4	Ensure Automatic Updates and Patching are Enabled (Manual)	□	□
5.5	Ensure Virtual Private Cloud (VPC) is Enabled (Manual)	□	□
5.6	Ensure Monitoring and Logging is Enabled (Manual)	□	□
5.7	Ensure Security Configurations are Reviewed Regularly (Manual)	□	□
5.8	Ensure Authentication and Access Control is Enabled (Manual)	□	□
5.9	Ensure Audit Logging is Enabled (Manual)	□	□
5.10	Ensure Security Configurations are Reviewed Regularly (Manual)	□	□
5.11	Ensure ElastiCache has Cluster Mode Enabled (Manual)	□	□
5.12	Ensure ElastiCache is deployed across multiple Availability Zones (AZs) (Manual)	□	□
5.13	Ensure ElastiCache has automatic backups enabled (Manual)	□	□
6	**Amazon MemoryDB for Redis**		

CIS Benchmark Recommendation		Set Correctly	
		Yes	No
6.1	Ensure Network Security is Enabled (Manual)	□	□
6.2	Ensure Data at Rest and in Transit is Encrypted (Manual)	□	□
6.3	Ensure Authentication and Access Control is Enabled (Manual)	□	□
6.4	Ensure Audit Logging is Enabled (Manual)	□	□
6.5	Ensure Security Configurations are Reviewed Regularly (Manual)	□	□
6.6	Ensure Monitoring and Alerting is Enabled (Manual)	□	□
6.7	Ensure MemoryDB has automatic backups enabled (Manual)	□	□
7	**Amazon DocumentDB**		
7.1	Ensure Network Architecture Planning (Manual)	□	□
7.2	Ensure VPC Security is Configured (Manual)	□	□
7.3	Ensure Encryption at Rest is Enabled (Manual)	□	□
7.4	Ensure Encryption in Transit is Enabled (Manual)	□	□
7.5	Ensure to Implement Access Control and Authentication (Manual)	□	□
7.6	Ensure Audit Logging is Enabled (Manual)	□	□
7.7	Ensure Regular Updates and Patches (Manual)	□	□
7.8	Ensure to Implement Monitoring and Alerting (Manual)	□	□
7.9	Ensure to Implement Backup and Disaster Recovery (Manual)	□	□
7.10	Ensure to Configure Backup Window (Manual)	□	□

Internal Only - General

CIS Benchmark Recommendation		Set Correctly	
		Yes	No
7.11	Ensure to Conduct Security Assessments (Manual)	□	□
7.12	Ensure DocumentDB has delete protection enabled (Manual)	□	□
8	**Amazon Keyspaces (formerly Amazon Managed Apache Cassandra Service)**		
8.1	Ensure Keyspace Security is Configured (Manual)	□	□
8.2	Ensure Network Security is Enabled (Manual)	□	□
8.3	Ensure Data at Rest and in Transit is Encrypted (Manual)	□	□
8.4	Ensure Amazon Keyspaces tables have Point-in-Time Recovery (PITR) enabled (Manual)	□	□
9	**Amazon Neptune**		
9.1	Ensure Network Security is Enabled (Manual)	□	□
9.2	Ensure Data at Rest is Encrypted (Manual)	□	□
9.3	Ensure Data in Transit is Encrypted (Manual)	□	□
9.4	Ensure Authentication and Access Control is Enabled (Manual)	□	□
9.5	Ensure Audit Logging is Enabled (Manual)	□	□
9.6	Ensure Security Configurations are Reviewed Regularly (Manual)	□	□
9.7	Ensure Monitoring and Alerting is Enabled (Manual)	□	□
9.8	Ensure Neptune Database is not Publicly accessible (Manual)	□	□
9.9	Ensure Neptune Database has automatic backups enabled (Manual)	□	□

CIS Benchmark Recommendation		**Set Correctly**	
		Yes	No
9.10	Ensure Database has delete protection enabled (Manual)	□	□
9.11	Ensure Neptune DB instances are deployed across multiple Availability Zones (AZs) (Manual)	□	□
10	**Amazon Timestream**		
10.1	Ensure Data Ingestion is Secure (Manual)	□	□
10.2	Ensure Data at Rest is Encrypted (Manual)	□	□
10.3	Ensure Encryption in Transit is Configured (Manual)	□	□
10.4	Ensure Access Control and Authentication is Enabled (Manual)	□	□
10.5	Ensure Fine-Grained Access Control is Enabled (Manual)	□	□
10.6	Ensure Audit Logging is Enabled (Manual)	□	□
10.7	Ensure Regular Updates and Patches are Installed (Manual)	□	□
10.8	Ensure Monitoring and Alerting is Enabled (Manual)	□	□
10.9	Ensure to Review and Update the Security Configuration (Manual)	□	□
10.10	Ensure Database has automated Backups enabled (Manual)	□	□
11	**Amazon Ledger Database Services (QLDB)**		
11.1	Ensure to Implement Identity and Access Management (IAM) (Manual)	□	□
11.2	Ensure Network Access is Secure (Manual)	□	□
11.3	Ensure Data at Rest is Encrypted (Manual)	□	□

CIS Benchmark Recommendation		Set Correctly	
		Yes	No
11.4	Ensure Data in Transit is Encrypted (Manual)	□	□
11.5	Ensure to Implement Access Control and Authentication (Manual)	□	□
11.6	Ensure Monitoring and Logging is Enabled (Manual)	□	□
11.7	Ensure to Enable Backup and Recovery (Manual)	□	□

Internal Only - General

Appendix: CIS Controls v7 IG 1 Mapped Recommendations

Recommendation		Set Correctly	
		Yes	No
2.1	Ensure the Use of Security Groups	□	□
2.4	Ensure IAM Roles and Policies are Created	□	□
2.5	Ensure Database Audit Logging is Enabled	□	□
2.7	Ensure Least Privilege Access	□	□
3.1	Ensure to Choose the Appropriate Database Engine	□	□
3.4	Ensure to Configure Security Groups	□	□
3.7	Ensure to Implement Access Control and Authentication	□	□
4.1	Ensure AWS Identity and Access Management (IAM) is in use	□	□
4.2	Ensure Fine-Grained Access Control is implemented	□	□
4.7	Ensure Monitor and Audit Activity is enabled	□	□
5.1	Ensure Secure Access to ElastiCache	□	□
5.6	Ensure Monitoring and Logging is Enabled	□	□
5.8	Ensure Authentication and Access Control is Enabled	□	□
5.9	Ensure Audit Logging is Enabled	□	□
6.3	Ensure Authentication and Access Control is Enabled	□	□
6.4	Ensure Audit Logging is Enabled	□	□
6.6	Ensure Monitoring and Alerting is Enabled	□	□
7.5	Ensure to Implement Access Control and Authentication	□	□
7.6	Ensure Audit Logging is Enabled	□	□
7.8	Ensure to Implement Monitoring and Alerting	□	□
7.10	Ensure to Configure Backup Window	□	□
8.1	Ensure Keyspace Security is Configured	□	□
9.4	Ensure Authentication and Access Control is Enabled	□	□
9.5	Ensure Audit Logging is Enabled	□	□
9.7	Ensure Monitoring and Alerting is Enabled	□	□
10.4	Ensure Access Control and Authentication is Enabled	□	□

Internal Only - General

Recommendation		Set Correctly	
		Yes	No
10.5	Ensure Fine-Grained Access Control is Enabled	□	□
10.6	Ensure Audit Logging is Enabled	□	□
10.8	Ensure Monitoring and Alerting is Enabled	□	□
11.1	Ensure to Implement Identity and Access Management (IAM)	□	□
11.5	Ensure to Implement Access Control and Authentication	□	□
11.6	Ensure Monitoring and Logging is Enabled	□	□

Appendix: CIS Controls v7 IG 2 Mapped Recommendations

Recommendation		Set Correctly	
		Yes	No
2.1	Ensure the Use of Security Groups	□	□
2.3	Ensure Data in Transit Encryption is Enforced	□	□
2.4	Ensure IAM Roles and Policies are Created	□	□
2.5	Ensure Database Audit Logging is Enabled	□	□
2.6	Ensure Passwords are Regularly Rotated	□	□
2.7	Ensure Least Privilege Access	□	□
3.1	Ensure to Choose the Appropriate Database Engine	□	□
3.3	Ensure to Create a Virtual Private Cloud (VPC)	□	□
3.4	Ensure to Configure Security Groups	□	□
3.6	Enable Encryption in Transit	□	□
3.7	Ensure to Implement Access Control and Authentication	□	□
4.1	Ensure AWS Identity and Access Management (IAM) is in use	□	□
4.2	Ensure Fine-Grained Access Control is implemented	□	□
4.4	Ensure DynamoDB Encryption in Transit	□	□
4.5	Ensure VPC Endpoints are configured	□	□
4.7	Ensure Monitor and Audit Activity is enabled	□	□
5.1	Ensure Secure Access to ElastiCache	□	□
5.2	Ensure Network Security is Enabled	□	□
5.3	Ensure Encryption at Rest and in Transit is configured	□	□
5.5	Ensure Virtual Private Cloud (VPC) is Enabled	□	□
5.6	Ensure Monitoring and Logging is Enabled	□	□
5.8	Ensure Authentication and Access Control is Enabled	□	□
5.9	Ensure Audit Logging is Enabled	□	□
6.1	Ensure Network Security is Enabled	□	□
6.2	Ensure Data at Rest and in Transit is Encrypted	□	□
6.3	Ensure Authentication and Access Control is Enabled	□	□

Recommendation		Set Correctly	
		Yes	No
6.4	Ensure Audit Logging is Enabled	□	□
6.6	Ensure Monitoring and Alerting is Enabled	□	□
7.1	Ensure Network Architecture Planning	□	□
7.2	Ensure VPC Security is Configured	□	□
7.4	Ensure Encryption in Transit is Enabled	□	□
7.5	Ensure to Implement Access Control and Authentication	□	□
7.6	Ensure Audit Logging is Enabled	□	□
7.8	Ensure to Implement Monitoring and Alerting	□	□
7.10	Ensure to Configure Backup Window	□	□
7.11	Ensure to Conduct Security Assessments	□	□
8.1	Ensure Keyspace Security is Configured	□	□
8.2	Ensure Network Security is Enabled	□	□
8.3	Ensure Data at Rest and in Transit is Encrypted	□	□
9.1	Ensure Network Security is Enabled	□	□
9.3	Ensure Data in Transit is Encrypted	□	□
9.4	Ensure Authentication and Access Control is Enabled	□	□
9.5	Ensure Audit Logging is Enabled	□	□
9.7	Ensure Monitoring and Alerting is Enabled	□	□
10.3	Ensure Encryption in Transit is Configured	□	□
10.4	Ensure Access Control and Authentication is Enabled	□	□
10.5	Ensure Fine-Grained Access Control is Enabled	□	□
10.6	Ensure Audit Logging is Enabled	□	□
10.8	Ensure Monitoring and Alerting is Enabled	□	□
11.1	Ensure to Implement Identity and Access Management (IAM)	□	□
11.2	Ensure Network Access is Secure	□	□
11.4	Ensure Data in Transit is Encrypted	□	□
11.5	Ensure to Implement Access Control and Authentication	□	□
11.6	Ensure Monitoring and Logging is Enabled	□	□

Appendix: CIS Controls v7 IG 3 Mapped Recommendations

Recommendation		Set Correctly	
		Yes	No
2.1	Ensure the Use of Security Groups	□	□
2.2	Ensure Data at Rest is Encrypted	□	□
2.3	Ensure Data in Transit Encryption is Enforced	□	□
2.4	Ensure IAM Roles and Policies are Created	□	□
2.5	Ensure Database Audit Logging is Enabled	□	□
2.6	Ensure Passwords are Regularly Rotated	□	□
2.7	Ensure Least Privilege Access	□	□
3.1	Ensure to Choose the Appropriate Database Engine	□	□
3.3	Ensure to Create a Virtual Private Cloud (VPC)	□	□
3.4	Ensure to Configure Security Groups	□	□
3.5	Enable Encryption at Rest	□	□
3.6	Enable Encryption in Transit	□	□
3.7	Ensure to Implement Access Control and Authentication	□	□
4.1	Ensure AWS Identity and Access Management (IAM) is in use	□	□
4.2	Ensure Fine-Grained Access Control is implemented	□	□
4.3	Ensure DynamoDB Encryption at Rest	□	□
4.4	Ensure DynamoDB Encryption in Transit	□	□
4.5	Ensure VPC Endpoints are configured	□	□
4.7	Ensure Monitor and Audit Activity is enabled	□	□
5.1	Ensure Secure Access to ElastiCache	□	□
5.2	Ensure Network Security is Enabled	□	□
5.3	Ensure Encryption at Rest and in Transit is configured	□	□
5.5	Ensure Virtual Private Cloud (VPC) is Enabled	□	□
5.6	Ensure Monitoring and Logging is Enabled	□	□
5.8	Ensure Authentication and Access Control is Enabled	□	□
5.9	Ensure Audit Logging is Enabled	□	□

Internal Only - General

Recommendation		Set Correctly	
		Yes	No
6.1	Ensure Network Security is Enabled	□	□
6.2	Ensure Data at Rest and in Transit is Encrypted	□	□
6.3	Ensure Authentication and Access Control is Enabled	□	□
6.4	Ensure Audit Logging is Enabled	□	□
6.6	Ensure Monitoring and Alerting is Enabled	□	□
7.1	Ensure Network Architecture Planning	□	□
7.2	Ensure VPC Security is Configured	□	□
7.3	Ensure Encryption at Rest is Enabled	□	□
7.4	Ensure Encryption in Transit is Enabled	□	□
7.5	Ensure to Implement Access Control and Authentication	□	□
7.6	Ensure Audit Logging is Enabled	□	□
7.8	Ensure to Implement Monitoring and Alerting	□	□
7.10	Ensure to Configure Backup Window	□	□
7.11	Ensure to Conduct Security Assessments	□	□
8.1	Ensure Keyspace Security is Configured	□	□
8.2	Ensure Network Security is Enabled	□	□
8.3	Ensure Data at Rest and in Transit is Encrypted	□	□
9.1	Ensure Network Security is Enabled	□	□
9.2	Ensure Data at Rest is Encrypted	□	□
9.3	Ensure Data in Transit is Encrypted	□	□
9.4	Ensure Authentication and Access Control is Enabled	□	□
9.5	Ensure Audit Logging is Enabled	□	□
9.7	Ensure Monitoring and Alerting is Enabled	□	□
10.2	Ensure Data at Rest is Encrypted	□	□
10.3	Ensure Encryption in Transit is Configured	□	□
10.4	Ensure Access Control and Authentication is Enabled	□	□
10.5	Ensure Fine-Grained Access Control is Enabled	□	□
10.6	Ensure Audit Logging is Enabled	□	□
10.8	Ensure Monitoring and Alerting is Enabled	□	□
11.1	Ensure to Implement Identity and Access Management (IAM)	□	□

Internal Only - General

Recommendation		Set Correctly	
		Yes	No
11.2	Ensure Network Access is Secure	□	□
11.3	Ensure Data at Rest is Encrypted	□	□
11.4	Ensure Data in Transit is Encrypted	□	□
11.5	Ensure to Implement Access Control and Authentication	□	□
11.6	Ensure Monitoring and Logging is Enabled	□	□

Appendix: CIS Controls v7 Unmapped Recommendations

Recommendation		Set Correctly	
		Yes	No
2.9	Ensure Database is not Publicly accessible	□	□
2.10	Ensure Database has IAM Auth is Enabled	□	□
2.11	Ensure Database has delete protection enabled	□	□
3.12	Ensure Database is not Publicly accessible	□	□
3.13	Ensure Database has IAM Auth is Enabled	□	□
3.14	Ensure Database has delete protection enabled	□	□
4.8	Ensure Database has delete protection enabled	□	□
4.9	Ensure Database has Backup enabled	□	□
5.11	Ensure ElastiCache has Cluster Mode Enabled	□	□
5.12	Ensure ElastiCache is deployed across multiple Availability Zones (AZs)	□	□
5.13	Ensure ElastiCache has automatic backups enabled	□	□
6.7	Ensure MemoryDB has automatic backups enabled	□	□
7.12	Ensure DocumentDB has delete protection enabled	□	□
8.4	Ensure Amazon Keyspaces tables have Point-in-Time Recovery (PITR) enabled	□	□
9.8	Ensure Neptune Database is not Publicly accessible	□	□
9.9	Ensure Neptune Database has automatic backups enabled	□	□
9.10	Ensure Database has delete protection enabled	□	□
9.11	Ensure Neptune DB instances are deployed across multiple Availability Zones (AZs)	□	□
10.10	Ensure Database has automated Backups enabled	□	□

Appendix: CIS Controls v8 IG 1 Mapped Recommendations

Recommendation		Set Correctly	
		Yes	No
2.1	Ensure the Use of Security Groups	□	□
2.4	Ensure IAM Roles and Policies are Created	□	□
2.5	Ensure Database Audit Logging is Enabled	□	□
2.6	Ensure Passwords are Regularly Rotated	□	□
2.7	Ensure Least Privilege Access	□	□
3.1	Ensure to Choose the Appropriate Database Engine	□	□
3.4	Ensure to Configure Security Groups	□	□
3.7	Ensure to Implement Access Control and Authentication	□	□
4.1	Ensure AWS Identity and Access Management (IAM) is in use	□	□
4.2	Ensure Fine-Grained Access Control is implemented	□	□
4.7	Ensure Monitor and Audit Activity is enabled	□	□
4.9	Ensure Database has Backup enabled	□	□
5.1	Ensure Secure Access to ElastiCache	□	□
5.6	Ensure Monitoring and Logging is Enabled	□	□
5.8	Ensure Authentication and Access Control is Enabled	□	□
5.9	Ensure Audit Logging is Enabled	□	□
6.3	Ensure Authentication and Access Control is Enabled	□	□
6.4	Ensure Audit Logging is Enabled	□	□
6.6	Ensure Monitoring and Alerting is Enabled	□	□
7.5	Ensure to Implement Access Control and Authentication	□	□
7.6	Ensure Audit Logging is Enabled	□	□
7.8	Ensure to Implement Monitoring and Alerting	□	□
7.10	Ensure to Configure Backup Window	□	□
8.1	Ensure Keyspace Security is Configured	□	□
9.4	Ensure Authentication and Access Control is Enabled	□	□
9.5	Ensure Audit Logging is Enabled	□	□

Internal Only - General

Recommendation		Set Correctly	
		Yes	No
9.7	Ensure Monitoring and Alerting is Enabled	□	□
10.4	Ensure Access Control and Authentication is Enabled	□	□
10.5	Ensure Fine-Grained Access Control is Enabled	□	□
10.6	Ensure Audit Logging is Enabled	□	□
10.8	Ensure Monitoring and Alerting is Enabled	□	□
11.1	Ensure to Implement Identity and Access Management (IAM)	□	□
11.5	Ensure to Implement Access Control and Authentication	□	□
11.6	Ensure Monitoring and Logging is Enabled	□	□

Appendix: CIS Controls v8 IG 2 Mapped Recommendations

Recommendation		Set Correctly	
		Yes	No
2.1	Ensure the Use of Security Groups	□	□
2.2	Ensure Data at Rest is Encrypted	□	□
2.3	Ensure Data in Transit Encryption is Enforced	□	□
2.4	Ensure IAM Roles and Policies are Created	□	□
2.5	Ensure Database Audit Logging is Enabled	□	□
2.6	Ensure Passwords are Regularly Rotated	□	□
2.7	Ensure Least Privilege Access	□	□
2.9	Ensure Database is not Publicly accessible	□	□
2.10	Ensure Database has IAM Auth is Enabled	□	□
3.1	Ensure to Choose the Appropriate Database Engine	□	□
3.3	Ensure to Create a Virtual Private Cloud (VPC)	□	□
3.4	Ensure to Configure Security Groups	□	□
3.5	Enable Encryption at Rest	□	□
3.6	Enable Encryption in Transit	□	□
3.7	Ensure to Implement Access Control and Authentication	□	□
3.12	Ensure Database is not Publicly accessible	□	□
3.13	Ensure Database has IAM Auth is Enabled	□	□
4.1	Ensure AWS Identity and Access Management (IAM) is in use	□	□
4.2	Ensure Fine-Grained Access Control is implemented	□	□
4.3	Ensure DynamoDB Encryption at Rest	□	□
4.4	Ensure DynamoDB Encryption in Transit	□	□
4.5	Ensure VPC Endpoints are configured	□	□
4.7	Ensure Monitor and Audit Activity is enabled	□	□
4.9	Ensure Database has Backup enabled	□	□
5.1	Ensure Secure Access to ElastiCache	□	□
5.2	Ensure Network Security is Enabled	□	□

Recommendation		Set Correctly	
		Yes	No
5.3	Ensure Encryption at Rest and in Transit is configured	□	□
5.5	Ensure Virtual Private Cloud (VPC) is Enabled	□	□
5.6	Ensure Monitoring and Logging is Enabled	□	□
5.8	Ensure Authentication and Access Control is Enabled	□	□
5.9	Ensure Audit Logging is Enabled	□	□
6.1	Ensure Network Security is Enabled	□	□
6.2	Ensure Data at Rest and in Transit is Encrypted	□	□
6.3	Ensure Authentication and Access Control is Enabled	□	□
6.4	Ensure Audit Logging is Enabled	□	□
6.6	Ensure Monitoring and Alerting is Enabled	□	□
7.1	Ensure Network Architecture Planning	□	□
7.2	Ensure VPC Security is Configured	□	□
7.3	Ensure Encryption at Rest is Enabled	□	□
7.4	Ensure Encryption in Transit is Enabled	□	□
7.5	Ensure to Implement Access Control and Authentication	□	□
7.6	Ensure Audit Logging is Enabled	□	□
7.8	Ensure to Implement Monitoring and Alerting	□	□
7.10	Ensure to Configure Backup Window	□	□
7.11	Ensure to Conduct Security Assessments	□	□
8.1	Ensure Keyspace Security is Configured	□	□
8.2	Ensure Network Security is Enabled	□	□
8.3	Ensure Data at Rest and in Transit is Encrypted	□	□
9.1	Ensure Network Security is Enabled	□	□
9.2	Ensure Data at Rest is Encrypted	□	□
9.3	Ensure Data in Transit is Encrypted	□	□
9.4	Ensure Authentication and Access Control is Enabled	□	□
9.5	Ensure Audit Logging is Enabled	□	□
9.7	Ensure Monitoring and Alerting is Enabled	□	□
9.8	Ensure Neptune Database is not Publicly accessible	□	□
10.2	Ensure Data at Rest is Encrypted	□	□
10.3	Ensure Encryption in Transit is Configured	□	□

Recommendation		Set Correctly	
		Yes	No
10.4	Ensure Access Control and Authentication is Enabled	□	□
10.5	Ensure Fine-Grained Access Control is Enabled	□	□
10.6	Ensure Audit Logging is Enabled	□	□
10.8	Ensure Monitoring and Alerting is Enabled	□	□
11.1	Ensure to Implement Identity and Access Management (IAM)	□	□
11.2	Ensure Network Access is Secure	□	□
11.3	Ensure Data at Rest is Encrypted	□	□
11.4	Ensure Data in Transit is Encrypted	□	□
11.5	Ensure to Implement Access Control and Authentication	□	□
11.6	Ensure Monitoring and Logging is Enabled	□	□

Appendix: CIS Controls v8 IG 3 Mapped Recommendations

Recommendation		Set Correctly	
		Yes	No
2.1	Ensure the Use of Security Groups	□	□
2.2	Ensure Data at Rest is Encrypted	□	□
2.3	Ensure Data in Transit Encryption is Enforced	□	□
2.4	Ensure IAM Roles and Policies are Created	□	□
2.5	Ensure Database Audit Logging is Enabled	□	□
2.6	Ensure Passwords are Regularly Rotated	□	□
2.7	Ensure Least Privilege Access	□	□
2.9	Ensure Database is not Publicly accessible	□	□
2.10	Ensure Database has IAM Auth is Enabled	□	□
3.1	Ensure to Choose the Appropriate Database Engine	□	□
3.3	Ensure to Create a Virtual Private Cloud (VPC)	□	□
3.4	Ensure to Configure Security Groups	□	□
3.5	Enable Encryption at Rest	□	□
3.6	Enable Encryption in Transit	□	□
3.7	Ensure to Implement Access Control and Authentication	□	□
3.12	Ensure Database is not Publicly accessible	□	□
3.13	Ensure Database has IAM Auth is Enabled	□	□
4.1	Ensure AWS Identity and Access Management (IAM) is in use	□	□
4.2	Ensure Fine-Grained Access Control is implemented	□	□
4.3	Ensure DynamoDB Encryption at Rest	□	□
4.4	Ensure DynamoDB Encryption in Transit	□	□
4.5	Ensure VPC Endpoints are configured	□	□
4.7	Ensure Monitor and Audit Activity is enabled	□	□
4.9	Ensure Database has Backup enabled	□	□
5.1	Ensure Secure Access to ElastiCache	□	□
5.2	Ensure Network Security is Enabled	□	□

Internal Only - General

Recommendation		Set Correctly	
		Yes	No
5.3	Ensure Encryption at Rest and in Transit is configured	□	□
5.5	Ensure Virtual Private Cloud (VPC) is Enabled	□	□
5.6	Ensure Monitoring and Logging is Enabled	□	□
5.8	Ensure Authentication and Access Control is Enabled	□	□
5.9	Ensure Audit Logging is Enabled	□	□
6.1	Ensure Network Security is Enabled	□	□
6.2	Ensure Data at Rest and in Transit is Encrypted	□	□
6.3	Ensure Authentication and Access Control is Enabled	□	□
6.4	Ensure Audit Logging is Enabled	□	□
6.6	Ensure Monitoring and Alerting is Enabled	□	□
7.1	Ensure Network Architecture Planning	□	□
7.2	Ensure VPC Security is Configured	□	□
7.3	Ensure Encryption at Rest is Enabled	□	□
7.4	Ensure Encryption in Transit is Enabled	□	□
7.5	Ensure to Implement Access Control and Authentication	□	□
7.6	Ensure Audit Logging is Enabled	□	□
7.8	Ensure to Implement Monitoring and Alerting	□	□
7.10	Ensure to Configure Backup Window	□	□
7.11	Ensure to Conduct Security Assessments	□	□
8.1	Ensure Keyspace Security is Configured	□	□
8.2	Ensure Network Security is Enabled	□	□
8.3	Ensure Data at Rest and in Transit is Encrypted	□	□
9.1	Ensure Network Security is Enabled	□	□
9.2	Ensure Data at Rest is Encrypted	□	□
9.3	Ensure Data in Transit is Encrypted	□	□
9.4	Ensure Authentication and Access Control is Enabled	□	□
9.5	Ensure Audit Logging is Enabled	□	□
9.7	Ensure Monitoring and Alerting is Enabled	□	□
9.8	Ensure Neptune Database is not Publicly accessible	□	□
10.2	Ensure Data at Rest is Encrypted	□	□
10.3	Ensure Encryption in Transit is Configured	□	□

Recommendation		Set Correctly	
		Yes	No
10.4	Ensure Access Control and Authentication is Enabled	□	□
10.5	Ensure Fine-Grained Access Control is Enabled	□	□
10.6	Ensure Audit Logging is Enabled	□	□
10.8	Ensure Monitoring and Alerting is Enabled	□	□
11.1	Ensure to Implement Identity and Access Management (IAM)	□	□
11.2	Ensure Network Access is Secure	□	□
11.3	Ensure Data at Rest is Encrypted	□	□
11.4	Ensure Data in Transit is Encrypted	□	□
11.5	Ensure to Implement Access Control and Authentication	□	□
11.6	Ensure Monitoring and Logging is Enabled	□	□

Internal Only - General

Appendix: CIS Controls v8 Unmapped Recommendations

Recommendation		Set Correctly	
		Yes	No
	No unmapped recommendations to CIS Controls v8	□	□

Appendix: Change History

Date	Version	Changes for this version
Dec 12, 2025	2.0.0	UPDATE - Proposal for Database Public access
Dec 12, 2025	2.0.0	UPDATE - Proposal for Delete Protection
Dec 12, 2025	2.0.0	UPDATE - Proposal for IAM Auth
Dec 12, 2025	2.0.0	UPDATE - Proposal for Encryption at rest for Aurora
Dec 12, 2025	2.0.0	UPDATE - Proposed changes to "enforce" EIT at DB level

Internal Only - General