

CIS SingleStore 8 Benchmark

v1.0.0 - 05-14-2025

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.

Table of Contents

| | |
|---|-----------|
| Terms of Use | 1 |
| Table of Contents | 2 |
| Overview | 5 |
| Important Usage Information | 5 |
| Key Stakeholders | 5 |
| Apply the Correct Version of a Benchmark | 6 |
| Exceptions | 6 |
| Remediation | 7 |
| Summary | 7 |
| Target Technology Details | 8 |
| Intended Audience..... | 8 |
| Consensus Guidance | 9 |
| Typographical Conventions..... | 10 |
| Recommendation Definitions..... | 11 |
| Title..... | 11 |
| Assessment Status..... | 11 |
| Automated | 11 |
| Manual..... | 11 |
| Profile | 11 |
| Description..... | 11 |
| Rationale Statement | 11 |
| Impact Statement..... | 12 |
| Audit Procedure..... | 12 |
| Remediation Procedure..... | 12 |
| Default Value..... | 12 |
| References | 12 |
| CIS Critical Security Controls® (CIS Controls®)..... | 12 |
| Additional Information..... | 12 |
| Profile Definitions | 13 |
| Acknowledgements | 14 |
| Recommendations | 15 |
| 1 Installation and Patching | 15 |
| 1.1 Perform a binary signature check on the SingleStore packages (Manual) | 16 |
| 1.2 Ensure the appropriate SingleStore software version/patches are installed (Manual)..... | 20 |
| 2 Authentication..... | 22 |
| 2.1 Configure secure root password (Automated) | 23 |

| | |
|---|------------|
| 2.2 Configure user authentication (Automated) | 26 |
| 2.3 Generate TLS Certificates for SingleStore node authentication (Automated) | 29 |
| 2.4 Configure secure client and intra-cluster connections (Automated) (Automated) | 31 |
| 2.5 Enable Wire Encryption and Kerberos on HDFS Pipelines (Manual) | 34 |
| 3 Authorization | 36 |
| 3.1 Ensure least privilege for database accounts (Manual) | 37 |
| 3.2 Ensure that role-based access control is enabled and configured appropriately (Manual) | 39 |
| 3.3 Ensure that SingleStore is run using a non-privileged, dedicated service account (Manual) | 41 |
| 3.4 Ensure that each role for each SingleStore database is needed and grants only the necessary privileges (Manual) | 43 |
| 3.5 Review Superuser/Admin Roles (Manual) | 45 |
| 3.6 Delete Unnecessary Default Users (Manual) | 47 |
| 3.7 Synchronize Permissions Across Your Cluster (Manual) | 49 |
| 3.8 Implement Row-Level Security (RLS) (Manual) | 51 |
| 4 Data Encryption | 53 |
| 4.1 Ensure legacy TLS protocols are disabled (Automated) | 54 |
| 4.2 Ensure Encryption of Data in Transit TLS (Transport Encryption) (Automated) | 56 |
| 4.3 Ensure Federal Information Processing Standard (FIPS) is enabled (Automated) | 58 |
| 4.4 Ensure Encryption of Data at Rest (Manual) | 62 |
| 4.5 Configure Encryption Mode (Automated) | 64 |
| 5 Audit Logging | 66 |
| 5.1 Ensure that system activity is audited (Automated) | 67 |
| 5.2 Ensure that audit filters are configured properly (Manual) | 72 |
| 5.3 Ensure that log entries are preserved (Automated) | 74 |
| 6 Operating System Hardening | 76 |
| 6.1 Ensure that SingleStore uses a non-default port (Automated) | 77 |
| 6.2 Ensure that operating system resource limits are set for SingleStore (Manual) | 79 |
| 6.3 Configure Host-Based Security (Manual) | 82 |
| 6.4 Disable the Data API (Automated) | 84 |
| 6.5 Disable Code Engine (Automated) | 86 |
| 6.6 Configure Dedicated Admin Connections (Automated) | 88 |
| 6.7 Configure hiding license-related variables (Automated) | 90 |
| 7 File Permissions | 92 |
| 7.1 Ensure appropriate key file permissions are set (Manual) | 93 |
| 7.2 Ensure appropriate data dir permissions are set. (Manual) | 95 |
| 7.3 Ensure appropriate FILE READ and FILE WRITE privileges for users. (Manual) | 97 |
| Appendix: Summary Table | 99 |
| Appendix: CIS Controls v7 IG 1 Mapped Recommendations | 102 |
| Appendix: CIS Controls v7 IG 2 Mapped Recommendations | 103 |
| Appendix: CIS Controls v7 IG 3 Mapped Recommendations | 105 |
| Appendix: CIS Controls v7 Unmapped Recommendations | 107 |
| Appendix: CIS Controls v8 IG 1 Mapped Recommendations | 108 |
| Appendix: CIS Controls v8 IG 2 Mapped Recommendations | 109 |
| Appendix: CIS Controls v8 IG 3 Mapped Recommendations | 111 |
| Appendix: CIS Controls v8 Unmapped Recommendations | 113 |

Appendix: Change History 114

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.

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.

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 completed 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.

Target Technology Details

This document, CIS SingleStore 8 Benchmark, provides prescriptive guidance for establishing a secure configuration posture for SingleStore version/s 8.x. This guide was tested against SingleStore 8.9.7 running on Ubuntu Linux but applies to other distributions as well.

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 to us at feedback@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 that incorporate SingleStore.

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 |
|---|---|
| <code>Stylized Monospace font</code> | Used for blocks of code, command, and script examples. Text should be interpreted exactly as presented. |
| <code>Monospace font</code> | Used for inline code, commands, UI/Menu selections or examples. Text should be interpreted exactly as presented. |
| < <code>Monospace font in brackets</code> > | Text set in angle brackets denote a variable requiring substitution for a real value. |
| <i>Italic font</i> | 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.

Profile Definitions

The following configuration profiles are defined by this Benchmark:

- **Level 1**

Items in this profile apply to SingleStore and intend to:

- be practical and prudent;
- provide a clear security benefit; and
- not inhibit the utility of the technology beyond acceptable means.

- **Level 2**

This profile extends the Level 1 profile. Items in this profile apply to SingleStore and exhibit one or more of the following characteristics:

- are intended for environments or use cases where security is paramount
- acts as defense in depth measure
- may negatively inhibit the utility or performance of the technology.

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:

Authors

Steven Hart

Editor

Phil White, Center for Internet Security

Contributors

Francisco Godinho

Volodymyr Tkachuk

Tigran Avanesov

Eric Li

Recommendations

1 Installation and Patching

This section provides guidance on ensuring that the SingleStore software is up to date to eliminate easily avoidable vulnerabilities.

1.1 Perform a binary signature check on the SingleStore packages (Manual)

Profile Applicability:

- Level 1

Description:

The first step in deploying your cluster is to download and install the SingleStore Tools on one of the hosts in your cluster. This host will be designated as the main deployment host for deploying SingleStore across your other hosts and setting up your cluster.

These tools perform all major cluster operations including downloading the latest version of SingleStore onto your hosts, assigning and configuring nodes in your cluster, and other management operations.

Rationale:

To ensure the integrity of your SingleStore packages you should perform a binary signature check. This prevents the risk of a supply chain compromise and potentially running malware on your host.

Audit:

You must install cosign before running these steps. See the referenced link for installation instructions.

You can use these steps to verify the authenticity of singlestoredb-server, singlestoredb-toolbox, singlestoredb-studio, and singlestore-client files. This feature will be available in SingleStore Toolbox at a later date.

1. Run the curl command to obtain the list of fields required to verify a file's authenticity.

```

curl https://release.memsql.com/production/index/singlestoredbserver/8.9.json
{
  "releaseID": "2ab21226-4bfe-4bad-9064-550672f13dfe",
  "version": "8.9.7",
  "commit": "106641aldd57894096168ae88bc05e6b0c68f8f2",
  "packages": {
    "memsql-symbols-tar": {
      "Path": "production/tar/x86_64/memsql-symbols-8.9.7-
106641aldd.x86_64.tar.gz",
      "Sha256Sum":
"a4d1fdfa1fd34829e0e3d42fa159ab53d9baffde97d2d8711ed3c68c14b22c68"
    },
    "singlestoredb-server-deb": {
      "Path": "production/debian/pool/singlestoredb-
server8.9.7_106641aldd_amd64.deb",
      "Sha256Sum":
"16843bf29a0b53bd3d79cab2ad3cb836ded8aea976c79e15487a9cb9e7845b5a",
      "Signature": "production/release/signatures/singlestoredb-
server8.9.7_106641aldd_amd64.deb.sigstore.json"
    },
    "singlestoredb-server-rpm": {
      "Path": "production/rpm/x86_64/singlestoredb-server8.9.7-
106641aldd.x86_64.rpm",
      "Sha256Sum":
"656251393fb4de07897749166cdc3c525dc3ab44b695dd76a63d6c944e60434f",
      "Signature": "production/release/signatures/singlestoredb-server8.9.7-
106641aldd.x86_64.rpm.sigstore.json"
    },
    "singlestoredb-server-tar": {
      "Path": "production/tar/x86_64/singlestoredb-server-8.9.7-
106641aldd.x86_64.tar.gz",
      "Sha256Sum":
"a528b9dd9f22110011e5ca7597d87c7d45216366d0fe0cf1307225baf74cd627",
      "Signature": "production/release/signatures/singlestoredb-server-8.9.7-
106641aldd.x86_64.tar.gz.sigstore.json"
    }
  },
  "dockerImages": {
    "nodealma": [
      "singlestore/node:alma-8.9.7-106641aldd",
      "memsql/node:alma-8.9.7-106641aldd",
      "gcr.io/singlestore-public/memsql/node:alma-8.9.7-106641aldd",
      "gcr.io/singlestore-public/mirror/docker.io/memsql/node:alma-8.9.7-
106641aldd"
    ]
  },
  "attestation": {
    "oidcProvider": "https://oidc.eks.us-east-
1.amazonaws.com/id/CCDCDBA1379A5596AB5B2E46DCA385BC",
    "subject": "https://kubernetes.io/namespaces/freya-
production/serviceaccounts/job-worker"
  }
}

```

2. Download the signature file from the release server. This file can be downloaded either via a browser or via curl.
 - Browser: Copy and paste the following line in the address bar on your browser and save the file, or click the "Download signature" button next to the package.

```
https://release.memsql.com/production/release/signatures/singlestoredb-server8.9.7_106641aldd_amd64.deb.sigstore.json
```

- Command line: Run the following curl command package:

```
curl -O  
https://release.memsql.com/production/release/signatures/singlestoredb-server8.9.7_106641aldd_amd64.deb.sigstore.json
```

3. Once the signature file is downloaded, run the following command to verify the authenticity of the **singlestoredb-server8.9.7_106641aldd_amd64.deb** file:

```
echo -n 16843bf29a0b53bd3d79cab2ad3cb836ded8aea976c79e15487a9cb9e7845b5a |  
cosign verify-blob --certificate-oidc-issuer https://oidc.eks.us-east-  
1.amazonaws.com/id/CCDCDBA1379A5596AB5B2E46DCA385BC \  
--certificate-identity https://kubernetes.io/namespaces/freya-  
staging/serviceaccounts/job-worker \  
--bundle singlestoredb-server8.9.7_106641aldd_amd64.deb.sigstore.json \  
--new-bundle-format -
```

Verified OK

Remediation:

Download the SingleStore packages from the linked SingleStore reference page onto a device with access to the main deployment host. Then verify the authenticity of the packages again following the steps above.







Default Value:

Signature verification must be performed by the user.

References:

1. <https://beta.docs.singlestore.com/db/v8.9/deploy/linux/ui-offline-deb/#offline-installation-debian-distribution>
2. https://docs.sigstore.dev/cosign/system_config/installation/

CIS Controls:

| Controls Version | Control | IG 1 | IG 2 | IG 3 |
|------------------|--|------|---|---|
| v8 | 2.5 <u>Allowlist Authorized Software</u> Use technical controls, such as application allowlisting, to ensure that only authorized software can execute or be accessed. Reassess bi-annually, or more frequently. | |  |  |
| v8 | 2.6 <u>Allowlist Authorized Libraries</u> Use technical controls to ensure that only authorized software libraries, such as specific .dll, .ocx, .so, etc., files, are allowed to load into a system process. Block unauthorized libraries from loading into a system process. Reassess bi-annually, or more frequently. | |  |  |
| v7 | 2.7 <u>Utilize Application Whitelisting</u> Utilize application whitelisting technology on all assets to ensure that only authorized software executes and all unauthorized software is blocked from executing on assets. | | |  |
| v7 | 2.8 <u>Implement Application Whitelisting of Libraries</u> The organization's application whitelisting software must ensure that only authorized software libraries (such as *.dll, *.ocx, *.so, etc) are allowed to load into a system process. | | |  |

1.2 Ensure the appropriate SingleStore software version/patches are installed (Manual)

Profile Applicability:

- Level 1

Description:

The SingleStore installation version, along with the patch level, should be the most recent that is compatible with the organization's operational needs. Regularly review the latest release notes, maintenance release changelogs

(<https://support.singlestore.com/hc/en-us/sections/360011370892-ANNOUNCEMENTS>), and security bulletins

(<https://www.singlestore.com/security/bulletins/>) to keep up with the latest security updates.

Rationale:

Using the most recent SingleStore software version along with all applicable patches, helps limit the possibilities for vulnerabilities in the software. The installation version and/or patches applied should be selected according to the needs of the organization. At a minimum, the software version should be supported.

Audit:

Run the following command from within the SingleStore shell to determine if the SingleStore software version complies with your organization's operational needs:

```
SELECT @@memsql_version;
```

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```

Remediation:

Upgrade to the latest version of the SingleStore software:

1. Take a backup of your database cluster
2. Upgrade to the latest version of Toolbox
3. Upgrade your database cluster







Default Value:

Patches are not installed by default.

References:

1. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/maintain-your-cluster/upgrade-or-uninstall-singlestore/upgrade-singlestore/>
2. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/maintain-your-cluster/upgrade-or-uninstall-singlestore/upgrade-to-singlestore-8-9/>
3. <https://support.singlestore.com/hc/en-us/sections/360011370892-ANNOUNCEMENTS>
4. <https://docs.singlestore.com/db/v8.9/support/>
5. <https://www.singlestore.com/security/bulletins/>
6. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
7. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

CIS Controls:

| Controls Version | Control | IG 1 | IG 2 | IG 3 |
|------------------|--|---|---|---|
| v8 | 2.3 Address Unauthorized Software Ensure that unauthorized software is either removed from use on enterprise assets or receives a documented exception. Review 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. |  |  |  |

2 Authentication

This section contains recommendations for requiring authentication before allowing access to the SingleStore database.

2.1 Configure secure root password (Automated)

Profile Applicability:

- Level 1

Description:

When SingleStore is installed, the root user is created on each SingleStore node. You are required to set a password when running `sdb-admin create-node`, but this can be set to a blank password for testing purposes. As a best practice, you should set a secure password for the root user.

Rationale:

Failure to set a password for the root user can enable unauthorized access to the SingleStore database.

It's highly recommended that password length and complexity also be in-place.

Audit:

Try to connect to the SingleStore server from the SingleStore client as the root user without specifying the password option (`-p/-password`).

```
singlestore -u root
```

If you are able to connect to the server without specifying the password option a password has not been configured for the root user. Follow the remediation steps to set a secure password.

```
singlestore>
```

If the root user has been configured with a password attempting to connect to the server without the password option will result in the following error message.

```
Access denied for user 'root'@'localhost' (using password: NO)
```

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```


Remediation:

Once your cluster is deployed, you can change the password for the root user with the `sdb-admin change-root-password` command. This command configures the root password for a single SingleStore node. To configure the password on all nodes in a SingleStore cluster, run:

```
sdb-admin change-root-password --all --yes --password <secure_password>
```

Changing the root password is an online operation for both aggregators and leaves and you do not have to perform any additional operations; however, you must ensure your nodes are running before calling the `change-root-password` command.

Warning

Setting the root password via a command-line argument is often not secure and SingleStore recommends safer, alternative methods to configure passwords. While it is convenient to set the root password using a command-line argument, it is often recommended against this practice for security reasons. The command-line argument accepts passwords entered as plain text, which makes them vulnerable to being discovered in the list of processes running on the system during application runtime. The plain text passwords could also be saved and accessed in the history file if your command-line interpreter maintains a history. For example, in Bash, the command history including password inputs is logged at `~/.bash_history`. In addition, the root password supplied as a command-line argument is displayed on-screen and is visible to anyone who is reading the user's screen.

Some of the more secure, alternative methods to set SingleStore root password are as follows.

- Set the root password using the `MEMSQL_PASSWORD` environment variable. This option is best suited for automated applications.
- As of SingleStore Toolbox 1.6.4, Toolbox commands can solicit the root password interactively from users. The passwords entered in an interactive prompt are not displayed on-screen and are effectively secured from anyone reading the user's screen. The interactive prompt is invoked if neither the `--password` flag nor the `MEMSQL_PASSWORD` environment variable has been used to set the root password.
- In zsh, `start subshell`, `unset HISTFILE`, `exit`
- In bash, `set +o history`




Default Value:

The root user password is set when SingleStore is installed, however, the password can be set to blank for testing purposes.

References:

1. <https://docs.singlestore.com/db/v8.9/security/configure-singlestore-user-accounts/secure-the-initial-singlestore-user-accounts/>
2. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-admin-commands/create-node/>
3. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-admin-commands/change-root-password/>
4. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-toolbox/>
5. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
6. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

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. |  |  |  |

2.2 Configure user authentication (Automated)

Profile Applicability:

- Level 1

Description:

Configure user authentication according to your organization's needs and technology. SingleStore supports multiple authentication methods.

- Password
- JWT
- Kerberos
- Connection links
- PAM
- PAM-based authentication using Active Directory
- SAML 2.0

For specific instructions on each type of configuration, please refer to the referenced links to our documentation.

Rationale:

Failure to authenticate clients, users, and servers can enable unauthorized access to the SingleStore database and can prevent tracing actions back to their sources.

It's highly recommended that password length and complexity also be in-place.

Audit:

Try to connect to the SingleStore server from the SingleStore client as a given user without specifying the password option (**-p/-password**).

```
singlestore -u <username>
```

If you are able to connect to the server without specifying the password option a password has not been configured for the user. Follow the remediation steps to configure a secure password or use one of the available authentication options in the references section.

```
singlestore>
```

If the user has been configured with a password, attempting to connect to the server without the password option will result in the following error message.

```
Access denied for user '<username>'@'localhost' (using password: NO)
```

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```

Remediation:

If using local authentication, set a password policy according to your organization's policies. SingleStore recommends following NIST's guidelines on password and passphrase security and allows you to set password expiration, and complexity and restrict reuse.

```
SET GLOBAL password_min_length = 12;
SET GLOBAL password_min_numeric_chars = 1;
SET GLOBAL password_min_special_chars = 1;
SET GLOBAL password_min_lowercase_chars = 1;
SET GLOBAL password_min_uppercase_chars = 1;
SET GLOBAL password_max_consec_sequential_chars = 2;
SET GLOBAL password_max_consec_repeat_chars = 3;
SET GLOBAL password_expiration_seconds = 0;
```

You can set the user's password using the **SET PASSWORD** command.

```
SET PASSWORD FOR '<username>'@'localhost' = PASSWORD('<secure_password>');
```

Alternatively, you can configure one of the available authentication options in the references section.

Once your users and roles have been created or during creation, enable the lockout policy for users and roles by setting **FAILED_LOGIN_ATTEMPTS** and **PASSWORD_LOCK_TIME** according to your organization's policy.

Notes:

- SingleStore recommends setting this policy on roles and not on individual users. You can then have users associated with the role to facilitate policy management;
- If a user is associated with more than one role with different password lock times, the larger **PASSWORD_LOCK_TIME** value is applied.
- If a user and a role the user is tied to have conflicting **FAILED_LOGIN_ATTEMPTS** settings, the lower value is applied.








Default Value:

By default there is no requirement to set a user's password or authentication mechanism.

References:

1. <https://docs.singlestore.com/cloud/reference/sql-reference/security-management-commands/set-password/>
2. <https://docs.singlestore.com/db/v8.9/security/authentication/>
3. <https://docs.singlestore.com/db/v8.9/security/authentication/configuring-a-password-policy/>
4. <https://docs.singlestore.com/db/v8.9/security/authentication/authenticate-via-jwt/>
5. <https://docs.singlestore.com/db/v8.9/security/authentication/kerberos-authentication/>
6. <https://docs.singlestore.com/db/v8.9/security/authentication/configuring-and-using-connection-links/>
7. <https://docs.singlestore.com/db/v8.9/security/authentication/pam-authentication/>
8. <https://docs.singlestore.com/db/v8.9/security/authentication/authenticate-with-pam-using-active-directory/>
9. <https://docs.singlestore.com/db/v8.9/security/authentication/saml-authentication/>
10. <https://docs.singlestore.com/db/v8.9/security/authentication/configuring-a-password-policy/>
11. <https://docs.singlestore.com/db/v8.7/security/configure-singlestore-user-accounts/set-a-failed-login-attempt-lockout-policy/>
12. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
13. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

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. |  |  |  |
| v8 | 6.3 Require MFA for Externally-Exposed Applications Require all externally-exposed enterprise or third-party applications to enforce MFA, where supported. Enforcing MFA through a directory service or SSO provider is a satisfactory implementation of this Safeguard. | |  |  |
| v7 | 16.3 Require Multi-factor Authentication Require multi-factor authentication for all user accounts, on all systems, whether managed onsite or by a third-party provider. | |  |  |

2.3 Generate TLS Certificates for SingleStore node authentication (Automated)

Profile Applicability:

- Level 2

Description:

To enable SingleStore node authentication over TLS you must generate certificates and keys (or use existing ones, but sharing keys across different services is not recommended in general).

Rationale:

Each SingleStore node which will be receiving TLS connections needs a server certificate and key - these can be the same or different for all servers. The server certificate(s) should be signed by a CA certificate.

Audit:

To verify that the server is configured with a TLS certificate check the `memsql.cnf` file for ssl settings.

```
grep ssl memsql.cnf
ssl_ca           = ./certs/ca-cert.pem
ssl_cert         = ./certs/server-cert.pem
ssl_key          = ./certs/server-key.pem
```

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```

Remediation:

Follow the example steps from the referenced documentation for generating certificates and keys.










Default Value:

TLS certificates and keys are not generated or configured by default.

References:

1. <https://docs.singlestore.com/db/v8.9/security/encryption/ssl-secure-connections/generating-ssl-certificates/>
2. <https://docs.singlestore.com/db/v8.9/security/encryption/ssl-secure-connections/configuring-singlestore-for-secure-connections/>
3. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
4. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

CIS Controls:

| Controls Version | Control | IG 1 | IG 2 | IG 3 |
|------------------|--|--|---|---|
| v8 | 3.10 <u>Encrypt Sensitive Data in Transit</u> Encrypt sensitive data in transit. Example implementations can include: Transport Layer Security (TLS) and Open Secure Shell (OpenSSH). | |  |  |
| v8 | 4.1 <u>Establish and Maintain a Secure Configuration Process</u> Establish and maintain a secure configuration process for enterprise assets (end-user devices, including portable and mobile, non-computing/IoT devices, and servers) and software (operating systems and applications). Review and update documentation annually, or when significant enterprise changes occur that could impact this Safeguard. |  |  |  |
| v7 | 14.4 <u>Encrypt All Sensitive Information in Transit</u> Encrypt all sensitive information in transit. | |  |  |
| v7 | 18.11 <u>Use Standard Hardening Configuration Templates for Databases</u> For applications that rely on a database, use standard hardening configuration templates. All systems that are part of critical business processes should also be tested. | |  |  |

2.4 Configure secure client and intra-cluster connections (Automated) (Automated)

Profile Applicability:

- Level 2

Description:

This section describes how to enable secure connections between clients and the SingleStore cluster, as well as between nodes within the SingleStore cluster. This requires configuring the `ssl_cert`, `ssl_key`, and `ssl_ca` settings on all SingleStore nodes.

Rationale:

This configuration secures intra-cluster communication by making each SingleStore node connect to other SingleStore nodes only over secure connections authenticated by a valid server certificate signed by the CA cert specified by the `ssl_ca` setting.

Note that, depending on the client configuration, a client connecting to SingleStore may or may not use a secure connection even when SSL is enabled on the server. See the Server Configuration to Require Secure Client Connections section of the referenced documentation.

Audit:

Check the `memsql.cnf` configuration file for `ssl_cert`, `ssl_key`, and `ssl_ca` settings on all SingleStore nodes.

```
grep ssl memsql.cnf
```

If the `ssl_cert`, `ssl_key`, and `ssl_ca` settings are missing from your `memsql.cnf` configuration files you do not have secure client and intra-cluster connections configured.

Here is an example of the output when these settings are configured:

```
ssl_ca           = ./certs/ca-cert.pem
ssl_cert         = ./certs/server-cert.pem
ssl_key          = ./certs/server-key.pem
```

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```


Remediation:

1. Place `server-cert.pem`, `server-key.pem`, and `ca-cert.pem` files in the `certs` directory on each SingleStore node in the cluster. You can copy the files from the Generating SSL Certificates section of the linked documentation to all nodes.

Note that the `certs` directory and its contents must be owned by the `memsql` user and group (e.g., `chown -R memsql:memsql <directory>` after copying the certificates to directory).

2. Update the SingleStore configuration for all nodes to set the `ssl_cert`, `ssl_key`, and `ssl_ca` settings to the paths to the `server-cert.pem`, `server-key.pem`, and `ca-cert.pem` files, respectively. These can be absolute paths, or relative to the SingleStore installation directory. You can do this by using `sdb-admin update-config`. For example:

```
sdb-admin update-config --all --key ssl_cert --value ./certs/server-cert.pem
sdb-admin update-config --all --key ssl_key --value ./certs/server-key.pem
sdb-admin update-config --all --key ssl_ca --value ./certs/ca-cert.pem
```

3. Alternatively, edit the `memsql.cnf` file on all aggregators to add the certificate paths in the `[server]` section. For example:

```
ssl_ca           = ./certs/ca-cert.pem
ssl_cert         = ./certs/server-cert.pem
ssl_key          = ./certs/server-key.pem
```

4. Restart all nodes.

```
sdb-admin restart-node --all
```

It is also recommended to add `REQUIRE SSL`, as described in the linked documentation, to the `GRANT` statement of all SingleStore accounts used to connect to aggregator and leaf nodes in `ADD AGGREGATOR` and `ADD LEAF` statements (by default, root).

Additionally, ensure `node_replication_ssl_only` is set to `OFF` to ensure `intra-cluster` and `cross-cluster` communication is done through SSL/TLS.




Default Value:

By default secure client and intra-cluster connections are not configured.

References:

1. <https://docs.singlestore.com/db/v8.9/security/encryption/ssl-secure-connections/server-configuration-for-secure-client-and-intra-cluster-connections/>
2. <https://docs.singlestore.com/db/v8.9/security/encryption/ssl-secure-connections/generating-ssl-certificates/>
3. <https://docs.singlestore.com/db/v8.9/security/encryption/ssl-secure-connections/server-configuration-to-require-secure-client-connections/>
4. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
5. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

CIS Controls:

| Controls Version | Control | IG 1 | IG 2 | IG 3 |
|------------------|---|------|---|---|
| v8 | <u>6.6 Establish and Maintain an Inventory of Authentication and Authorization Systems</u> Establish and maintain an inventory of the enterprise's authentication and authorization systems, including those hosted on-site or at a remote service provider. Review and update the inventory, at a minimum, annually, or more frequently. | |  |  |
| v7 | <u>1.8 Utilize Client Certificates to Authenticate Hardware Assets</u> Use client certificates to authenticate hardware assets connecting to the organization's trusted network. | | |  |

2.5 Enable Wire Encryption and Kerberos on HDFS Pipelines (Manual)

Profile Applicability:

- Level 2

Description:

If using SingleStoreDB with HDFS (Hadoop Distributed File System) pipelines, enable advanced HDFS pipeline features that leverage Kerberos authentication and wire encryption.

Rationale:

When you use HDFS pipelines, you extract data from an HDFS file path, optionally transform the data, and load it to a SingleStore table.

With Load Data from HDFS Using a Pipeline, you can encrypt your pipeline's connection to HDFS and you can authenticate your pipeline using Kerberos. SingleStore supports Data Transfer Protocol (DTP), which encrypts your pipeline's connection to HDFS.

Audit:

Check your deployment for HDFS Pipelines.

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```

Remediation:

Follow the example steps from the referenced documentation for configuring wire encryption.










Default Value:

This capability is not configured by default.

References:

1. <https://docs.singlestore.com/db/v8.9/load-data/load-data-with-pipelines/how-to-load-data-using-pipelines/load-data-from-hdfs-using-a-pipeline/>
2. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
3. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

CIS Controls:

| Controls Version | Control | IG 1 | IG 2 | IG 3 |
|------------------|--|---|---|---|
| v8 | 3.10 <u>Encrypt Sensitive Data in Transit</u> Encrypt sensitive data in transit. Example implementations can include: Transport Layer Security (TLS) and Open Secure Shell (OpenSSH). | |  |  |
| v8 | 4.1 <u>Establish and Maintain a Secure Configuration Process</u> Establish and maintain a secure configuration process for enterprise assets (end-user devices, including portable and mobile, non-computing/IoT devices, and servers) and software (operating systems and applications). Review and update documentation annually, or when significant enterprise changes occur that could impact this Safeguard. |  |  |  |
| v7 | 14.4 <u>Encrypt All Sensitive Information in Transit</u> Encrypt all sensitive information in transit. | |  |  |
| v7 | 18.11 <u>Use Standard Hardening Configuration Templates for Databases</u> For applications that rely on a database, use standard hardening configuration templates. All systems that are part of critical business processes should also be tested. | |  |  |

3 Authorization

SingleStore supports granting access to data and commands through a "role-based" approach. SingleStore provides recommended roles containing different levels of access commonly needed in a database system. In addition, you can create custom-roles.

3.1 Ensure least privilege for database accounts (Manual)

Profile Applicability:

- Level 1

Description:

SingleStore provides a set of recommended roles to be used as a starting point for its RBAC functionality.

The Database Administrator role is responsible for creating and removing databases and has the ability to restore backups. It cannot execute backups, nor can it read any of the data within the database.

The Cluster Administrator role has a minimal set of privileges required to run a SingleStore cluster.

Rationale:

Ensuring highly privileged Roles are granted only for database and cluster administrators, and roles are not scoped to "admin" databases will reduce attack surface and follows the least privilege principle.

Audit:

Check for users with database roles "dba_role" and "cluster_role", and within the "dba" and "cluster" groups, using the following commands.

```
SHOW USERS FOR ROLE 'dba_role';  
SHOW USERS FOR GROUP 'dba';  
SHOW USERS FOR ROLE 'cluster_role';  
SHOW USERS FOR GROUP 'cluster';
```

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```

Remediation:

Review users with Database Administrator and Cluster Administrator access. Remove any users who should not have this level of access.







Default Value:

By default no users have Database Administrator or Cluster Administrator access. RBAC for these roles must be configured by the customer.

References:

1. <https://docs.singlestore.com/db/v8.9/security/administration/role-based-access-control-rbac-at-database-level/>
2. <https://docs.singlestore.com/db/v8.9/security/configure-singlestore-user-accounts/inspect-permissions/>
3. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
4. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

CIS Controls:

| Controls Version | Control | IG 1 | IG 2 | IG 3 |
|------------------|--|---|---|---|
| v8 | 5.4 Restrict Administrator Privileges to Dedicated Administrator Accounts Restrict administrator privileges to dedicated administrator accounts on enterprise assets. Conduct general computing activities, such as internet browsing, email, and productivity suite use, from the user's primary, non-privileged account. |  |  |  |
| v7 | 4.3 Ensure the Use of Dedicated Administrative Accounts Ensure that all users with administrative account access use a dedicated or secondary account for elevated activities. This account should only be used for administrative activities and not internet browsing, email, or similar activities. |  |  |  |

3.2 Ensure that role-based access control is enabled and configured appropriately (Manual)

Profile Applicability:

- Level 1

Description:

Role-based access control (RBAC) is a method of regulating access to resources based on the roles of individual users within an enterprise. A user is granted one or more roles that determine the user's access to database resources and operations. Outside of role assignments, the user has no access to the system. SingleStore can use RBAC to govern access to SingleStore systems. The Compliance Officer role must be configured to manage role and schema authorizations, and the Security Officer role must be configured to manage users and groups.

Rationale:

When properly implemented, RBAC enables users to carry out a wide range of authorized tasks by dynamically regulating their actions according to flexible functions. This allows an organization to control employees' access to all database tables through RBAC.

Audit:

Connect to SingleStore with the Compliance Officer and Security Officer privileges and run the following commands to identify users' roles and privileges:

```
SHOW USERS;  
SHOW USERS FOR ROLE 'role';  
SHOW USERS FOR GROUP 'group';  
SHOW GROUPS;  
SHOW GROUPS FOR ROLE 'role';  
SHOW GROUPS FOR USER 'user'@'%';  
SHOW ROLES;  
SHOW ROLES FOR USER 'user'@'%';  
SHOW ROLES FOR GROUP 'group';
```

Verify that the appropriate role or roles have been configured for each user. Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```


Remediation:

1. Establish roles for SingleStore.
2. Assign the appropriate privileges to each role.
3. Establish a group for each role.
4. Assign the appropriate users to each group.
5. Remove any individual privileges assigned to users that are now addressed by the roles.
6. See the reference below for more information.

Default Value:

By default RBAC roles and groups are not configured. RBAC roles and groups must be configured by the customer.

References:

1. <https://docs.singlestore.com/db/v8.9/security/administration/role-based-access-control-rbac-at-database-level/>
2. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
3. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

CIS Controls:

| Controls Version | Control | IG 1 | IG 2 | IG 3 |
|------------------|--|------|------|------|
| v8 | 6.8 Define and Maintain Role-Based Access Control Define and maintain role-based access control, through determining and documenting the access rights necessary for each role within the enterprise to successfully carry out its assigned duties. Perform access control reviews of enterprise assets to validate that all privileges are authorized, on a recurring schedule at a minimum annually, or more frequently. | | | ● |
| 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. | ● | ● | ● |

3.3 Ensure that SingleStore is run using a non-privileged, dedicated service account (Manual)

Profile Applicability:

- Level 1

Description:

The SingleStore service should not be run using a privileged account such as 'root' because this unnecessarily exposes the operating system to high risk.

Rationale:

Using a non-privileged, dedicated service account restricts the database from accessing the critical areas of the operating system which are not required by SingleStore. This will also mitigate the potential for unauthorized access via a compromised, privileged account on the operating system.

Audit:

Run the following command to get listing of all SingleStore instances, the **PID number**, and the **PID owner**.

```
ps -ef | grep -E "memsqld|singlestoredb"
```

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```

Remediation:

1. Create a dedicated user for performing SingleStore database activity.
2. Set the Database data files, the keyfile, and the SSL private key files to only be readable by the SingleStore user.
3. Set the log files to only be writable by the SingleStore user and readable only by root.






Default Value:

Not configured.

References:

1. <https://docs.singlestore.com/db/v8.9/deploy/>
2. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
3. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

CIS Controls:

| Controls Version | Control | IG 1 | IG 2 | IG 3 |
|------------------|--|---|---|---|
| v8 | <u>5.5 Establish and Maintain an Inventory of Service Accounts</u> Establish and maintain an inventory of service accounts. The inventory, at a minimum, must contain department owner, review date, and purpose. Perform service account reviews to validate that all active accounts are authorized, on a recurring schedule at a minimum quarterly, or more frequently. | |  |  |
| v7 | <u>4.3 Ensure the Use of Dedicated Administrative Accounts</u> Ensure that all users with administrative account access use a dedicated or secondary account for elevated activities. This account should only be used for administrative activities and not internet browsing, email, or similar activities. |  |  |  |

3.4 Ensure that each role for each SingleStore database is needed and grants only the necessary privileges (Manual)

Profile Applicability:

- Level 1

Description:

Reviewing all roles periodically and eliminating unneeded roles as well as unneeded privileges from necessary roles helps minimize the privileges that each user has.

Rationale:

Although role-based access control (RBAC) has many advantages for regulating access to resources, over time some roles may no longer be needed, and some roles may grant privileges that are no longer needed.

Audit:

Perform the following command to view all roles on the database on which the command runs, as well as the privileges granted by each role. Ensure that only necessary roles are listed and only the necessary privileges are listed for each role.

```
SHOW ROLES;  
SHOW GRANTS FOR ROLE 'dba_role';  
SHOW GRANTS FOR ROLE 'cluster_role';
```

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```

Remediation:

To revoke a role use the **DROP** command.







To change privileges for a user-defined role use the **GRANT** command.

```
DROP ROLE 'role_name';  
GRANT CLUSTER on *.* to ROLE 'cluster_role';
```

References:

1. <https://docs.singlestore.com/db/v8.9/security/administration/role-based-access-control-rbac-at-database-level/>
2. <https://docs.singlestore.com/cloud/reference/sql-reference/security-management-commands/show-grants/>
3. <https://docs.singlestore.com/cloud/reference/sql-reference/security-management-commands/drop-role/>
4. <https://docs.singlestore.com/cloud/reference/sql-reference/security-management-commands/grant/>
5. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
6. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

CIS Controls:

| Controls Version | Control | IG 1 | IG 2 | IG 3 |
|------------------|--|---|---|---|
| v8 | 5.4 Restrict Administrator Privileges to Dedicated Administrator Accounts Restrict administrator privileges to dedicated administrator accounts on enterprise assets. Conduct general computing activities, such as internet browsing, email, and productivity suite use, from the user's primary, non-privileged account. |  |  |  |
| 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 Review Superuser/Admin Roles (Manual)

Profile Applicability:

- Level 2

Description:

Roles provide several advantages that make it easier to manage privileges in a database system. Security administrators can control access to their databases in a way that mirrors the structure of their organizations (they can create roles in the database that map directly to the job functions in their organizations). The assignment of privileges is simplified. Instead of granting the same set of privileges to each individual user in a particular job function, the administrator can grant this set of privileges to a role representing that job function and then grant that role to each user in that job function.

Rationale:

Reviewing the Superuser/Admin roles within a database helps minimize the possibility of privileged unwanted access.

Audit:

Superuser roles provide the ability to assign any user any privilege on any database, which means that users with one of these roles can assign themselves any privilege on any database.

```
SHOW USERS;  
SHOW ROLES;  
SHOW GRANTS FOR 'user'@'%';  
SHOW GRANTS FOR ROLE 'role';
```

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```

Remediation:

Remove a user from a group with a superuser role on the current database.

```
REVOKE GROUP 'group_name' FROM user;
```










Default Value:

None.

References:

1. <https://docs.singlestore.com/db/v8.9/security/administration/role-based-access-control-rbac-at-database-level/>
2. <https://docs.singlestore.com/db/v8.9/reference/singlestore-operator-reference/create-a-superadmin-user/>
3. <https://docs.singlestore.com/db/v8.9/reference/sql-reference/security-management-commands/revoke-group/>
4. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
5. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

CIS Controls:

| Controls Version | Control | IG 1 | IG 2 | IG 3 |
|------------------|--|---|---|---|
| v8 | 5.4 Restrict Administrator Privileges to Dedicated Administrator Accounts Restrict administrator privileges to dedicated administrator accounts on enterprise assets. Conduct general computing activities, such as internet browsing, email, and productivity suite use, from the user's primary, non-privileged account. |  |  |  |
| v7 | 4.3 Ensure the Use of Dedicated Administrative Accounts Ensure that all users with administrative account access use a dedicated or secondary account for elevated activities. This account should only be used for administrative activities and not internet browsing, email, or similar activities. |  |  |  |
| v7 | 16.8 Disable Any Unassociated Accounts Disable any account that cannot be associated with a business process or business owner. |  |  |  |

3.6 Delete Unnecessary Default Users (Manual)

Profile Applicability:

- Level 2

Description:

If upgrading from SingleStoreDB 5.8 or earlier to 6.0 and then gradually to 8.1, 8.5, or 8.7, drop unnecessary default users.

Rationale:

In MemSQL 6.0 and later, the only default user created on each SingleStore node during installation is the `'root'@'%'` user, which should be configured as described in the previous section of this doc.

In MemSQL 5.8 and earlier, several default users are created on each SingleStore node during installation. SingleStore recommends deleting all of these default users except for the `'root'@'%'` user.

Audit:

Check users configured in the database.

```
SHOW USERS;
```

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```

Remediation:

To delete these users, use the DROP USER command. On each SingleStore node (including both aggregators and leaves), log in as the root user or another user with sufficient permissions, and run:

```
DROP USER ''@'localhost';  
DROP USER ''@'127.0.0.1';  
DROP USER 'root'@'localhost';  
DROP USER 'dashboard'@'%;'  
DROP USER 'dashboard'@'localhost';
```





Default Value:

None.

References:

1. <https://docs.singlestore.com/db/v8.9/security/configure-singlestore-user-accounts/secure-the-initial-singlestore-user-accounts/#deleting-unnecessary-default-users>
2. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
3. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

CIS Controls:

| Controls Version | Control | IG 1 | IG 2 | IG 3 |
|------------------|--|---|---|---|
| v8 | 6.8 Define and Maintain Role-Based Access Control Define and maintain role-based access control, through determining and documenting the access rights necessary for each role within the enterprise to successfully carry out its assigned duties. Perform access control reviews of enterprise assets to validate that all privileges are authorized, on a recurring schedule at a minimum annually, or more frequently. | | |  |
| 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. |  |  |  |

3.7 Synchronize Permissions Across Your Cluster (Manual)

Profile Applicability:

- Level 2

Description:

Enable permissions synchronization across the cluster to keep non-root user accounts, roles, and groups in check automatically and to avoid potential divergent configurations that may create security gaps.

Rationale:

You can ease user account management by automatically keeping non-root user accounts, roles, and groups in sync across your cluster. This functionality is enabled through a new engine variable named `sync_permissions`. All user account operations performed on the master aggregator are propagated automatically to each child aggregator in the cluster. For example, if an administrator connects to the master aggregator, issues a `GRANT` to change a user's password, and then tries to connect to a child aggregator as that user using the old password, the old password will be rejected. If this command is issued in the child aggregator, it is first forwarded to the master aggregator, then propagated in the same way.

Audit:

Check the `sync_permissions` variable in the database.

```
SELECT @@sync_permissions;
```

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```

Remediation:

Connect to the master aggregator as root and drop all non-root users, if present. Any groups or roles you have created will be removed after you enable `sync_permissions`. Set the `sync_permissions` variable:

```
SET GLOBAL sync_permissions = ON;
```

Notes:

- This is a permanent change. Once `sync_permissions` is enabled, you cannot turn it off and you can no longer create any local non-root users, groups, or roles; once you set this value, it will be set on all child aggregators in your cluster;
- If there are or you expect the cluster to have > 100 users, setting this flag might result in a performance hit on the system (may not be recommended depending on your setup and context).





Default Value:

`sync_permissions` is set to **ON** by default.

References:

1. <https://docs.singlestore.com/db/v8.9/security/administration/synchronizing-permissions-across-your-cluster/>
2. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
3. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

CIS Controls:

| Controls Version | Control | IG 1 | IG 2 | IG 3 |
|------------------|---|---|---|---|
| v8 | 6.8 <u>Define and Maintain Role-Based Access Control</u> Define and maintain role-based access control, through determining and documenting the access rights necessary for each role within the enterprise to successfully carry out its assigned duties. Perform access control reviews of enterprise assets to validate that all privileges are authorized, on a recurring schedule at a minimum annually, or more frequently. | | |  |
| v7 | 14.6 <u>Protect Information through Access Control Lists</u> 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 Implement Row-Level Security (RLS) (Manual)

Profile Applicability:

- Level 2

Description:

For sensitive workloads or data you would like to see restricted to certain users or roles based on a need-to-know basis, it is recommended to enable RLS (Row-Level Security). It's possible to create tables with RLS enabled or alter existing tables to incorporate RLS.

RLS works for read-only, you cannot restrict write access on a table using this configuration. RLS restricts the results of a query returned for each user based on their permissions but does not guarantee security against side-channel attacks. For example, a malicious user with access to an RLS view may connive another user with a different set of permissions on a table to extract information. A carefully designed query can trigger exceptions on non-permitted data, timing dependence, **EXPLAIN** command options, etc. For example, using a query with "**1/(total_value - 200000)**" on a Tenders table may result in a "**divide by zero**" error, thereby indicating there is at least one tender with this value.

Rationale:

Row-Level Security (RLS) allows only those users who have the required permissions to access data by rows in a database. For example, RLS can be used to restrict each salesman to access only those rows in a table that are relevant to their sales details.

RLS restricts users' access to data at the database level instead of the application level. The database applies this access control to the table whenever a query runs, irrespective of which application needs the data.

Row-level security in SingleStore is achieved by creating a view on a table with a special roles column. RLS works for read-only, you cannot restrict write access on a table using this configuration.

Audit:

For a table to be used with row-level security, it must have a **VARBINARY** column where a row entry in the column contains a comma separated list of roles which have access to that row. There are special formatting constraints for the roles columns which are discussed below.

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```

Remediation:

For a given role, the **ACCESS_ROLES** field will be used to specify which roles have access to that row. The **DATA_1** through **DATA_3** columns are data stored in a table. It is important that each role name in **ACCESS_ROLES** be surrounded by a preceding and trailing comma.

In addition to applying RLS using an access-control list field, you can also restrict access based on the data in the rows of a table. For example, grant access based on an owner id or restrict access to only rows for a specific year, 2022.

To create a new table with a roles column use an appropriate version of the following command:

```
CREATE TABLE <table>(ACCESS_ROLES VARBINARY(<SIZE>) DEFAULT ",", ...);
```

It is important that the default value for **ACCESS_ROLES** be a comma: **", "** for row-level security to work correctly.

The **<SIZE>** of the **ACCESS_ROLES** column should be set to match the expected number of roles.

Default Value:

RLS is not configured by default.

References:

1. <https://docs.singlestore.com/db/v8.9/security/administration/row-level-security-rls-deployment-guide/>
2. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
3. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

CIS Controls:

| Controls Version | Control | IG 1 | IG 2 | IG 3 |
|------------------|--|------|------|------|
| v8 | 6.8 Define and Maintain Role-Based Access Control Define and maintain role-based access control, through determining and documenting the access rights necessary for each role within the enterprise to successfully carry out its assigned duties. Perform access control reviews of enterprise assets to validate that all privileges are authorized, on a recurring schedule at a minimum annually, or more frequently. | | | ● |
| 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 Data Encryption

This section contains recommendations for securing data at rest (stored) and data in motion (transiting) for SingleStore.

4.1 Ensure legacy TLS protocols are disabled (Automated)

Profile Applicability:

- Level 2

Description:

Only modern TLS protocols should be enabled in SingleStore for all client connections and upstream connections. Removing legacy TLS and SSL protocols (SSL 3.0, TLS 1.0 and 1.1), and enabling emerging and stable TLS protocols (TLS 1.2, and TLS 1.3), ensures users are able to take advantage of strong security capabilities and protects them from insecure legacy protocols.

Rationale:

SSL 3.0, TLS 1.0 and 1.1 have known vulnerabilities and should no longer be used.

Audit:

To verify that the server disables legacy TLS protocols check the configured TLS versions.

```
SELECT @@tls_version;
```

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```

Remediation:

The following command updates the TLS version to TLSv1.2 on all nodes in the cluster.

```
sdb-admin update-config --key tls_version --value TLSv1.2 --all
```

The following command restarts all nodes in the cluster for the new `tls_version` value to take effect.

```
sdb-admin stop-node --all  
sdb-admin start-node --all
```

Confirm the new `tls_version`.

```
SELECT @@tls_version;  
+-----+  
| @@tls_version |  
+-----+  
| TLSv1.2      |  
+-----+  
1 row in set (0.01 sec)
```

Additionally, review the list of SSL/TLS ciphers enumerated in `ssl_cipher` - note that you can further restrict the ciphersuites to be used by modifying this variable.










Default Value:

TLSv1, TLSv1.1, TLSv1.2

References:

1. <https://docs.singlestore.com/db/v8.9/security/encryption/ssl-secure-connections/specifying-the-tls-version/>
2. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
3. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

CIS Controls:

| Controls Version | Control | IG 1 | IG 2 | IG 3 |
|------------------|--|---|---|---|
| v8 | 3.10 <u>Encrypt Sensitive Data in Transit</u> Encrypt sensitive data in transit. Example implementations can include: Transport Layer Security (TLS) and Open Secure Shell (OpenSSH). | |  |  |
| v8 | 4.1 <u>Establish and Maintain a Secure Configuration Process</u> Establish and maintain a secure configuration process for enterprise assets (end-user devices, including portable and mobile, non-computing/IoT devices, and servers) and software (operating systems and applications). Review and update documentation annually, or when significant enterprise changes occur that could impact this Safeguard. |  |  |  |
| v7 | 14.4 <u>Encrypt All Sensitive Information in Transit</u> Encrypt all sensitive information in transit. | |  |  |
| v7 | 18.11 <u>Use Standard Hardening Configuration Templates for Databases</u> For applications that rely on a database, use standard hardening configuration templates. All systems that are part of critical business processes should also be tested. | |  |  |

4.2 Ensure Encryption of Data in Transit TLS (Transport Encryption) (Automated)

Profile Applicability:

- Level 1

Description:

Use TLS to protect all incoming and outgoing connections between the SingleStore server and client connections. TLS ensures that SingleStore traffic is only readable by the intended client.

Rationale:

This prevents sniffing of cleartext traffic or performing a man-in-the-middle attack between the SingleStore server and connecting clients.

Audit:

Check that the SingleStore users have the **REQUIRE SSL** clause added to their **GRANT** statement.

```
SHOW USERS;  
SHOW GRANTS FOR 'user'@'%';
```

Example output:

```
+-----+  
| Grants for user@%                               |  
+-----+  
| GRANT ALL PRIVILEGES ON *.* TO 'user'@'%' REQUIRE SSL |  
+-----+
```

Check the **default_user_require_ssl** engine variable. This engine variable controls the default value for **REQUIRE SSL** in the **CREATE USER DDL** statement.

If it is enabled and **REQUIRE** is not specified in the **CREATE USER** statement, **SSL** will be required by default.

Once it is turned on, it impacts only the newly created users. It will not enforce **SSL** for existing users.

This variable can sync to all aggregators.

```
select @@default_user_require_ssl;
```

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```

Remediation:

To make the server restrict access to clients over TLS only, add the **REQUIRE SSL** clause to the user's **GRANT** statement, for example:

```
GRANT all ON *.* TO 'user'@'%' REQUIRE SSL;
```

Set the **default_user_require_ssl** engine variable to set **REQUIRE SSL** for all newly created users.

```
SET GLOBAL default_user_require_ssl = 1;
```





Default Value:

By default the **REQUIRE SSL** clause is not added to the **CREATE USER DDL** statement and the **default_user_require_ssl** engine variable is set to **OFF**.

References:

1. <https://docs.singlestore.com/db/v8.9/security/encryption/ssl-secure-connections/server-configuration-to-require-secure-client-connections/>
2. <https://docs.singlestore.com/cloud/reference/configuration-reference/engine-variables/list-of-engine-variables/>
3. <https://docs.singlestore.com/cloud/reference/configuration-reference/engine-variables/assigning-expressions-to-variables/>
4. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
5. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

CIS Controls:

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

4.3 Ensure Federal Information Processing Standard (FIPS) is enabled (Automated)

Profile Applicability:

- Level 2

Description:

The Federal Information Processing Standards (FIPS) developed by the National Institute of Standards and Technology (NIST) is a set of standards relating to the security of data and other information technology resources. These standards help protect the confidentiality, integrity and authenticity of information. The FIPS protocol establishes security standards to protect against unauthorized access to data via cyberattacks and other threats.

Rationale:

FIPS sets specific requirements for cybersecurity such as computer encryption schemes, key generation methods, computer security, and interoperability among others.

When `ssl_fips_mode` is set to `ON`, the FIPS mode is enabled, which is a security standard that sets a stringent limit on what security algorithms are allowed, and mandates the use of specific key lengths and hash functions. This keeps the node more resistant to external attacks when an SSL/TLS connection is in use.

When `ssl_fips_mode` is set to `ON`, certain cryptographic algorithms and hash functions, such as MD5, are disabled because they do not satisfy the standards of FIPS mode.

Enabling FIPS mode will restrict the TLS version to 1.2 for cluster communications.

SingleStore uses the OpenSSL FIPS module.

Audit:

Check the `ssl_fips_mode` engine variable. This engine variable is used to enable SSL/TLS FIPS mode on each node.

```
SELECT @@ssl_fips_mode;
```

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```

Remediation:

The `ssl_fips_mode` engine variable is only settable at startup. This variable can only be set while the node is offline and any changes made to this variable will take effect on the next start of the node.

You may enable and configure FIPS mode using the following two methods.

Method 1: Use SingleStore Tools (Preferred Method)

1. Update the FIPS mode configuration variables using the `sdb-admin update-config` command. Use the `--all` flag to update the variable settings on all nodes. For example, execute the following commands:

```
sdb-admin update-config --all --key "ssl-fips-mode" --value "ON"
```

2. Restart the nodes

```
sdb-admin restart-node --all
```

3. Ensure that SingleStore starts successfully. Once started, validate that your settings have been loaded successfully by executing the following SQL command in a SQL client.

```
SELECT @@ssl_fips_mode;
+-----+
| @@ssl_fips_mode |
+-----+
|                1 |
+-----+
```

Method 2: Modify the memsql.cnf File

Note: Always ensure that each node in your cluster has been stopped before making FIPS mode configuration changes in the `memsql.cnf` file.

1. Open a new console window with access to the node you want to configure.
2. Stop any SingleStore processes on the node.

```
sdb-admin stop-node --all
```

3. After a node has been stopped, navigate to the `memsql.cnf` path for the node and open the file with a text editor. Add the **FIPS mode** variable.

```
ssl_fips_mode = ON
```

4. When your configuration is complete, save the `memsql.cnf` file and exit the text editor.

Warning: Repeat the configuration update process for each node in your cluster before continuing.

5. Start the node.

```
sdb-admin start-node --all
```

6. Ensure that SingleStore starts successfully. Once started, validate that your settings have been loaded successfully by executing the following SQL command in a SQL client.

```
SELECT @@ssl_fips_mode;
+-----+
| @@ssl_fips_mode |
+-----+
|                1 |
+-----+
```






Default Value:

By default the `ssl_fips_mode` engine variable is set to **OFF**.

References:

1. <https://docs.singlestore.com/db/v8.9/security/encryption/configuring-ssl-tls-fips/>
2. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-deploy-commands/setup-cluster/>
3. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/memsqlctl-commands/create-node/>
4. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
5. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

CIS Controls:

| Controls Version | Control | IG 1 | IG 2 | IG 3 |
|------------------|---|------|---|---|
| v8 | 3.10 <u>Encrypt Sensitive Data in Transit</u> Encrypt sensitive data in transit. Example implementations can include: Transport Layer Security (TLS) and Open Secure Shell (OpenSSH). | |  |  |
| v7 | 14.4 <u>Encrypt All Sensitive Information in Transit</u> Encrypt all sensitive information in transit. | |  |  |
| v7 | 14.8 <u>Encrypt Sensitive Information at Rest</u> 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 Encryption of Data at Rest (Manual)

Profile Applicability:

- Level 2

Description:

Encryption of data at rest must be enabled to ensure compliance with security and privacy standards including HIPAA, PCI-DSS, and FERPA.

Encryption at rest, when used in conjunction with transport encryption and good security policies that protect relevant accounts, passwords, and encryption keys.

Rationale:

Unauthorized users, such as intruders who are attempting security attacks, cannot read the data from storage and back up media unless they have the master encryption key to decrypt it.

Audit:

You can use the `dmsetup status` command to check for LUKS-encrypted partitions.

```
sudo dmsetup status
```

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```

Remediation:

SingleStore is compatible with at-rest disk-based encryption via LUKS (Linux Unified Key Setup). Configure your block device to be encrypted with LUKS and then install SingleStore on the encrypted volume. Specific steps for different versions of Linux are available in our documentation.

Example Setup Process

- Prepare block device
- Encrypt block device with LUKS
- Create filesystem (i.e., `mkfs.ext4 /dev/mapper/myencryptedvolume`)
- Mount filesystem (i.e., `mount /dev/mapper/myencryptedvolume /data`)
- Install SingleStore normally to encrypted location using the SingleStore management tools

SingleStore also has partner integrations available (IBM Guardium Data Encryption and CipherTrust Transparent Encryption) should you prefer an alternative to LUKS encryption.




Default Value:

Encryption of data at rest is not enabled by default.

References:

1. <https://docs.singlestore.com/db/v8.9/security/encryption/#encryption-at-rest>
2. <https://docs.singlestore.com/db/v8.9/security/encryption/securing-data-at-rest-with-ibm-guardium-data-encryption/>
3. <https://docs.singlestore.com/db/v8.9/security/encryption/securing-data-at-rest-with-ciphertrust-transparent-encryption/>
4. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
5. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

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. | | |  |

4.5 Configure Encryption Mode (Automated)

Profile Applicability:

- Level 2

Description:

You can use the `aes_default_encryption_mode` variable to control the supported key size. It takes a value in `aes-keylen-mode` format, where `keylen` is the key length in bits and `mode` is the encryption mode. The value is not case-sensitive. Permitted `keylen` values are `128` and `256`. Permitted `mode` values are `ECB`, `GCM`, and `CBC`.

Rationale:

Configure the encryption mode required by your organization's security policy.

Audit:

Check the `aes_default_encryption_mode` variable.

```
SELECT @@aes_default_encryption_mode;
```

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```

Remediation:

Set the variable to the mode you require.

```
sdb-admin update-config --all --key "aes_default_encryption_mode" --value "aes-128-ecb"
```




Default Value:

Encryption mode `aes-128-ecb` is enabled by default.

References:

1. <https://docs.singlestore.com/db/v8.9/reference/configuration-reference/engine-variables/list-of-engine-variables/>
2. <https://docs.singlestore.com/db/v8.9/reference/sql-reference/string-functions/aes-encrypt/>
3. <https://docs.singlestore.com/db/v8.9/reference/sql-reference/string-functions/aes-decrypt/>
4. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
5. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

CIS Controls:

| Controls Version | Control | IG 1 | IG 2 | IG 3 |
|------------------|---|------|---|---|
| v8 | 3.11 <u>Encrypt Sensitive Data at Rest</u> 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 <u>Encrypt Sensitive Information at Rest</u> 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 Audit Logging

This section contains recommendations related to configuring audit logging in SingleStore.

5.1 Ensure that system activity is audited (Automated)

Profile Applicability:

- Level 1

Description:

SingleStore logs all database activities and writes the generated logs to an external location. SingleStore provides multiple logging levels, and each level provides limited or exhaustive information about user actions and database responses.

Rationale:

This feature is useful for performing common information security tasks such as auditing, investigating suspicious activity, and validating access control policies.

Audit:

Check if the `auditlog_level` has been set in the `memsql.cnf` file.

```
grep auditlog_level memsql.cnf
```

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```

Remediation:

To enable audit logging, set `auditlog_level` to a value other than OFF. For example, the `ALL-RESULTS-INCLUDING-PARSE-FAILS` level is inclusive of `ALL-RESULTS` but also logs invalid statements that fail to parse. See the Audit Logging Levels documentation referenced below for more information about logging levels.

Consider enabling synchronous audit log record persistence by enabling `auditlog_disk_sync` if required by your organization's security needs. By default, this value is set to OFF and audit log disk syncs are delayed.

You may enable and configure audit logging using the following two methods. With each method, you can set the variables that are described in the previous section. Note that you can only set these variables to take effect when a node starts, as opposed to taking effect while a node is running.

Toolbox preserves the node's base directory (or "basedir") during an upgrade. By default, Toolbox sets the `auditlogsdir` relative to the node's base directory, and the value for `auditlogsdir` to `auditlogs`. For the tarball-based deployments, a node's default base directory is `~/memsql/nodes/<hash>`, however, the base directory can be anywhere on the filesystem.

When changing the value of `auditlogsdir`, SingleStore suggests using either:

- A relative path
- An absolute path outside of the memsql directory (i.e., outside of `/var/lib/memsql`).

For example, `/var/log/memsql` or `/var/log/singlestore` are suitable provided that the `memsql:memsql` permissions are also set on this directory.

Set `auditlogsdir` to a trusted and secured local or network directory where audit logs will be written.

Notes:

- User credentials and PII information contained in all valid statements and queries is obfuscated in audit logs. When invalid statements cannot be parsed, the literal query text is included in the log entry. This text may contain sensitive information;
- Always ensure that the log file location is secured appropriately and that extra precaution is taken when processing the logs (particularly when selecting logging levels that include `INCLUDING-PARSE-FAILS`);

Method 1: Use SingleStore Tools (Preferred Method)

1. Update the audit logging configuration variables using the `sdb-admin update-config` command. Use the `--all` flag to update the variable settings on all nodes. For example, execute the following commands:

```
sdb-admin update-config --all --key "auditlog_level" --value "ADMIN-ONLY"
sdb-admin update-config --all --key "auditlog_disk_sync" --value "OFF"
sdb-admin update-config --all --key "auditlog_rotation_size" --value
"134217728"
sdb-admin update-config --all --key "auditlog_rotation_time" --value "3600"
sdb-admin update-config --all --key "auditlogsdir" --value "auditlogs"
```

2. Restart the nodes

```
sdb-admin restart-node --all
```

3. Ensure that SingleStore starts successfully. Once started, validate that your settings have been loaded successfully by executing the following SQL command in a SQL client.

```
SHOW GLOBAL VARIABLES LIKE 'audit%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| auditlog_disk_sync | OFF |
| auditlog_level | ADMIN-ONLY |
| auditlog_retention_period | 0 |
| auditlog_rotation_size | 134217728 |
| auditlog_rotation_time | 3600 |
| auditlogsdir | auditlogs |
+-----+-----+
```

Method 2: Modify the `memsql.cnf` File

Note: Always ensure that each node in your cluster has been stopped before making audit logging configuration changes in the `memsql.cnf` file.

1. Open a new console window with access to the node you want to configure.
2. Stop any SingleStore processes on the node.

```
sdb-admin stop-node --all
```

3. After a node has been stopped, navigate to the `memsql.cnf` path for the node and open the file with a text editor. Add the required audit logging variables.

For example:

```
auditlog_level = ADMIN-ONLY
auditlog_disk_sync = OFF
auditlog_rotation_size = 134217728
auditlog_rotation_time = 3600
auditlogsdir = auditlogs
```

4. When your configuration is complete, save the `memsql.cnf` file and exit the text editor.

Warning: Repeat the configuration update process for each node in your cluster before continuing.

5. Start the node.

```
sdb-admin start-node --all
```

6. Ensure that SingleStore starts successfully. Once started, validate that your settings have been loaded successfully by executing the following SQL command in a SQL client.

```
SHOW GLOBAL VARIABLES LIKE 'audit%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| auditlog_disk_sync | OFF |
| auditlog_level | ADMIN-ONLY |
| auditlog_retention_period | 0 |
| auditlog_rotation_size | 134217728 |
| auditlog_rotation_time | 3600 |
| auditlogsdir | auditlogs |
+-----+-----+
```









Default Value:

By default `auditlog_level` is set to OFF.

References:

1. <https://docs.singlestore.com/db/v8.9/security/audit-logging/>
2. <https://docs.singlestore.com/db/v8.9/security/audit-logging/configuring-audit-logging/>
3. <https://docs.singlestore.com/db/v8.9/security/audit-logging/audit-logging-levels/>
4. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
5. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

CIS Controls:

| Controls Version | Control | IG 1 | IG 2 | IG 3 |
|------------------|---|---|---|---|
| v8 | 8.2 Collect Audit Logs Collect audit logs. Ensure that logging, per the enterprise's audit log management process, has been enabled across enterprise assets. |  |  |  |
| v7 | 6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices. |  |  |  |
| v7 | 6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements. | |  |  |

5.2 Ensure that audit filters are configured properly (Manual)

Profile Applicability:

- Level 2

Description:

SingleStore supports auditing of various operations.

There are 11 logging levels that can be specified by the `auditlog_level` variable in a node's `memsql.cnf` file. These levels can be organized into three categories, each with increasing verbosity:

- Logging is disabled:
 - `OFF`
- Log only valid statements and queries:
 - `LOGINS-ONLY`
 - `ADMIN-ONLY`
 - `WRITES-ONLY`
 - `ALL-QUERIES`
 - `ALL-QUERIES-PLAINTEXT`
 - `ALL-RESULTS`
- Log valid and invalid statements and queries:
 - `ADMIN-ONLY-INCLUDING-PARSE-FAILS`
 - `WRITES-ONLY-INCLUDING-PARSE-FAILS`
 - `ALL-QUERIES-INCLUDING-PARSE-FAILS`
 - `ALL-QUERIES-PLAINTEXT-INCLUDING-PARSE-FAILS`
 - `ALL-RESULTS-INCLUDING-PARSE-FAILS`

Rationale:

All operations carried out on the database can be configured for logging. This helps in backtracking and tracing any incident that occurs.

Audit:

To verify that audit filters are configured on MongoDB as per the organization's requirements, run the following command:

```
SHOW GLOBAL VARIABLES LIKE 'audit%';
```

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```

Remediation:

Set the audit logging level based on the organization's requirements.











Default Value:

By default auditlog_level is set to **OFF**.

References:

1. <https://docs.singlestore.com/db/v8.9/security/audit-logging/>
2. <https://docs.singlestore.com/db/v8.9/security/audit-logging/configuring-audit-logging/>
3. <https://docs.singlestore.com/db/v8.9/security/audit-logging/audit-logging-levels/>
4. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
5. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

CIS Controls:

| Controls Version | Control | IG 1 | IG 2 | IG 3 |
|------------------|--|---|---|---|
| v8 | 8.2 Collect Audit Logs Collect audit logs. Ensure that logging, per the enterprise's audit log management process, has been enabled across enterprise assets. |  |  |  |
| v8 | 8.5 Collect Detailed Audit Logs Configure detailed audit logging for enterprise assets containing sensitive data. Include event source, date, username, timestamp, source addresses, destination addresses, and other useful elements that could assist in a forensic investigation. | |  |  |
| v7 | 6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices. |  |  |  |
| v7 | 6.3 Enable Detailed Logging Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements. | |  |  |

5.3 Ensure that log entries are preserved (Automated)

Profile Applicability:

- Level 2

Description:

SingleStore allows a log retention period to be configured. Configuring a retention period will enable purging of audit log records.

Rationale:

While purging old audit log records may conserve disk space, it will destroy records that may be needed for forensic analysis.

Audit:

Check the `auditlog_retention_period` setting.

```
SHOW GLOBAL VARIABLES LIKE 'audit%';
```

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```

Remediation:

Setting this to `0` so that log files will be kept on the server indefinitely.

Default Value:

The default value is `0`.

References:

1. <https://docs.singlestore.com/db/v8.9/security/audit-logging/configuring-audit-logging/>
2. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
3. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

CIS Controls:

| Controls Version | Control | IG 1 | IG 2 | IG 3 |
|------------------|---|------|------|------|
| v8 | 8.10 <u>Retain Audit Logs</u> Retain audit logs across enterprise assets for a minimum of 90 days. | | ● | ● |
| v7 | 6.4 <u>Ensure adequate storage for logs</u> Ensure that all systems that store logs have adequate storage space for the logs generated. | | ● | ● |

6 Operating System Hardening

This section contains recommendations related to hardening the operating system running below SingleStore.

6.1 Ensure that SingleStore uses a non-default port (Automated)

Profile Applicability:

- Level 1

Description:

Changing the default port used by SingleStore makes it harder for attackers to find the database and target it.

Rationale:

Standard ports are used in automated attacks and by attackers to verify which applications are running on a server.

Impact:

Hackers frequently scan IP addresses for commonly used ports, so it's not uncommon to use a different port to "fly under the radar". This is just to avoid detection, other than that there is no added safety by using a different port.

Audit:

To verify the port number used by MongoDB, execute the following command and ensure that the port number is not **3306**:

```
grep port data/master/memsql.cnf
```

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```

Remediation:

Change the port for SingleStore server to a number other than **3306** and restart all nodes.

```
sdb-admin update-config --all --key "port" --value "3307"  
sdb-admin restart-node --all
```





Default Value:

The default port is **3306**.

References:

1. <https://docs.singlestore.com/db/v8.9/reference/configuration-reference/cluster-configuration/system-requirements-and-recommendations/#default-network-ports>
2. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
3. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

CIS Controls:

| Controls Version | Control | IG 1 | IG 2 | IG 3 |
|------------------|--|------|---|---|
| v8 | 4.8 <u>Uninstall or Disable Unnecessary Services on Enterprise Assets and Software</u> Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function. | |  |  |
| v7 | 9.2 <u>Ensure Only Approved Ports, Protocols and Services Are Running</u> Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system. | |  |  |

6.2 Ensure that operating system resource limits are set for SingleStore (Manual)

Profile Applicability:

- Level 2

Description:

Most Linux operating systems provide ways to control the usage of system resources such as threads, files and network at an individual user or process level.

Rationale:

The per-user limitations for resources are called ulimits, and they prevent single users from consuming too much system resources.

Audit:

To verify the resource limits set for MongoDB, run the following commands.
Extract the process ID for SingleStore:

```
ps -ef | grep memsqld
```

View the limits associated with the ID.

```
cat /proc/50/limits
```

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```


Remediation:

For optimal performance, SingleStore recommends setting `ulimits` to higher values than the default Linux settings. The `ulimit` settings can be configured in the `/etc/security/limits.conf` file, in the `/etc/security/limits.d` file, or directly via shell commands.

A SingleStore cluster uses a substantial number of client and server connections between aggregators and leaf nodes to run queries and cluster operations. SingleStore recommends setting the Linux file descriptor and maximum process limits to the values listed below to account for these connections. Failing to increase this limit can significantly degrade performance and even cause connection limit errors. The `ulimit` settings can be configured in the `/etc/security/limits.conf` file, or directly via shell commands.

Permanently increase the open files limit and the max user processes limit for the `memsql` user by editing the `/etc/security/limits.conf` file as the root user and adding the following lines:

| | | | |
|---------------------|-------------------|---------------------|----------------------|
| <code>memsql</code> | <code>soft</code> | <code>NOFILE</code> | <code>1024000</code> |
| <code>memsql</code> | <code>hard</code> | <code>NOFILE</code> | <code>1024000</code> |
| <code>memsql</code> | <code>soft</code> | <code>nproc</code> | <code>128000</code> |
| <code>memsql</code> | <code>hard</code> | <code>nproc</code> | <code>128000</code> |

Note: Each node must be restarted for the changed `ulimit` settings to take effect. The `file-max` setting configures the maximum number of file handles (file descriptor limit) for the entire system. On the contrary, `ulimit` settings are only enforced on a process level. Hence, the `file-max` value must be higher than the `NOFILE` setting. Increase the maximum number of file handles configured for the entire system in `/proc/sys/fs/file-max`. To make the change permanent, append or modify the `fs.file-max` line in the `/etc/sysctl.conf` file.





Default Value:

Not configured.

References:

1. <https://docs.singlestore.com/db/v8.9/reference/configuration-reference/cluster-configuration/system-requirements-and-recommendations/#configure-linux-ulimit-settings>
2. <https://docs.singlestore.com/db/v8.9/reference/configuration-reference/cluster-configuration/system-requirements-and-recommendations/#configure-file-descriptor-and-maximum-process-limits>
3. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
4. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

CIS Controls:

| Controls Version | Control | IG 1 | IG 2 | IG 3 |
|------------------|---|------|---|---|
| v8 | <p><u>16.7 Use Standard Hardening Configuration Templates for Application Infrastructure</u></p> <p>Use standard, industry-recommended hardening configuration templates for application infrastructure components. This includes underlying servers, databases, and web servers, and applies to cloud containers, Platform as a Service (PaaS) components, and SaaS components. Do not allow in-house developed software to weaken configuration hardening.</p> | |  |  |
| v7 | <p><u>8.3 Enable Operating System Anti-Exploitation Features/ Deploy Anti-Exploit Technologies</u></p> <p>Enable anti-exploitation features such as Data Execution Prevention (DEP) or Address Space Layout Randomization (ASLR) that are available in an operating system or deploy appropriate toolkits that can be configured to apply protection to a broader set of applications and executables.</p> | |  |  |

6.3 Configure Host-Based Security (Manual)

Profile Applicability:

- Level 2

Description:

Restrict network access to where SingleStoreDB will be/is installed via firewall or network policies (e.g. if using Amazon EC2 for running the cluster, security groups can be configured to allow inbound network access only from allowed IP address ranges).

You can also set the `bind-address` variable to restrict the range of IP addresses which are allowed to connect to SingleStore. For example, if you set it to `127.0.0.1`, you will only be able to connect to SingleStore locally. See `bind_address` in the referenced documentation for Non-Sync Variables List.

Rationale:

Restricting network access to SingleStore reduces the attack surface.

Audit:

Check the `bind-address` variable.

```
SELECT @@bind_address;
```

If the address is `0.0.0.0` (IPv4) or `::` (IPv6), SingleStore accepts connections on all network interfaces; otherwise, it only accepts connections for the given IP address. Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```

Remediation:

Set the `bind-address` variable to restrict connections to only a specific IP address.

```
sdb-admin update-config --all --key "bind_address" --value "127.0.0.1"
```

Note:

- When `allow_ipv6` is set to `true`, this variable can be set to an IPv6 address.

Changing this setting will require restarting all nodes.

```
sdb-admin restart-node --all
```





Default Value:

SingleStore accepts connections on all network interfaces by default, i.e. `bind-address` = `0.0.0.0` (or `::` in the case of IPv6 enabled).

References:

1. <https://docs.singlestore.com/db/v8.9/security/configuring-host-based-security/>
2. <https://docs.singlestore.com/db/v8.9/reference/configuration-reference/engine-variables/list-of-engine-variables/#non-sync-variables-list>
3. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
4. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

CIS Controls:

| Controls Version | Control | IG 1 | IG 2 | IG 3 |
|------------------|---|------|--|--|
| v8 | 4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function. | |  |  |
| v7 | 9.2 Ensure Only Approved Ports, Protocols and Services Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system. | |  |  |

6.4 Disable the Data API (Automated)

Profile Applicability:

- Level 2

Description:

SingleStore provides the Data API to execute SQL statements against your database over an HTTP connection. The Data API can be used to integrate SingleStore with serverless architecture, develop custom applications, and build seamless integrations with applications.

Rationale:

Unless necessary, disable Data API endpoints by setting the `http_api` variable to `OFF` on each aggregator node.

Audit:

Check if the data api is enabled:

```
grep http_api memsql.cnf
```

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```

Remediation:

Set the `data api` variable to `OFF` on each aggregator and restart all nodes.

```
sdb-admin update-config --all --key "http_api" --value "OFF"  
sdb-admin restart-node --all
```





Default Value:

The `data api` is `ON` by default.

References:

1. <https://docs.singlestore.com/db/v8.9/reference/configuration-reference/engine-variables/list-of-engine-variables/>
2. <https://docs.singlestore.com/db/v8.9/reference/data-api/>
3. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
4. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

CIS Controls:

| Controls Version | Control | IG 1 | IG 2 | IG 3 |
|------------------|---|------|---|---|
| v8 | <u>4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software</u> Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function. | |  |  |
| v7 | <u>9.2 Ensure Only Approved Ports, Protocols and Services Are Running</u> Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system. | |  |  |

6.5 Disable Code Engine (Automated)

Profile Applicability:

- Level 2

Description:

The Code Engine feature in SingleStore supports creating functions (UDFs and TVFs) using code compiled to WebAssembly (Wasm). The Code Engine uses the wasmtime runtime to compile and run WebAssembly code.

Rationale:

Unless necessary, disable Code Engine by setting `enable_wasm` to `OFF`.

Audit:

Check if the Code Engine is enabled:

```
grep enable_wasm memsql.cnf
```

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```

Remediation:

Set `enable_wasm` to `OFF`.

```
sdb-admin update-config --all --key "enable_wasm" --value "OFF"
```





Default Value:

The `Code Engine` is `OFF` by default.

References:

1. <https://docs.singlestore.com/db/v8.9/reference/configuration-reference/engine-variables/list-of-engine-variables/>
2. <https://docs.singlestore.com/db/v8.9/reference/code-engine-powered-by-wasm/>
3. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
4. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

CIS Controls:

| Controls Version | Control | IG 1 | IG 2 | IG 3 |
|------------------|---|------|---|---|
| v8 | <u>4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software</u> Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function. | |  |  |
| v7 | <u>9.2 Ensure Only Approved Ports, Protocols and Services Are Running</u> Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system. | |  |  |

6.6 Configure Dedicated Admin Connections (Automated)

Profile Applicability:

- Level 2

Description:

To log in to a dedicated admin connection, you should first log in to the machine where the master aggregator is running. If a master aggregator is not running, then log into a child aggregator.

Rationale:

This uses a Unix domain socket to designate that this is a dedicated admin connection. This is only available in the file system and cannot be used over a network, which is why you must log on to the aggregator before connecting using this method.

Once you log in, if commands do not run, you can kill connections and queries to free resources to allow others to connect to the system.

A similar approach will also allow you to log on to a leaf node directly, if necessary. The connection is node-local.

Audit:

Check this variables setting:

```
Select @@max_dedicated_admin_connections;
```

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```

Remediation:

Specifies the number of connections reserved for admin users (users granted the **SUPER** permission). This is the number of connections over and above the **max_connections**. Its purpose is to allow admin users to login even if **max_connections** number of non-admin users are already logged in. For example, if **max_connections** is set to **100** and **max_dedicated_admin_connections** is set to **5**, then even if **100** non-admin user connections are active at any time, **5** admin users can still log in. This is to prevent administrative users from being locked out of the system during heavy traffic.

```
sdb-admin update-config --all --key "max_dedicated_admin_connections" --value "5"
```





Default Value:

The default is **5**.

References:

1. <https://docs.singlestore.com/db/v8.9/reference/configuration-reference/engine-variables/list-of-engine-variables/>
2. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/dedicated-admin-connections/>
3. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
4. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

CIS Controls:

| Controls Version | Control | IG 1 | IG 2 | IG 3 |
|------------------|--|------|---|---|
| v8 | 4.8 <u>Uninstall or Disable Unnecessary Services on Enterprise Assets and Software</u> Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function. | |  |  |
| v7 | 9.2 <u>Ensure Only Approved Ports, Protocols and Services Are Running</u> Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system. | |  |  |

6.7 Configure hiding license-related variables (Automated)

Profile Applicability:

- Level 2

Description:

Hides license-related variables from **SHOW** commands. This variable can sync to all nodes, including aggregator and leaf nodes.

Rationale:

Use this variable if you would like to hide license-related variables from **SHOW** commands.

Audit:

Check this variables setting:

```
SELECT @@license_visibility;
```

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```

Remediation:

Configure this variable to your requirements.

```
sdb-admin update-config --all --key "license_visibility" --value "1"
```





Default Value:

The default is **ON**.

References:

1. <https://docs.singlestore.com/db/v8.9/reference/configuration-reference/engine-variables/list-of-engine-variables/>
2. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
3. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

CIS Controls:

| Controls Version | Control | IG 1 | IG 2 | IG 3 |
|------------------|---|------|---|---|
| v8 | <u>4.8 Uninstall or Disable Unnecessary Services on Enterprise Assets and Software</u> Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function. | |  |  |
| v7 | <u>9.2 Ensure Only Approved Ports, Protocols and Services Are Running</u> Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system. | |  |  |

7 File Permissions

This section provides recommendations for setting permissions for the key file and the database file.

7.1 Ensure appropriate key file permissions are set (Manual)

Profile Applicability:

- Level 1

Description:

In the Shared Cluster, the certificate or keyfile is utilized for authentication. Implementing proper file permissions on the certificate or keyfile will prevent unauthorized access.

Rationale:

Protecting the certificate and keyfile strengthens authentication in the sharded cluster and prevents unauthorized access to the SingleStore database.

Audit:

Find the location of the certificate and keyfile.

```
grep ssl memsql.cnf
```

Check the permissions of the files.

```
ls -l certificate_and_keyfile_locations
```

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```

Remediation:

Set the certificate and keyfile ownership to memsql.

```
sudo chown memsql:memsql certificate  
sudo chown memsql:memsql keyfile
```

Remove other permissions from the keyfile.

```
chmod 600 keyfile
```





Default Value:

Not configured.

References:

1. <https://docs.singlestore.com/db/v8.9/security/encryption/ssl-secure-connections/generating-ssl-certificates/>
2. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
3. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

CIS Controls:

| Controls Version | Control | IG 1 | IG 2 | IG 3 |
|------------------|---|------|---|---|
| v8 | 3.11 <u>Encrypt Sensitive Data at Rest</u> 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 | 16.4 <u>Encrypt or Hash all Authentication Credentials</u> Encrypt or hash with a salt all authentication credentials when stored. | |  |  |

7.2 Ensure appropriate data dir permissions are set. (Manual)

Profile Applicability:

- Level 1

Description:

The SingleStore data directory needs to be protected with the appropriate file permissions and ownership. This directory contains snapshots, logs, and columnstore segments. You can only set the `datadir` variable while the node is offline. The changes to this variable will take effect on the next start of the node.

Rationale:

This will restrict unauthorized users from accessing the SingleStore data directory.

Audit:

To verify that the permissions for the SingleStore data directory are configured securely, run the following commands.
Identify the nodes.

```
sdb-admin list-nodes
```

Describe a node based on a **memsql ID**.

```
sdb-admin describe-node --memsql-id 19F394927E | grep datadir
```

List the ownership and permissions of the **datadir**.

```
ls -al datadir
```

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```

Remediation:

Set ownership of the data directory to memsql and remove other permissions using the following commands.

```
chmod 750 datadir  
chown memsql:memsql datadir
```







Default Value:

Not configured.

References:

1. <https://docs.singlestore.com/db/v8.9/reference/configuration-reference/engine-variables/list-of-engine-variables/>
2. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/maintain-your-cluster/changing-the-default-data-directory-post-installation-for-a-leaf-node/>
3. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
4. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

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. |  |  |  |

7.3 Ensure appropriate FILE READ and FILE WRITE privileges for users. (Manual)

Profile Applicability:

- Level 1

Description:

The `secure_file_priv` global variable controls where users with the `FILE READ` and `FILE WRITE` privilege can read or save files. It should be set on all nodes to a directory that is not used by SingleStore or other software.

Rationale:

If not set, a user with the `FILE READ` or `FILE WRITE` privilege can tamper with the system by reading or creating files in sensitive locations. For information on how to set engine variables, see the engine variables overview.

Audit:

Check the `secure_file_priv` engine variable.

```
SELECT @@secure_file_priv;
```

Optionally, install the latest version of SingleStore Tools and generate the security recommendations report to review the security configuration.

```
sdb-report collect-and-check --security-recommendations
```

Remediation:

Specifies the directory to which any import or export operations should be limited.

```
sdb-admin update-config --all --key "secure_file_priv" --value  
"/home/singlestore"
```

Changing this setting will require restarting all nodes.

```
sdb-admin restart-node --all
```







Default Value:

By default this variable is set to `NULL` which allows unrestricted import and export operations.

References:

1. <https://docs.singlestore.com/db/v8.9/security/setting-secure-file-priv/>
2. <https://docs.singlestore.com/db/v8.9/reference/configuration-reference/engine-variables/>
3. <https://docs.singlestore.com/db/v8.9/reference/configuration-reference/engine-variables/list-of-engine-variables/>
4. <https://docs.singlestore.com/db/v8.9/user-and-cluster-administration/cluster-management-with-tools/singlestore-tools-installation/>
5. <https://docs.singlestore.com/db/v8.9/reference/singlestore-tools-reference/sdb-report-commands/check/>

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. |  |  |  |

Appendix: Summary Table

| CIS Benchmark Recommendation | | Set Correctly | |
|------------------------------|---|--------------------------|--------------------------|
| | | Yes | No |
| 1 | Installation and Patching | | |
| 1.1 | Perform a binary signature check on the SingleStore packages (Manual) | <input type="checkbox"/> | <input type="checkbox"/> |
| 1.2 | Ensure the appropriate SingleStore software version/patches are installed (Manual) | <input type="checkbox"/> | <input type="checkbox"/> |
| 2 | Authentication | | |
| 2.1 | Configure secure root password (Automated) | <input type="checkbox"/> | <input type="checkbox"/> |
| 2.2 | Configure user authentication (Automated) | <input type="checkbox"/> | <input type="checkbox"/> |
| 2.3 | Generate TLS Certificates for SingleStore node authentication (Automated) | <input type="checkbox"/> | <input type="checkbox"/> |
| 2.4 | Configure secure client and intra-cluster connections (Automated) (Automated) | <input type="checkbox"/> | <input type="checkbox"/> |
| 2.5 | Enable Wire Encryption and Kerberos on HDFS Pipelines (Manual) | <input type="checkbox"/> | <input type="checkbox"/> |
| 3 | Authorization | | |
| 3.1 | Ensure least privilege for database accounts (Manual) | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.2 | Ensure that role-based access control is enabled and configured appropriately (Manual) | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.3 | Ensure that SingleStore is run using a non-privileged, dedicated service account (Manual) | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.4 | Ensure that each role for each SingleStore database is needed and grants only the necessary privileges (Manual) | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.5 | Review Superuser/Admin Roles (Manual) | <input type="checkbox"/> | <input type="checkbox"/> |

| CIS Benchmark Recommendation | | Set Correctly | |
|------------------------------|---|--------------------------|--------------------------|
| | | Yes | No |
| 3.6 | Delete Unnecessary Default Users (Manual) | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.7 | Synchronize Permissions Across Your Cluster (Manual) | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.8 | Implement Row-Level Security (RLS) (Manual) | <input type="checkbox"/> | <input type="checkbox"/> |
| 4 | Data Encryption | | |
| 4.1 | Ensure legacy TLS protocols are disabled (Automated) | <input type="checkbox"/> | <input type="checkbox"/> |
| 4.2 | Ensure Encryption of Data in Transit TLS (Transport Encryption) (Automated) | <input type="checkbox"/> | <input type="checkbox"/> |
| 4.3 | Ensure Federal Information Processing Standard (FIPS) is enabled (Automated) | <input type="checkbox"/> | <input type="checkbox"/> |
| 4.4 | Ensure Encryption of Data at Rest (Manual) | <input type="checkbox"/> | <input type="checkbox"/> |
| 4.5 | Configure Encryption Mode (Automated) | <input type="checkbox"/> | <input type="checkbox"/> |
| 5 | Audit Logging | | |
| 5.1 | Ensure that system activity is audited (Automated) | <input type="checkbox"/> | <input type="checkbox"/> |
| 5.2 | Ensure that audit filters are configured properly (Manual) | <input type="checkbox"/> | <input type="checkbox"/> |
| 5.3 | Ensure that log entries are preserved (Automated) | <input type="checkbox"/> | <input type="checkbox"/> |
| 6 | Operating System Hardening | | |
| 6.1 | Ensure that SingleStore uses a non-default port (Automated) | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.2 | Ensure that operating system resource limits are set for SingleStore (Manual) | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.3 | Configure Host-Based Security (Manual) | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.4 | Disable the Data API (Automated) | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.5 | Disable Code Engine (Automated) | <input type="checkbox"/> | <input type="checkbox"/> |

| CIS Benchmark Recommendation | | Set Correctly | |
|------------------------------|--|--------------------------|--------------------------|
| | | Yes | No |
| 6.6 | Configure Dedicated Admin Connections (Automated) | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.7 | Configure hiding license-related variables (Automated) | <input type="checkbox"/> | <input type="checkbox"/> |
| 7 | File Permissions | | |
| 7.1 | Ensure appropriate key file permissions are set (Manual) | <input type="checkbox"/> | <input type="checkbox"/> |
| 7.2 | Ensure appropriate data dir permissions are set. (Manual) | <input type="checkbox"/> | <input type="checkbox"/> |
| 7.3 | Ensure appropriate FILE READ and FILE WRITE privileges for users. (Manual) | <input type="checkbox"/> | <input type="checkbox"/> |

Appendix: CIS Controls v7 IG 1 Mapped Recommendations

| Recommendation | | Set Correctly | |
|----------------|--|--------------------------|--------------------------|
| | | Yes | No |
| 1.2 | Ensure the appropriate SingleStore software version/patches are installed | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.1 | Ensure least privilege for database accounts | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.2 | Ensure that role-based access control is enabled and configured appropriately | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.3 | Ensure that SingleStore is run using a non-privileged, dedicated service account | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.4 | Ensure that each role for each SingleStore database is needed and grants only the necessary privileges | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.5 | Review Superuser/Admin Roles | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.6 | Delete Unnecessary Default Users | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.7 | Synchronize Permissions Across Your Cluster | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.8 | Implement Row-Level Security (RLS) | <input type="checkbox"/> | <input type="checkbox"/> |
| 5.1 | Ensure that system activity is audited | <input type="checkbox"/> | <input type="checkbox"/> |
| 5.2 | Ensure that audit filters are configured properly | <input type="checkbox"/> | <input type="checkbox"/> |
| 7.2 | Ensure appropriate data dir permissions are set. | <input type="checkbox"/> | <input type="checkbox"/> |
| 7.3 | Ensure appropriate FILE READ and FILE WRITE privileges for users. | <input type="checkbox"/> | <input type="checkbox"/> |

Appendix: CIS Controls v7 IG 2 Mapped Recommendations

| Recommendation | | Set Correctly | |
|----------------|--|--------------------------|--------------------------|
| | | Yes | No |
| 1.2 | Ensure the appropriate SingleStore software version/patches are installed | <input type="checkbox"/> | <input type="checkbox"/> |
| 2.2 | Configure user authentication | <input type="checkbox"/> | <input type="checkbox"/> |
| 2.3 | Generate TLS Certificates for SingleStore node authentication | <input type="checkbox"/> | <input type="checkbox"/> |
| 2.5 | Enable Wire Encryption and Kerberos on HDFS Pipelines | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.1 | Ensure least privilege for database accounts | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.2 | Ensure that role-based access control is enabled and configured appropriately | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.3 | Ensure that SingleStore is run using a non-privileged, dedicated service account | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.4 | Ensure that each role for each SingleStore database is needed and grants only the necessary privileges | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.5 | Review Superuser/Admin Roles | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.6 | Delete Unnecessary Default Users | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.7 | Synchronize Permissions Across Your Cluster | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.8 | Implement Row-Level Security (RLS) | <input type="checkbox"/> | <input type="checkbox"/> |
| 4.1 | Ensure legacy TLS protocols are disabled | <input type="checkbox"/> | <input type="checkbox"/> |
| 4.2 | Ensure Encryption of Data in Transit TLS (Transport Encryption) | <input type="checkbox"/> | <input type="checkbox"/> |
| 4.3 | Ensure Federal Information Processing Standard (FIPS) is enabled | <input type="checkbox"/> | <input type="checkbox"/> |
| 5.1 | Ensure that system activity is audited | <input type="checkbox"/> | <input type="checkbox"/> |
| 5.2 | Ensure that audit filters are configured properly | <input type="checkbox"/> | <input type="checkbox"/> |
| 5.3 | Ensure that log entries are preserved | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.1 | Ensure that SingleStore uses a non-default port | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.2 | Ensure that operating system resource limits are set for SingleStore | <input type="checkbox"/> | <input type="checkbox"/> |

| Recommendation | | Set Correctly | |
|----------------|---|--------------------------|--------------------------|
| | | Yes | No |
| 6.3 | Configure Host-Based Security | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.4 | Disable the Data API | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.5 | Disable Code Engine | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.6 | Configure Dedicated Admin Connections | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.7 | Configure hiding license-related variables | <input type="checkbox"/> | <input type="checkbox"/> |
| 7.1 | Ensure appropriate key file permissions are set | <input type="checkbox"/> | <input type="checkbox"/> |
| 7.2 | Ensure appropriate data dir permissions are set. | <input type="checkbox"/> | <input type="checkbox"/> |
| 7.3 | Ensure appropriate FILE READ and FILE WRITE privileges for users. | <input type="checkbox"/> | <input type="checkbox"/> |

Appendix: CIS Controls v7 IG 3 Mapped Recommendations

| Recommendation | | Set Correctly | |
|----------------|--|--------------------------|--------------------------|
| | | Yes | No |
| 1.1 | Perform a binary signature check on the SingleStore packages | <input type="checkbox"/> | <input type="checkbox"/> |
| 1.2 | Ensure the appropriate SingleStore software version/patches are installed | <input type="checkbox"/> | <input type="checkbox"/> |
| 2.2 | Configure user authentication | <input type="checkbox"/> | <input type="checkbox"/> |
| 2.3 | Generate TLS Certificates for SingleStore node authentication | <input type="checkbox"/> | <input type="checkbox"/> |
| 2.4 | Configure secure client and intra-cluster connections (Automated) | <input type="checkbox"/> | <input type="checkbox"/> |
| 2.5 | Enable Wire Encryption and Kerberos on HDFS Pipelines | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.1 | Ensure least privilege for database accounts | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.2 | Ensure that role-based access control is enabled and configured appropriately | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.3 | Ensure that SingleStore is run using a non-privileged, dedicated service account | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.4 | Ensure that each role for each SingleStore database is needed and grants only the necessary privileges | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.5 | Review Superuser/Admin Roles | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.6 | Delete Unnecessary Default Users | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.7 | Synchronize Permissions Across Your Cluster | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.8 | Implement Row-Level Security (RLS) | <input type="checkbox"/> | <input type="checkbox"/> |
| 4.1 | Ensure legacy TLS protocols are disabled | <input type="checkbox"/> | <input type="checkbox"/> |
| 4.2 | Ensure Encryption of Data in Transit TLS (Transport Encryption) | <input type="checkbox"/> | <input type="checkbox"/> |
| 4.3 | Ensure Federal Information Processing Standard (FIPS) is enabled | <input type="checkbox"/> | <input type="checkbox"/> |
| 4.4 | Ensure Encryption of Data at Rest | <input type="checkbox"/> | <input type="checkbox"/> |
| 4.5 | Configure Encryption Mode | <input type="checkbox"/> | <input type="checkbox"/> |
| 5.1 | Ensure that system activity is audited | <input type="checkbox"/> | <input type="checkbox"/> |

| Recommendation | | Set Correctly | |
|----------------|--|--------------------------|--------------------------|
| | | Yes | No |
| 5.2 | Ensure that audit filters are configured properly | <input type="checkbox"/> | <input type="checkbox"/> |
| 5.3 | Ensure that log entries are preserved | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.1 | Ensure that SingleStore uses a non-default port | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.2 | Ensure that operating system resource limits are set for SingleStore | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.3 | Configure Host-Based Security | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.4 | Disable the Data API | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.5 | Disable Code Engine | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.6 | Configure Dedicated Admin Connections | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.7 | Configure hiding license-related variables | <input type="checkbox"/> | <input type="checkbox"/> |
| 7.1 | Ensure appropriate key file permissions are set | <input type="checkbox"/> | <input type="checkbox"/> |
| 7.2 | Ensure appropriate data dir permissions are set. | <input type="checkbox"/> | <input type="checkbox"/> |
| 7.3 | Ensure appropriate FILE READ and FILE WRITE privileges for users. | <input type="checkbox"/> | <input type="checkbox"/> |

Appendix: CIS Controls v7 Unmapped Recommendations

| Recommendation | | Set Correctly | |
|----------------|--------------------------------|--------------------------|--------------------------|
| | | Yes | No |
| 2.1 | Configure secure root password | <input type="checkbox"/> | <input type="checkbox"/> |

Appendix: CIS Controls v8 IG 1 Mapped Recommendations

| Recommendation | | Set Correctly | |
|----------------|--|--------------------------|--------------------------|
| | | Yes | No |
| 1.2 | Ensure the appropriate SingleStore software version/patches are installed | <input type="checkbox"/> | <input type="checkbox"/> |
| 2.1 | Configure secure root password | <input type="checkbox"/> | <input type="checkbox"/> |
| 2.2 | Configure user authentication | <input type="checkbox"/> | <input type="checkbox"/> |
| 2.3 | Generate TLS Certificates for SingleStore node authentication | <input type="checkbox"/> | <input type="checkbox"/> |
| 2.5 | Enable Wire Encryption and Kerberos on HDFS Pipelines | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.1 | Ensure least privilege for database accounts | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.4 | Ensure that each role for each SingleStore database is needed and grants only the necessary privileges | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.5 | Review Superuser/Admin Roles | <input type="checkbox"/> | <input type="checkbox"/> |
| 4.1 | Ensure legacy TLS protocols are disabled | <input type="checkbox"/> | <input type="checkbox"/> |
| 5.1 | Ensure that system activity is audited | <input type="checkbox"/> | <input type="checkbox"/> |
| 5.2 | Ensure that audit filters are configured properly | <input type="checkbox"/> | <input type="checkbox"/> |
| 7.2 | Ensure appropriate data dir permissions are set. | <input type="checkbox"/> | <input type="checkbox"/> |
| 7.3 | Ensure appropriate FILE READ and FILE WRITE privileges for users. | <input type="checkbox"/> | <input type="checkbox"/> |

Appendix: CIS Controls v8 IG 2 Mapped Recommendations

| Recommendation | | Set Correctly | |
|----------------|--|--------------------------|--------------------------|
| | | Yes | No |
| 1.1 | Perform a binary signature check on the SingleStore packages | <input type="checkbox"/> | <input type="checkbox"/> |
| 1.2 | Ensure the appropriate SingleStore software version/patches are installed | <input type="checkbox"/> | <input type="checkbox"/> |
| 2.1 | Configure secure root password | <input type="checkbox"/> | <input type="checkbox"/> |
| 2.2 | Configure user authentication | <input type="checkbox"/> | <input type="checkbox"/> |
| 2.3 | Generate TLS Certificates for SingleStore node authentication | <input type="checkbox"/> | <input type="checkbox"/> |
| 2.4 | Configure secure client and intra-cluster connections (Automated) | <input type="checkbox"/> | <input type="checkbox"/> |
| 2.5 | Enable Wire Encryption and Kerberos on HDFS Pipelines | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.1 | Ensure least privilege for database accounts | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.3 | Ensure that SingleStore is run using a non-privileged, dedicated service account | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.4 | Ensure that each role for each SingleStore database is needed and grants only the necessary privileges | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.5 | Review Superuser/Admin Roles | <input type="checkbox"/> | <input type="checkbox"/> |
| 4.1 | Ensure legacy TLS protocols are disabled | <input type="checkbox"/> | <input type="checkbox"/> |
| 4.2 | Ensure Encryption of Data in Transit TLS (Transport Encryption) | <input type="checkbox"/> | <input type="checkbox"/> |
| 4.3 | Ensure Federal Information Processing Standard (FIPS) is enabled | <input type="checkbox"/> | <input type="checkbox"/> |
| 4.4 | Ensure Encryption of Data at Rest | <input type="checkbox"/> | <input type="checkbox"/> |
| 4.5 | Configure Encryption Mode | <input type="checkbox"/> | <input type="checkbox"/> |
| 5.1 | Ensure that system activity is audited | <input type="checkbox"/> | <input type="checkbox"/> |
| 5.2 | Ensure that audit filters are configured properly | <input type="checkbox"/> | <input type="checkbox"/> |
| 5.3 | Ensure that log entries are preserved | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.1 | Ensure that SingleStore uses a non-default port | <input type="checkbox"/> | <input type="checkbox"/> |

| Recommendation | | Set Correctly | |
|----------------|--|--------------------------|--------------------------|
| | | Yes | No |
| 6.2 | Ensure that operating system resource limits are set for SingleStore | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.3 | Configure Host-Based Security | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.4 | Disable the Data API | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.5 | Disable Code Engine | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.6 | Configure Dedicated Admin Connections | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.7 | Configure hiding license-related variables | <input type="checkbox"/> | <input type="checkbox"/> |
| 7.1 | Ensure appropriate key file permissions are set | <input type="checkbox"/> | <input type="checkbox"/> |
| 7.2 | Ensure appropriate data dir permissions are set. | <input type="checkbox"/> | <input type="checkbox"/> |
| 7.3 | Ensure appropriate FILE READ and FILE WRITE privileges for users. | <input type="checkbox"/> | <input type="checkbox"/> |

Appendix: CIS Controls v8 IG 3 Mapped Recommendations

| Recommendation | | Set Correctly | |
|----------------|--|--------------------------|--------------------------|
| | | Yes | No |
| 1.1 | Perform a binary signature check on the SingleStore packages | <input type="checkbox"/> | <input type="checkbox"/> |
| 1.2 | Ensure the appropriate SingleStore software version/patches are installed | <input type="checkbox"/> | <input type="checkbox"/> |
| 2.1 | Configure secure root password | <input type="checkbox"/> | <input type="checkbox"/> |
| 2.2 | Configure user authentication | <input type="checkbox"/> | <input type="checkbox"/> |
| 2.3 | Generate TLS Certificates for SingleStore node authentication | <input type="checkbox"/> | <input type="checkbox"/> |
| 2.4 | Configure secure client and intra-cluster connections (Automated) | <input type="checkbox"/> | <input type="checkbox"/> |
| 2.5 | Enable Wire Encryption and Kerberos on HDFS Pipelines | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.1 | Ensure least privilege for database accounts | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.2 | Ensure that role-based access control is enabled and configured appropriately | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.3 | Ensure that SingleStore is run using a non-privileged, dedicated service account | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.4 | Ensure that each role for each SingleStore database is needed and grants only the necessary privileges | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.5 | Review Superuser/Admin Roles | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.6 | Delete Unnecessary Default Users | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.7 | Synchronize Permissions Across Your Cluster | <input type="checkbox"/> | <input type="checkbox"/> |
| 3.8 | Implement Row-Level Security (RLS) | <input type="checkbox"/> | <input type="checkbox"/> |
| 4.1 | Ensure legacy TLS protocols are disabled | <input type="checkbox"/> | <input type="checkbox"/> |
| 4.2 | Ensure Encryption of Data in Transit TLS (Transport Encryption) | <input type="checkbox"/> | <input type="checkbox"/> |
| 4.3 | Ensure Federal Information Processing Standard (FIPS) is enabled | <input type="checkbox"/> | <input type="checkbox"/> |
| 4.4 | Ensure Encryption of Data at Rest | <input type="checkbox"/> | <input type="checkbox"/> |
| 4.5 | Configure Encryption Mode | <input type="checkbox"/> | <input type="checkbox"/> |

| Recommendation | | Set Correctly | |
|----------------|--|--------------------------|--------------------------|
| | | Yes | No |
| 5.1 | Ensure that system activity is audited | <input type="checkbox"/> | <input type="checkbox"/> |
| 5.2 | Ensure that audit filters are configured properly | <input type="checkbox"/> | <input type="checkbox"/> |
| 5.3 | Ensure that log entries are preserved | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.1 | Ensure that SingleStore uses a non-default port | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.2 | Ensure that operating system resource limits are set for SingleStore | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.3 | Configure Host-Based Security | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.4 | Disable the Data API | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.5 | Disable Code Engine | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.6 | Configure Dedicated Admin Connections | <input type="checkbox"/> | <input type="checkbox"/> |
| 6.7 | Configure hiding license-related variables | <input type="checkbox"/> | <input type="checkbox"/> |
| 7.1 | Ensure appropriate key file permissions are set | <input type="checkbox"/> | <input type="checkbox"/> |
| 7.2 | Ensure appropriate data dir permissions are set. | <input type="checkbox"/> | <input type="checkbox"/> |
| 7.3 | Ensure appropriate FILE READ and FILE WRITE privileges for users. | <input type="checkbox"/> | <input type="checkbox"/> |

Appendix: CIS Controls v8 Unmapped Recommendations

| Recommendation | | Set Correctly | |
|----------------|--|--------------------------|--------------------------|
| | | Yes | No |
| | No unmapped recommendations to CIS Controls v8 | <input type="checkbox"/> | <input type="checkbox"/> |

Appendix: Change History

| Date | Version | Changes for this version |
|-----------|---------|--------------------------|
| 5/14/2025 | 1.0.0 | Initial release |