CIS MS SQL Server 2016 Instance STIG Benchmark

v1.0.0 - 09-12-2025

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

Target Technology Details

MS SQL Server 2016 Instance Secure Technical Implementation Guide (STIG)

Version: 3 Release: 5 Benchmark

Date: 02 Jul 2025

Intended Audience

This benchmark is intended for system and application administrators, security specialists, auditors, help desk, and platform deployment personnel who plan to develop, deploy, assess, or secure solutions that incorporate MS SQL Server 2016 Instance and are looking to comply with the STIG guidance

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 (or rules) for securing a technology or a supporting platform. STIG Benchmark profiles are used to identify which Vulnerability Severity Category Code (CAT) each rule is associated with.

Description

The Rule Title from the STIG.

Rationale Statement

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

Additional Information

References from the STIG Rule if applicable.

Profile Definitions

The following configuration profiles are defined by this Benchmark:

- **SEVERITY: CAT I**

  Items in this profile exhibit one or more of the following characteristics:

  - are considered to be high severity
  - are intended for environments or use cases where following STIG based security guidance is paramount.
  - acts as defense in depth measure.
  - may negatively inhibit the utility or performance of the technology.

  This profile is intended for servers and workstations.

- **SEVERITY: CAT II**

  Items in this profile exhibit one or more of the following characteristics:

  - are considered to be medium severity
  - are intended for environments or use cases where following STIG based security guidance is paramount.
  - acts as defense in depth measure.
  - may negatively inhibit the utility or performance of the technology.

  This profile is intended for servers and workstations.

- **SEVERITY: CAT III**

  Items in this profile exhibit one or more of the following characteristics:

  - are considered to be low severity
  - are intended for environments or use cases where following STIG based security guidance is paramount.
  - acts as defense in depth measure.
  - may negatively inhibit the utility or performance of the technology.

  This profile is intended for servers and workstations.

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:

The Recommendations in this Benchmark are a representation of the Rules in the unclassified DISA STIG for MS SQL Server 2016 Instance

Contributor

Tim Harrison CISSP, ICP, KMP, Center for Internet Security, New York

Recommendations

1 STIG RULES

Microsoft SQL Server 2016

MS SQL Server 2016 Instance Secure Technical Implementation Guide (STIG)

Version: 3 Release: 5 Benchmark

Date: 02 Jul 2025

CLASSIFICATION unclassified

1.1 SQL6-D0-003600 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must limit the number of concurrent sessions to an organization-defined number per user for all accounts and/or account types.

```
GROUP ID: V-213929
RULE ID: SV-213929r1018580
```

Rationale:

Database management includes the ability to control the number of users and user sessions utilizing SQL Server. Unlimited concurrent connections to SQL Server could allow a successful denial-of-service (DoS) attack by exhausting connection resources; and a system can also fail or be degraded by an overload of legitimate users. Limiting the number of concurrent sessions is helpful in reducing these risks.

This requirement addresses concurrent session control for a single account. It does not address concurrent sessions by a single user via multiple system accounts.

The capability to limit the number of concurrent sessions per user must be configured in or added to SQL Server (for example, by use of a logon trigger), when this is technically feasible. Note that it is not sufficient to limit sessions via a web server or application server alone, because legitimate users and adversaries can potentially connect to SQL Server by other means.

The organization will need to define the maximum number of concurrent sessions by account type, by account, or a combination thereof. In deciding on the appropriate number, it is important to consider the work requirements of the various types of users. For example, 2 might be an acceptable limit for general users accessing the database via an application; but 10 might be too few for a database administrator using a database management GUI tool, where each query tab and navigation pane may count as a separate session.

(Sessions may also be referred to as connections or logons, which for the purposes of this requirement are synonyms.)

Audit:

Review the system documentation to determine whether any concurrent session limits have been defined. If it does not, assume a limit of 10 for database administrators and 2 for all other users.

If a mechanism other than a logon trigger is used, verify its correct operation by the appropriate means. If it does not work correctly, this is a finding.

Due to excessive CPU consumption when utilizing a logon trigger, an alternative method of limiting concurrent sessions is setting the max connection limit within SQL Server to an appropriate value. This serves to block a distributed denial-of-service (DDOS) attack by limiting the attacker's connections while allowing a database administrator to still force a SQL connection.

In SQL Server Management Studio's Object Explorer tree: Right-click on the Server Name >> Select Properties >> Select Connections Tab

OR

Run the query:

```
EXEC sys.sp_configure N'user connections'
```

If the max connection limit is set to 0 (unlimited) or does not match the documented value, this is a finding.

Otherwise, determine if a logon trigger exists:

In SQL Server Management Studio's Object Explorer tree:

Expand [SQL Server Instance] >> Server Objects >> Triggers

OR

Run the query:

```
SELECT name FROM master.sys.server_triggers;
```

If no triggers are listed, this is a finding.

If triggers are listed, identify the trigger(s) limiting the number of concurrent sessions per user. If none are found, this is a finding. If they are present but disabled, this is a finding.

Examine the trigger source code for logical correctness and for compliance with the documented limit(s). If errors or variances exist, this is a finding.

Verify that the system does execute the trigger(s) each time a user session is established. If it does not operate correctly for all types of user, this is a finding.

Remediation:

If a trigger consumes too much CPU, an example alternative method for limiting the concurrent users is setting the max connection limit on SQL Server.

In SQL Server Management Studio's Object Explorer tree, right-click on the Server Name >> Select Properties >> Select Connections Tab >> Set the Maximum Number of Concurrent Connections to a value other than 0 (0 = unlimited), and document it.

OR

Run the query:

```
EXEC sys.sp_configure N'user connections','5000' /* this is an example max
limit */
```

```
GO
RECONFIGURE WITH OVERRIDE
GO
```

Restart SQL Server for the setting to take effect.

Reference: https://learn.microsoft.com/en-us/sql/database-engine/configure-windows/configure-the-user-connections-server-configuration-option?

Otherwise, establish the limit(s) appropriate to the type(s) of user account accessing the SQL Server instance, and record them in the system documentation. Implement one or more logon triggers to enforce the limit(s), without exposing the dynamic management views to general users.

Example script below:

```
CREATE TRIGGER SQL_STIG_Connection_Limit
ON ALL SERVER WITH EXECUTE AS 'renamed_sa' /*Make sure to use the renamed SA
account here*/
FOR LOGON
AS
BEGIN
If (Select COUNT(1) from sys.dm_exec_sessions WHERE is_user_process = 1 AND
original_login_name = ORIGINAL_LOGIN() ) >
  (CASE ORIGINAL_LOGIN()
              WHEN 'domain/ima.dba' THEN 150    /*this is a busy DBA's domain
account */
              WHEN 'application1_login' THEN 6  /* this is a SQL login for an
application */
              WHEN 'application2_login' THEN 20 /* this is a SQL login for
another application */
              …
              ELSE 1 /* All unspecified users are restricted to a single
login */
  END)
              BEGIN
                           PRINT 'The login [' + ORIGINAL_LOGIN() + '] has
exceeded its concurrent session limit.'
                           ROLLBACK;
              END
END;
```

Reference: https://msdn.microsoft.com/en-us/library/ms189799.aspx

Additional Information:

CCI-000054

Limit the number of concurrent sessions for each organization-defined account and/or account type to an organization-defined number.

- NIST SP 800-53::AC-10
- NIST SP 800-53A::AC-10.1 (ii)
- NIST SP 800-53 Revision 4::AC-10
- NIST SP 800-53 Revision 5::AC-10

1.2 SQL6-D0-003700 (Manual)

Profile Applicability:

- SEVERITY: CAT I

Description:

SQL Server must integrate with an organization-level authentication/access mechanism providing account management and automation for all users, groups, roles, and any other principals.

```
GROUP ID: V-213930
RULE ID: SV-213930r1043176
```

Rationale:

Enterprise environments make account management for applications and databases challenging and complex. A manual process for account management functions adds the risk of a potential oversight or other error. Managing accounts for the same person in multiple places is inefficient and prone to problems with consistency and synchronization.

A comprehensive application account management process that includes automation helps to ensure that accounts designated as requiring attention are consistently and promptly addressed.

Examples include, but are not limited to, using automation to take action on multiple accounts designated as inactive, suspended, or terminated, or by disabling accounts located in non-centralized account stores, such as multiple servers. Account management functions can also include: assignment of group or role membership; identifying account type; specifying user access authorizations (i.e., privileges); account removal, update, or termination; and administrative alerts. The use of automated mechanisms can include, for example, using email or text messaging to notify account managers when users are terminated or transferred; using the information system to monitor account usage; and using automated telephone notification to report atypical system account usage.

SQL Server must be configured to automatically utilize organization-level account management functions, and these functions must immediately enforce the organization's current account policy.

Automation may be composed of differing technologies that when placed together contain an overall mechanism supporting an organization's automated account management requirements.

Audit:

Determine whether SQL Server is configured to use only Windows authentication.

In the Object Explorer in SQL Server Management Studio (SSMS), right-click on the server instance.

- Select "Properties".
- Select the Security page.

If Windows Authentication Mode is selected, this is not a finding.

OR

In a query interface such as the SSMS Transact-SQL editor, run the statement:

```
SELECT CASE SERVERPROPERTY('IsIntegratedSecurityOnly')
WHEN 1 THEN 'Windows Authentication'
WHEN 0 THEN 'Windows and SQL Server Authentication'
END as [Authentication Mode]
```

If the returned value in the `Authentication Mode` column is `Windows Authentication`, this is not a finding.

Mixed mode (both SQL Server authentication and Windows authentication) is in use. If the need for mixed mode has not been documented and approved by the information system security officer (ISSO)/information system security manager (ISSM), this is a finding.

From the documentation, obtain the list of accounts authorized to be managed by SQL Server.

Determine the accounts (SQL Logins) actually managed by SQL Server. Run the statement:

```
SELECT name
FROM sys.sql_logins
WHERE type_desc = 'SQL_LOGIN' AND is_disabled = 0;
```

If any accounts listed by the query are not listed in the documentation, this is a finding.

Remediation:

If mixed mode is required, document the need and justification; describe the measures taken to ensure the use of SQL Server authentication is kept to a minimum; describe the measures taken to safeguard passwords; and list or describe the SQL Logins used.

Documentation must be approved by the ISSO/ISSM.

If mixed mode is not required, disable it as follows:

- In the SSMS Object Explorer, right-click on the server instance.
- Select "Properties".
- Select the Security page.
- Click on the radio button for "Windows Authentication Mode".
- Click on "OK".
- Restart the SQL Server instance.

OR

Run the statement:

```
USE [master]
EXEC xp_instance_regwrite N'HKEY_LOCAL_MACHINE',
N'Software\Microsoft\MSSQLServer\MSSQLServer', N'LoginMode', REG_DWORD, 2
GO
```

Restart the SQL Server instance.

For each account being managed by SQL Server but not requiring it, drop or disable the SQL Login. Replace it with an appropriately configured account, as needed.

To drop or disable a login in the SSMS Object Explorer:

- Navigate to "Security Logins".
- Right-click on the login name and click "Delete" or "Disable".

To drop or disable a login by using a query:

```
USE master;
DROP LOGIN login_name;
ALTER LOGIN login_name DISABLE;
```

Dropping a login does not delete the equivalent database user(s). There may be more than one database containing a user mapped to the login. Drop the user(s) unless still needed.

To drop a user in the SSMS Object Explorer:

- Navigate to Databases >> Security Users.
- Right-click on the user name.
- Click "Delete".

To drop a user via a query:

```
USE database_name;
DROP USER <user_name>;
```

Additional Information:

CCI-000015

Support the management of system accounts using organization-defined automated mechanisms.

- NIST SP 800-53::AC-2 (1)
- NIST SP 800-53A::AC-2 (1).1
- NIST SP 800-53 Revision 4::AC-2 (1)
- NIST SP 800-53 Revision 5::AC-2 (1)

1.3 SQL6-D0-003800 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must be configured to utilize the most-secure authentication method available.

```
GROUP ID: V-213931
RULE ID: SV-213931r1043176
```

Rationale:

Enterprise environments make account management for applications and databases challenging and complex. A manual process for account management functions adds the risk of a potential oversight or other error. Managing accounts for the same person in multiple places is inefficient and prone to problems with consistency and synchronization.

A comprehensive application account management process that includes automation helps to ensure that accounts designated as requiring attention are consistently and promptly addressed.

Examples include, but are not limited to, using automation to take action on multiple accounts designated as inactive, suspended, or terminated, or by disabling accounts located in non-centralized account stores, such as multiple servers. Account management functions can also include: assignment of group or role membership; identifying account type; specifying user access authorizations (i.e., privileges); account removal, update, or termination; and administrative alerts. The use of automated mechanisms can include, for example: using email or text messaging to notify account managers when users are terminated or transferred; using the information system to monitor account usage; and using automated telephone notification to report atypical system account usage.

SQL Server must be configured to automatically utilize organization-level account management functions, and these functions must immediately enforce the organization's current account policy.

Automation may be comprised of differing technologies that when placed together contain an overall mechanism supporting an organization's automated account management requirements.

SQL Server supports several authentication methods to allow operation in various environments, Kerberos, NTLM, and SQL Server. An instance of SQL Server must be configured to utilize the most-secure method available. Service accounts utilized by SQL Server should be unique to a given instance.

Audit:

If the SQL Server is not part of an Active Directory domain, this finding is Not Applicable.

Obtain the fully qualified domain name of the SQL Server instance:

- Launch Windows Explorer.
- Right-click on "Computer" or "This PC" (Varies by OS level), click "Properties".
- Note the value shown for "Full computer name".

*** **Note:** For a cluster, this value must be obtained from the Failover Cluster Manager. ***

Obtain the TCP port that is supporting the SQL Server instance:

- Click Start >> Type "SQL Server 2016 Configuration Manager" >> From the search results, click "SQL Server 2016 Configuration Manager".
- From the tree on the left, expand "SQL Server Network Configuration".
- Click "Protocols for <Instance Name>" where <Instance Name> is the name of the instance (MSSQLSERVER is the default name).
- In the right pane, right-click on "TCP/IP" and choose "Properties".
- In the window that opens, click the "IP Addresses" tab.
- Note the TCP port configured for the instance.

Obtain the service account that is running the SQL Server service:

- Click "Start".
- Type "SQL Server 2016 Configuration Manager".
- From the search results, click "SQL Server 2016 Configuration Manager".
- From the tree on the left, select "SQL Server Services".
- Note the account listed in the "Log On As" column for the SQL Server instance being reviewed.

Launch a command-line or PowerShell window.

Enter the following command where *<Service Account>* is the identity of the service account.

```
setspn -L <Service Account>
```

Example: `setspn -L CONTOSO\sql2016svc`

Review the Registered Service Principal Names returned.

If the listing does not contain the following supported service principal names (SPN) formats, this is a finding.

- Named instance

```
MSSQLSvc/<FQDN>:[<port> | <instancename>], where:
MSSQLSvc is the service that is being registered.
<FQDN> is the fully qualified domain name of the server.
<port> is the TCP port number.
<instancename> is the name of the SQL Server instance.
```

- Default instance

```
MSSQLSvc/<FQDN>:<port> | MSSQLSvc/<FQDN>, where:
MSSQLSvc is the service that is being registered.
<FQDN> is the fully qualified domain name of the server.
<port> is the TCP port number.
```

If the MSSQLSvc service is registered for any fully qualified domain names that do not match the current server, this may indicate the service account is shared across SQL Server instances. Review server documentation, if the sharing of service accounts across instances is not documented and authorized, this is a finding.

Remediation:

Ensure Service Principal Names (SPNs) are properly registered for the SQL Server instance.

Utilize the Microsoft Kerberos Configuration Manager to review Kerberos configuration issues for a given SQL Server instance.

https://www.microsoft.com/en-us/download/details.aspx?id=39046

Alternatively, SPNs for SQL Server can be manually registered.

For other connections that support Kerberos the SPN is registered in the format MSSQLSvc/*<FQDN>/<instancename>* for a named instance. The format for registering the default instance is MSSQLSvc/*<FQDN>*.

Using an account with permissions to register SPNs, issue the following commands from a command-prompt:

```
setspn -S MSSQLSvc/<Fully Qualified Domain Name> <Service Account>
setspn -S MSSQLSvc/<Fully Qualified Domain Name>:<TCP Port> <Service Account>
```

For a named instance, use:

```
setspn -S MSSQLSvc/<FQDN>:<instancename> <Service Account>
setspn -S MSSQLSvc/<FQDN>:<TCP Port> <Service Account>
```

Restart the SQL Server instance.

More information regarding this process is available at:

https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/register-a-service-principal-name-for-kerberos-connections#Manual

Additional Information:

CCI-000015

Support the management of system accounts using organization-defined automated mechanisms.

- NIST SP 800-53::AC-2 (1)
- NIST SP 800-53A::AC-2 (1).1
- NIST SP 800-53 Revision 4::AC-2 (1)
- NIST SP 800-53 Revision 5::AC-2 (1)

1.4 SQL6-D0-004000 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must protect against a user falsely repudiating by ensuring all accounts are individual, unique, and not shared.

```
GROUP ID: V-213933
RULE ID: SV-213933r960864
```

Rationale:

Non-repudiation of actions taken is required in order to maintain data integrity. Examples of particular actions taken by individuals include creating information, sending a message, approving information (e.g., indicating concurrence or signing a contract), and receiving a message.

Non-repudiation protects against later claims by a user of not having created, modified, or deleted a particular data item or collection of data in the database.

In designing a database, the organization must define the types of data and the user actions that must be protected from repudiation. The implementation must then include building audit features into the application data tables and configuring SQL Server's audit tools to capture the necessary audit trail. Design and implementation also must ensure that applications pass individual user identification to SQL Server, even where the application connects to SQL Server with a standard, shared account.

Audit:

Obtain the list of authorized SQL Server accounts in the system documentation.

Determine if any accounts are shared. A shared account is defined as a username and password that are used by multiple individuals to log into SQL Server. An example of a shared account is the SQL Server installation account. Windows Groups are not shared accounts as the group itself does not have a password.

If accounts are determined to be shared, determine if individuals are first individually authenticated.

If individuals are not individually authenticated before using the shared account (e.g., by the operating system or possibly by an application making calls to the database), this is a finding.

The key is individual accountability. If this can be traced, this is not a finding.

If accounts are determined to be shared, determine if they are directly accessible to end users. If so, this is a finding.

Review contents of audit logs, traces and data tables to confirm that the identity of the individual user performing the action is captured.

If shared identifiers are found, and not accompanied by individual identifiers, this is a finding.

Note: Privileged installation accounts may be required to be accessed by the DBA or other administrators for system maintenance. In these cases, each use of the account must be logged in some manner to assign accountability for any actions taken during the use of the account.

Remediation:

Remove user-accessible shared accounts and use individual user IDs.

Build/configure applications to ensure successful individual authentication prior to shared account access.

Ensure each user's identity is received and used in audit data in all relevant circumstances.

Design, develop, and implement a method to log use of any account to which more than one person has access. Restrict interactive access to shared accounts to the fewest persons possible.

Additional Information:

CCI-000166

Provide irrefutable evidence that an individual (or process acting on behalf of an individual) falsely denying having performed organization-defined actions to be covered by non-repudiation.

- NIST SP 800-53::AU-10
- NIST SP 800-53A::AU-10.1
- NIST SP 800-53 Revision 4::AU-10
- NIST SP 800-53 Revision 5::AU-10

1.5 SQL6-D0-003900 (Manual)

Profile Applicability:

- SEVERITY: CAT I

Description:

SQL Server must enforce approved authorizations for logical access to information and system resources in accordance with applicable access control policies.

```
GROUP ID: V-213932
RULE ID: SV-213932r960792
```

Rationale:

Authentication with a DoD-approved PKI certificate does not necessarily imply authorization to access SQL Server. To mitigate the risk of unauthorized access to sensitive information by entities that have been issued certificates by DoD-approved PKIs, all DoD systems, including databases, must be properly configured to implement access control policies.

Successful authentication must not automatically give an entity access to an asset or security boundary. Authorization procedures and controls must be implemented to ensure each authenticated entity also has a validated and current authorization. Authorization is the process of determining whether an entity, once authenticated, is permitted to access a specific asset. Information systems use access control policies and enforcement mechanisms to implement this requirement.

Access control policies include identity-based policies, role-based policies, and attribute-based policies. Access enforcement mechanisms include access control lists, access control matrices, and cryptography. These policies and mechanisms must be employed by the application to control access between users (or processes acting on behalf of users) and objects (e.g., devices, files, records, processes, programs, and domains) in the information system.

This requirement is applicable to access control enforcement applications, a category that includes database management systems. If SQL Server does not follow applicable policy when approving access, it may be in conflict with networks or other applications in the information system. This may result in users either gaining or being denied access inappropriately and in conflict with applicable policy.

Audit:

Review the system documentation to determine the required levels of protection for DBMS server securables, by type of login.

Review the permissions actually in place on the server.

If the actual permissions do not match the documented requirements, this is a finding.

Use the supplemental file "Instance permissions assignments to logins and `roles.sql`."

Remediation:

Use `GRANT`, `REVOKE`, `DENY`, `ALTER SERVER ROLE … ADD MEMBER …` and/or `ALTER SERVER ROLE …. DROP MEMBER` statements to add and remove permissions on server-level securables, bringing them into line with the documented requirements.

Additional Information:

CCI-000213

Enforce approved authorizations for logical access to information and system resources in accordance with applicable access control policies.

- NIST SP 800-53::AC-3
- NIST SP 800-53A::AC-3.1
- NIST SP 800-53 Revision 4::AC-3
- NIST SP 800-53 Revision 5::AC-3

1.6 SQL6-D0-004100 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must protect against a user falsely repudiating by ensuring the NT AUTHORITY SYSTEM account is not used for administration.

```
GROUP ID: V-213934
RULE ID: SV-213934r960864
```

Rationale:

Non-repudiation of actions taken is required in order to maintain data integrity. Examples of particular actions taken by individuals include creating information, sending a message, approving information (e.g., indicating concurrence or signing a contract), and receiving a message.

Non-repudiation protects against later claims by a user of not having created, modified, or deleted a particular data item or collection of data in the database.

In designing a database, the organization must define the types of data and the user actions that must be protected from repudiation. The implementation must then include building audit features into the application data tables and configuring the DBMS's audit tools to capture the necessary audit trail. Design and implementation also must ensure that applications pass individual user identification to the DBMS, even where the application connects to the DBMS with a standard, shared account.

Any user with enough access to the server can execute a task that will be run as `NT AUTHORITY\SYSTEM` either using task scheduler or other tools. At this point, `NT AUTHORITY\SYSTEM` essentially becomes a shared account because the operating system and SQL Server are unable to determine who created the process.

Prior to SQL Server 2012, `NT AUTHORITY\SYSTEM` was a member of the sysadmin role by default. This allowed jobs/tasks to be executed in SQL Server without the approval or knowledge of the DBA because it looked like operating system activity.

Audit:

Execute the following queries. The first query checks for Clustering and Availability Groups being provisioned in the Database Engine. The second query lists permissions granted to the Local System account.

```
SELECT
    SERVERPROPERTY('IsClustered') AS [IsClustered],
    SERVERPROPERTY('IsHadrEnabled') AS [IsHadrEnabled]
EXECUTE AS LOGIN = 'NT AUTHORITY\SYSTEM'
SELECT * FROM fn_my_permissions(NULL, 'server')
REVERT
```

```
GO
```

If `IsClustered` returns `1`, `IsHadrEnabled` returns `0`, and any permissions have been granted to the Local System account beyond `CONNECT SQL`, `VIEW SERVER STATE`, and `VIEW ANY DATABASE`, this is a finding.

If `IsHadrEnabled` returns `1` and any permissions have been granted to the Local System account beyond `CONNECT SQL`, `CREATE AVAILABILITY GROUP`, `ALTER ANY AVAILABILITY GROUP`, `VIEW SERVER STATE`, and `VIEW ANY DATABASE`, this is a finding.

If both `IsClustered` and `IsHadrEnabled` return `0` and any permissions have been granted to the Local System account beyond `CONNECT SQL` and `VIEW ANY DATABASE`, this is a finding.

Remediation:

Remove permissions that were identified as not allowed in the check content.

```
USE Master;
REVOKE <Permission> TO [NT AUTHORITY\SYSTEM]
GO
```

To grant permissions to services or applications, utilize the Service SID of the service or a domain service account.

Additional Information:

CCI-000166

Provide irrefutable evidence that an individual (or process acting on behalf of an individual) falsely denying having performed organization-defined actions to be covered by non-repudiation.

- NIST SP 800-53::AU-10
- NIST SP 800-53A::AU-10.1
- NIST SP 800-53 Revision 4::AU-10
- NIST SP 800-53 Revision 5::AU-10

1.7 SQL6-D0-004200 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must protect against a user falsely repudiating by ensuring only clearly unique Active Directory user accounts can connect to the instance.

```
GROUP ID: V-213935
RULE ID: SV-213935r960864
```

Rationale:

Non-repudiation of actions taken is required in order to maintain data integrity. Examples of particular actions taken by individuals include creating information, sending a message, approving information (e.g., indicating concurrence or signing a contract), and receiving a message.

Non-repudiation protects against later claims by a user of not having created, modified, or deleted a particular data item or collection of data in the database.

In designing a database, the organization must define the types of data and the user actions that must be protected from repudiation. The implementation must then include building audit features into the application data tables and configuring the DBMS's audit tools to capture the necessary audit trail. Design and implementation also must ensure that applications pass individual user identification to the DBMS, even where the application connects to the DBMS with a standard, shared account.

If the computer account of a remote computer is granted access to SQL Server, any service or scheduled task running as `NT AUTHORITY\SYSTEM` or `NT AUTHORITY\NETWORK SERVICE` can log into the instance and perform actions. These actions cannot be traced back to a specific user or process.

Audit:

Execute the following query:

```
SELECT name
FROM sys.server_principals
WHERE type in ('U','G')
AND name LIKE '%$'
```

If no logins are returned, this is not a finding.

If logins are returned, determine whether each login is a computer account.

- Launch PowerShell.
- Execute the following code:

Note: *<name>* represents the username portion of the login. For example, if the login is CONTOSO\user1$, the username is user1.

```
([ADSISearcher]"(&(ObjectCategory=Computer)(Name=<name>))").FindAll()
```

If no account information is returned, this is not a finding.

If account information is returned, this is a finding.

Remediation:

Remove all logins that were returned in the check content.

Additional Information:

CCI-000166

Provide irrefutable evidence that an individual (or process acting on behalf of an individual) falsely denying having performed organization-defined actions to be covered by non-repudiation.

- NIST SP 800-53::AU-10
- NIST SP 800-53A::AU-10.1
- NIST SP 800-53 Revision 4::AU-10
- NIST SP 800-53 Revision 5::AU-10

1.8 SQL6-D0-004300 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must be configured to generate audit records for DoD-defined auditable events within all DBMS/database components.

```
GROUP ID: V-213936
RULE ID: SV-213936r960879
```

Rationale:

Without the capability to generate audit records, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit records can be generated from various components within SQL Server (e.g., process, module). Certain specific application functionalities may be audited as well. The list of audited events is the set of events for which audits are to be generated. This set of events is typically a subset of the list of all events for which the system is capable of generating audit records.

DoD has defined the list of events for which SQL Server will provide an audit record generation capability as the following:

(i) Successful and unsuccessful attempts to access, modify, or delete privileges, security objects, security levels, or categories of information (e.g., classification levels);

(ii) Access actions, such as successful and unsuccessful logon attempts, privileged activities, or other system-level access, starting and ending time for user access to the system, concurrent logons from different workstations, successful and unsuccessful accesses to objects, all program initiations, and all direct access to the information system; and

(iii) All account creation, modification, disabling, and termination actions.

Organizations may define additional events requiring continuous or ad hoc auditing.

Audit:

Review the server documentation to determine if any additional events are required to be audited. If no additional events are required, this is not a finding.

Execute the following query to get all of the installed audits:

```
SELECT name AS 'Audit Name',
status_desc AS 'Audit Status',
audit_file_path AS 'Current Audit File'
```

```
FROM sys.dm_server_audit_status
```

All currently defined audits for the SQL server instance will be listed. If no audits are returned, this is a finding.

To view the actions being audited by the audits, execute the following query:

```
SELECT a.name AS 'AuditName',
s.name AS 'SpecName',
d.audit_action_name AS 'ActionName',
d.audited_result AS 'Result'
FROM sys.server_audit_specifications s
JOIN sys.server_audits a ON s.audit_guid = a.audit_guid
JOIN sys.server_audit_specification_details d ON s.server_specification_id =
d.server_specification_id
WHERE a.is_state_enabled = 1
```

Compare the documentation to the list of generated audit events. If there are any missing events, this is a finding.

Remediation:

Add all required audit events to the STIG Compliant audit specification server documentation.

Additional Information:

CCI-000169

Provide audit record generation capability for the event types the system is capable of auditing as defined in AU-2 a on organization-defined information system components.

- NIST SP 800-53::AU-12 a
- NIST SP 800-53A::AU-12.1 (ii)
- NIST SP 800-53 Revision 4::AU-12 a
- NIST SP 800-53 Revision 5::AU-12 a

1.9 SQL6-D0-004400 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must allow only the ISSM (or individuals or roles appointed by the ISSM) to select which auditable events are to be audited.

```
GROUP ID: V-213937
RULE ID: SV-213937r960882
```

Rationale:

Without the capability to restrict which roles and individuals can select which events are audited, unauthorized personnel may be able to prevent or interfere with the auditing of critical events.

Suppression of auditing could permit an adversary to evade detection.

Misconfigured audits can degrade the system's performance by overwhelming the audit log. Misconfigured audits may also make it more difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

Audit:

Obtain the list of approved audit maintainers from the system documentation.

Review the server roles and individual logins that have the following role memberships, all of which enable the ability to create and maintain audit definitions.

```
sysadmin
dbcreator
```

Review the server roles and individual logins that have the following permissions, all of which enable the ability to create and maintain audit definitions.

```
ALTER ANY SERVER AUDIT
CONTROL SERVER
ALTER ANY DATABASE
CREATE ANY DATABASE
```

Use the following query to determine the roles and logins that have the listed permissions:

```
SELECT-- DISTINCT
    CASE
        WHEN SP.class_desc IS NOT NULL THEN
            CASE
                WHEN SP.class_desc = 'SERVER' AND S.is_linked = 0 THEN
'SERVER'
                WHEN SP.class_desc = 'SERVER' AND S.is_linked = 1 THEN
'SERVER (linked)'
```

```
                ELSE SP.class_desc
            END
        WHEN E.name IS NOT NULL THEN 'ENDPOINT'
        WHEN S.name IS NOT NULL AND S.is_linked = 0 THEN 'SERVER'
        WHEN S.name IS NOT NULL AND S.is_linked = 1 THEN 'SERVER (linked)'
        WHEN P.name IS NOT NULL THEN 'SERVER_PRINCIPAL'
        ELSE '???'
    END                     AS [Securable Class],
    CASE
        WHEN E.name IS NOT NULL THEN E.name
        WHEN S.name IS NOT NULL THEN S.name
        WHEN P.name IS NOT NULL THEN P.name
        ELSE '???'
    END                     AS [Securable],
    P1.name                 AS [Grantee],
    P1.type_desc            AS [Grantee Type],
    sp.permission_name      AS [Permission],
    sp.state_desc           AS [State],
    P2.name                 AS [Grantor],
    P2.type_desc            AS [Grantor Type],
R.name    AS [Role Name]
FROM
    sys.server_permissions SP
    INNER JOIN sys.server_principals P1
        ON P1.principal_id = SP.grantee_principal_id
    INNER JOIN sys.server_principals P2
        ON P2.principal_id = SP.grantor_principal_id

    FULL OUTER JOIN sys.servers S
        ON  SP.class_desc = 'SERVER'
        AND S.server_id = SP.major_id

    FULL OUTER JOIN sys.endpoints E
        ON  SP.class_desc = 'ENDPOINT'
        AND E.endpoint_id = SP.major_id

    FULL OUTER JOIN sys.server_principals P
        ON  SP.class_desc = 'SERVER_PRINCIPAL'
        AND P.principal_id = SP.major_id

FULL OUTER JOIN sys.server_role_members SRM
ON P.principal_id = SRM.member_principal_id

LEFT OUTER JOIN sys.server_principals R
ON SRM.role_principal_id = R.principal_id
WHERE sp.permission_name IN ('ALTER ANY SERVER AUDIT','CONTROL SERVER','ALTER
ANY DATABASE','CREATE ANY DATABASE')
OR R.name IN ('sysadmin','dbcreator')
```

If any of the logins, roles, or role memberships returned have permissions that are not documented, or the documented audit maintainers do not have permissions, this is a finding.

Remediation:

Create a server role specifically for audit maintainers and give it permission to maintain audits without granting it unnecessary permissions (the role name used here is an example; other names may be used):

```
CREATE SERVER ROLE SERVER_AUDIT_MAINTAINERS;
GO
GRANT ALTER ANY SERVER AUDIT TO SERVER_AUDIT_MAINTAINERS;
GO
```

Use `REVOKE` and/or `DENY` and/or `ALTER SERVER ROLE ... DROP MEMBER ...` statements to remove the `ALTER ANY SERVER AUDIT` permission from all logins. Then, for each authorized login, run the statement:

```
ALTER SERVER ROLE SERVER_AUDIT_MAINTAINERS ADD MEMBER;
GO
```

Use `REVOKE` and/or `DENY` and/or `ALTER SERVER ROLE ... DROP MEMBER ...` statements to remove `CONTROL SERVER`, `ALTER ANY DATABASE` and `CREATE ANY DATABASE` permissions from logins that do not need them.

Additional Information:

CCI-000171

Allow organization-defined personnel or roles to select the event types that are to be logged by specific components of the system.

- NIST SP 800-53::AU-12 b
- NIST SP 800-53A::AU-12.1 (iii)
- NIST SP 800-53 Revision 4::AU-12 b
- NIST SP 800-53 Revision 5::AU-12 b

1.10 SQL6-D0-004700 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must initiate session auditing upon startup.

```
GROUP ID: V-213940
RULE ID: SV-213940r960888
```

Rationale:

Session auditing is for use when a user's activities are under investigation. To be sure of capturing all activity during those periods when session auditing is in use, it needs to be in operation for the whole time SQL Server is running.

Audit:

When Audits are enabled, they start up when the instance starts.

https://msdn.microsoft.com/en-us/library/cc280386.aspx#Anchor_2

Check if an audit is configured and enabled.

Execute the following query:

```
SELECT name AS 'Audit Name',
status_desc AS 'Audit Status',
audit_file_path AS 'Current Audit File'
FROM sys.dm_server_audit_status
WHERE status_desc = 'STARTED'
```

All currently defined audits for the SQL server instance will be listed. If no audits are returned, this is a finding.

Remediation:

Configure the SQL Audit(s) to automatically start during system start-up.

```
ALTER SERVER AUDIT [<Server Audit Name>] WITH STATE = ON
```

Execute the following query:

```
SELECT name AS 'Audit Name',
  status_desc AS 'Audit Status',
  audit_file_path AS 'Current Audit File'
FROM sys.dm_server_audit_status
WHERE status_desc = 'STARTED'
```

Ensure the SQL STIG Audit is configured to initiate session auditing upon startup.

Additional Information:

CCI-001464

Initiates session audits automatically at system start-up.

- NIST SP 800-53::AU-14 (1)
- NIST SP 800-53A::AU-14 (1).1
- NIST SP 800-53 Revision 4::AU-14 (1)
- NIST SP 800-53 Revision 5::AU-14 (1)

1.11 SQL6-D0-004600 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must generate audit records when successful/unsuccessful attempts to retrieve privileges/permissions occur.

```
GROUP ID: V-213939
RULE ID: SV-213939r960885
```

Rationale:

Under some circumstances, it may be useful to monitor who/what is reading privilege/permission/role information. Therefore, monitoring must be possible. DBMSs typically make such information available through views or functions.

This requirement addresses explicit requests for privilege/permission/role membership information. It does not refer to the implicit retrieval of privileges/permissions/role memberships that SQL Server continually performs to determine if any and every action on the database is permitted.

To aid in diagnosis, it is necessary to keep track of failed attempts in addition to the successful ones.

Satisfies: SRG-APP-000091-DB-000066

Audit:

Review the system documentation to determine if SQL Server is required to audit the retrieval of privilege/permission/role membership information.

If SQL Server is not required to audit the retrieval of privilege/permission/role membership information, this is not a finding.

If the documentation does not exist, this is a finding.

Determine if an audit is configured and started by executing the following query. If no records are returned, this is a finding.

```
SELECT name AS 'Audit Name',
status_desc AS 'Audit Status',
audit_file_path AS 'Current Audit File'
FROM sys.dm_server_audit_status
```

If the auditing the retrieval of privilege/permission/role membership information is required, execute the following query to verify the SCHEMA_OBJECT_ACCESS_GROUP is included in the server audit specification.

```
SELECT a.name AS 'AuditName',
```

```
s.name AS 'SpecName',
d.audit_action_name AS 'ActionName',
d.audited_result AS 'Result'
FROM sys.server_audit_specifications s
JOIN sys.server_audits a ON s.audit_guid = a.audit_guid
JOIN sys.server_audit_specification_details d ON s.server_specification_id =
d.server_specification_id
WHERE a.is_state_enabled = 1 AND d.audit_action_name =
'SCHEMA_OBJECT_ACCESS_GROUP'
```

If the SCHEMA_OBJECT_ACCESS_GROUP is not returned in an active audit, this is a finding.

Remediation:

Deploy an audit to audit the retrieval of privilege/permission/role membership information. See the supplemental file SQL 2016 Audit.sql

Additional Information:

CCI-000172

Generate audit records for the event types defined in AU-2 c that include the audit record content defined in AU-3.

- NIST SP 800-53::AU-12 c
- NIST SP 800-53A::AU-12.1 (iv)
- NIST SP 800-53 Revision 4::AU-12 c
- NIST SP 800-53 Revision 5::AU-12 c

1.12 SQL6-D0-005500 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must include additional, more detailed, organization-defined information in the audit records for audit events identified by type, location, or subject.

```
GROUP ID: V-213941
RULE ID: SV-213941r960909
```

Rationale:

Information system auditing capability is critical for accurate forensic analysis. Reconstruction of harmful events or forensic analysis is not possible if audit records do not contain enough information. To support analysis, some types of events will need information to be logged that exceeds the basic requirements of event type, time stamps, location, source, outcome, and user identity. If additional information is not available, it could negatively impact forensic investigations into user actions or other malicious events.

The organization must determine what additional information is required for complete analysis of the audited events. The additional information required is dependent on the type of information (e.g., sensitivity of the data and the environment within which it resides). At a minimum, the organization must employ either full-text recording of privileged commands or the individual identities of users of shared accounts, or both. The organization must maintain audit trails in sufficient detail to reconstruct events to determine the cause and impact of compromise.

Examples of detailed information the organization may require in audit records are full-text recording of privileged commands or the individual identities of shared account users.

Audit:

If a SQL Server Audit is not in use for audit purposes, this is a finding unless a third-party product is being used that can perform detailed auditing for SQL Server.

Review system documentation to determine whether SQL Server is required to audit any events, and any fields, in addition to those in the standard audit.

If there are none specified, this is not a finding.

If SQL Server Audit is in use, compare the audit specification(s) with the documented requirements.

If any such requirement is not satisfied by the audit specification(s) (or by supplemental, locally-deployed mechanisms), this is a finding.

Remediation:

Design and deploy an Audit that captures all auditable events and data items. In the event a third-party tool is used for auditing it must contain all the required information including but not limited to events, type, location, subject, date and time and by whom the change occurred.

Implement additional custom audits to capture the additional organizational required information.

Additional Information:

CCI-000135

Generate audit records containing the organization-defined additional information that is to be included in the audit records.

- NIST SP 800-53::AU-3 (1)
- NIST SP 800-53A::AU-3 (1).1 (ii)
- NIST SP 800-53 Revision 4::AU-3 (1)
- NIST SP 800-53 Revision 5::AU-3 (1)

1.13 SQL6-D0-005600 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must by default shut down upon audit failure, to include the unavailability of space for more audit log records; or must be configurable to shut down upon audit failure.

```
GROUP ID: V-213942
RULE ID: SV-213942r1043188
```

Rationale:

It is critical that when SQL Server is at risk of failing to process audit logs as required, it takes action to mitigate the failure. Audit processing failures include: software/hardware errors; failures in the audit capturing mechanisms; and audit storage capacity being reached or exceeded. Responses to audit failure depend upon the nature of the failure mode.

When the need for system availability does not outweigh the need for a complete audit trail, SQL Server should shut down immediately, rolling back all in-flight transactions.

Systems where audit trail completeness is paramount will most likely be at a lower MAC level than MAC I; the final determination is the prerogative of the application owner, subject to Authorizing Official concurrence. In any case, sufficient auditing resources must be allocated to avoid a shutdown in all but the most extreme situations.

Audit:

If the system documentation indicates that availability takes precedence over audit trail completeness, this is not applicable (NA).

If SQL Server Audit is in use, review the defined server audits by running the statement:

```
SELECT * FROM sys.server_audits;
```

By observing the [name] and [is_state_enabled] columns, identify the row or rows in use.

If the `[on_failure_desc]` is `SHUTDOWN SERVER INSTANCE` on this/these row(s), this is not a finding. Otherwise, this is a finding.

Remediation:

If SQL Server Audit is in use, configure SQL Server Audit to shut SQL Server down upon audit failure, to include running out of space for audit logs.

Run this T-SQL script for each identified audit:

```
ALTER SERVER AUDIT [AuditNameHere] WITH (STATE = OFF);
GO
ALTER SERVER AUDIT [AuditNameHere] WITH (ON_FAILURE = SHUTDOWN);
GO
ALTER SERVER AUDIT [AuditNameHere] WITH (STATE = ON);
GO
```

Additional Information:

CCI-000140

Take organization-defined actions upon audit failure include, shutting down the system, overwriting oldest audit records, and stopping the generation of audit records.

- NIST SP 800-53::AU-5 b
- NIST SP 800-53A::AU-5.1 (iv)
- NIST SP 800-53 Revision 4::AU-5 b
- NIST SP 800-53 Revision 5::AU-5 b

1.14 SQL6-D0-005700 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must be configurable to overwrite audit log records, oldest first (First-In-First-Out - FIFO), in the event of unavailability of space for more audit log records.

```
GROUP ID: V-213943
RULE ID: SV-213943r1043188
```

Rationale:

It is critical that when SQL Server is at risk of failing to process audit logs as required, it take action to mitigate the failure. Audit processing failures include; software/hardware errors; failures in the audit capturing mechanisms; and audit storage capacity being reached or exceeded. Responses to audit failure depend upon the nature of the failure mode.

When availability is an overriding concern, approved actions in response to an audit failure are as follows:

(i) If the failure was caused by the lack of audit record storage capacity, SQL Server must continue generating audit records, if possible (automatically restarting the audit service if necessary), overwriting the oldest audit records in a first-in-first-out manner.

(ii) If audit records are sent to a centralized collection server and communication with this server is lost or the server fails, SQL Server must queue audit records locally until communication is restored or until the audit records are retrieved manually. Upon restoration of the connection to the centralized collection server, action should be taken to synchronize the local audit data with the collection server.

Systems where availability is paramount will most likely be MAC I; the final determination is the prerogative of the application owner, subject to Authorizing Official concurrence. In any case, sufficient auditing resources must be allocated to avoid audit data loss in all but the most extreme situations.

Audit:

If the system documentation indicates that availability does not take precedence over audit trail completeness, this is not applicable (NA).

Execute the following query:

```
SELECT a.name 'audit_name',
    a.type_desc 'storage_type',
    f.max_rollover_files
FROM sys.server_audits a
LEFT JOIN sys.server_file_audits f ON a.audit_id = f.audit_id
```

```
WHERE a.is_state_enabled = 1
```

If no records are returned, this is a finding.

If the `storage_type` is `APPLICATION LOG` or `SECURITY LOG`, this is not a finding.

If the `storage_type` is `FILE` and `max_rollover_files` is greater than zero, this is not a finding. Otherwise, this is a finding.

Remediation:

If SQL Server Audit is in use, configure SQL Server Audit to continue to generate audit records, overwriting the oldest existing records, in the case of an auditing failure.

Run this T-SQL script for each identified audit:

```
ALTER SERVER AUDIT [AuditName] WITH (STATE = OFF);
GO
ALTER SERVER AUDIT [AuditName] to file (max_rollover_files = IntegerValue);
GO
ALTER SERVER AUDIT [AuditName] WITH (STATE = ON);
GO
```

Additional Information:

CCI-000140

Take organization-defined actions upon audit failure include, shutting down the system, overwriting oldest audit records, and stopping the generation of audit records.

- NIST SP 800-53::AU-5 b
- NIST SP 800-53A::AU-5.1 (iv)
- NIST SP 800-53 Revision 4::AU-5 b
- NIST SP 800-53 Revision 5::AU-5 b

1.15 SQL6-D0-005900 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

The audit information produced by SQL Server must be protected from unauthorized access, modification, and deletion.

```
GROUP ID: V-213944
RULE ID: SV-213944r960930
```

Rationale:

If audit data were to become compromised, then competent forensic analysis and discovery of the true source of potentially malicious system activity is difficult, if not impossible, to achieve. In addition, access to audit records provides information an attacker could potentially use to his or her advantage.

To ensure the veracity of audit data, the information system and/or the application must protect audit information from any and all unauthorized access. This includes read, write, copy, etc.

This requirement can be achieved through multiple methods which will depend upon system architecture and design. Some commonly employed methods include ensuring log files enjoy the proper file system permissions utilizing file system protections and limiting log data location.

Additionally, applications with user interfaces to audit records should not allow for the unfettered manipulation of or access to those records via the application. If the application provides access to the audit data, the application becomes accountable for ensuring that audit information is protected from unauthorized access. SQL Server is an application that is able to view and manipulate audit file data.

Audit information includes all information (e.g., audit records, audit settings, and audit reports) needed to successfully audit information system activity.

Satisfies: SRG-APP-000118-DB-000059, SRG-APP-000119-DB-000060, SRG-APP-000120-DB-000061

Audit:

If the database is setup to write audit logs using APPLICATION or SECURITY event logs rather than writing to a file, this is N/A.

Obtain the SQL Server audit file location(s) by running the following SQL script:

```
SELECT log_file_path AS "Audit Path"
FROM sys.server_file_audits
```

For each audit, the path column will give the location of the file.

Verify that all audit files have the correct permissions by doing the following for each audit file: Navigate to audit folder location(s) using a command prompt or Windows Explorer.

Right-click the file/folder and click "Properties". On the "Security" tab, verify that at most the following permissions are applied:

```
Administrator (read)
Users (none)
Audit Administrator (Full Control)
Auditors group (Read)
SQL Server Service SID OR Service Account (Full Control)
SQL Server SQL Agent Service SID OR Service Account, if SQL Server Agent is
in use. (Read, Execute, Write)
```

If any less restrictive permissions are present (and not specifically justified and approved), this is a finding.

Remediation:

Modify audit file permissions to meet the requirement to protect against unauthorized access.

Application event log and security log permissions are covered in the Windows Server STIGs. Be sure to reference these depending on the OS in use.

Navigate to audit folder location(s) using a command prompt or Windows Explorer. Right-click the file and click "Properties".

On the Security tab, modify the security permissions to:

```
Administrator (read)
Users (none)
Audit Administrator(Full Control)
Auditors group (Read)
SQL Server Service SID OR Service Account (Full Control) [Notes 1, 2]
SQL Server SQL Agent Service SID OR Service Account, if SQL Server Agent is
in use. (Read, Execute, Write) [Notes 1, 2]
```

Note 1: It is highly advisable to use a separate account for each service. When installing SQL Server in single-server mode, you can opt to have these provisioned for you. These automatically-generated accounts are referred to as virtual accounts. Each virtual account has an equivalent Service SID, with the same name. The installer also creates an equivalent SQL Server login, also with the same name. Applying folder and file permissions to Service SIDs, rather than to domain accounts or local computer accounts, provides tighter control, because these permissions are available only to the specific service when it is running, and not in any other context. (However, when using failover clustering, a domain account must be specified at installation, rather than a virtual account.) For more on this topic, see http://msdn.microsoft.com/en-us/library/ms143504(v=sql.130).aspx.

Note 2: Tips for adding a service SID/virtual account to a folder's permission list.

1. In Windows Explorer, right-click the folder and select "Properties".
2. Select the "Security" tab.
3. Click "Edit".
4. Click "Add".
5. Click "Locations".
6. Select the computer name.
7. Search for the name.
    - 7.a) SQL Server Service
        - 7.a.i) Type `NT SERVICE\MSSQL` and click "Check Names". (What you have just typed in is the first 16 characters of the name. At least one character must follow `NT SERVICE\`; you will be presented with a list of all matches. If you have typed in the full, correct name, step 7.a.ii is bypassed.)
        - 7.a.ii) Select the "MSSQL$" user and click "OK".
    - 7.b) SQL Agent Service
        - 7.b.i) Type `NT SERVICE\SQL` and click "Check Names".
        - 7.b.ii) Select the `SQLAgent$` user and click "OK".
8. Click "OK".
9. Permission like a normal user from here.

Additional Information:

CCI-000162

Protect audit information from unauthorized access.

- NIST SP 800-53::AU-9
- NIST SP 800-53A::AU-9.1
- NIST SP 800-53 Revision 4::AU-9
- NIST SP 800-53 Revision 5::AU-9 a

CCI-000163

Protect audit information from unauthorized modification.

- NIST SP 800-53::AU-9
- NIST SP 800-53A::AU-9.1
- NIST SP 800-53 Revision 4::AU-9
- NIST SP 800-53 Revision 5::AU-9 a

CCI-000164

Protect audit information from unauthorized deletion.

- NIST SP 800-53::AU-9
- NIST SP 800-53A::AU-9.1

- NIST SP 800-53 Revision 4::AU-9
- NIST SP 800-53 Revision 5::AU-9 a

1.16 SQL6-D0-006300 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must protect its audit configuration from authorized and unauthorized access and modification.

```
GROUP ID: V-213948
RULE ID: SV-213948r960942
```

Rationale:

Protecting audit data also includes identifying and protecting the tools used to view and manipulate log data. Therefore, protecting audit tools is necessary to prevent unauthorized operation on audit data.

Applications providing tools to interface with audit data will leverage user permissions and roles identifying the user accessing the tools and the corresponding rights the user enjoys in order make access decisions regarding the modification of audit tools. SQL Server is an application that does provide access to audit data.

Audit tools include, but are not limited to, vendor-provided and open source audit tools needed to successfully view and manipulate audit information system activity and records. Audit tools include custom queries and report generators.

If an attacker were to gain access to audit tools, they could analyze audit logs for system weaknesses or weaknesses in the auditing itself. An attacker could also manipulate logs to hide evidence of malicious activity.

Satisfies: SRG-APP-000121-DB-000202, SRG-APP-000123-DB-000204

Audit:

Check the server documentation for a list of approved users with access to SQL Server Audits.

To create, alter, or drop a server audit, principals require the ALTER ANY SERVER AUDIT or the `CONTROL SERVER` permission.

Review the SQL Server permissions granted to principals. Look for permissions `ALTER ANY SERVER AUDIT`, `ALTER ANY DATABASE AUDIT`, `CONTROL SERVER`:

```
SELECT login.name, perm.permission_name, perm.state_desc
FROM sys.server_permissions perm
JOIN sys.server_principals login
ON perm.grantee_principal_id = login.principal_id
WHERE permission_name in ('ALTER ANY DATABASE AUDIT', 'ALTER ANY SERVER
AUDIT', 'CONTROL SERVER')
```

```
and login.name not like '##MS_%';
```

If unauthorized accounts have these privileges, this is a finding.

Remediation:

Remove audit-related permissions from individuals and roles not authorized to have them.

```
USE master;
DENY [ALTER ANY SERVER AUDIT] TO [User];
GO
```

Additional Information:

CCI-001494

Protect audit tools from unauthorized modification.

- NIST SP 800-53::AU-9
- NIST SP 800-53A::AU-9.1
- NIST SP 800-53 Revision 4::AU-9
- NIST SP 800-53 Revision 5::AU-9 a

1.17 SQL6-D0-006500 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must limit privileges to change software modules and links to software external to SQL Server.

```
GROUP ID: V-213950
RULE ID: SV-213950r960960
```

Rationale:

If the system were to allow any user to make changes to software libraries, then those changes might be implemented without undergoing the appropriate testing and approvals that are part of a robust change management process.

Accordingly, only qualified and authorized individuals must be allowed to obtain access to information system components for purposes of initiating changes, including upgrades and modifications.

Unmanaged changes that occur to the database software libraries or configuration can lead to unauthorized or compromised installations.

Audit:

Review Server documentation to determine the authorized owner and users or groups with modify rights for this SQL instance's binary files. Additionally check the owner and users or groups with modify rights for shared software library paths on disk.

If any unauthorized users are granted modify rights or the owner is incorrect, this is a finding.

To determine the location for these instance-specific binaries, Launch SQL Server Management Studio (SSMS) >> Connect to the instance to be reviewed >> Right-click server name in Object Explorer >> Click Facets >> Select the Server facet >> Record the value for the `RootDirectory` facet property.

Navigate to the folder above, and review the `Binn` subdirectory.

Remediation:

Change the ownership of all shared software libraries on disk to the authorized account. Remove any modify permissions granted to unauthorized users or groups.

Additional Information:

CCI-001499

Limit privileges to change software resident within software libraries.

- NIST SP 800-53::CM-5 (6)
- NIST SP 800-53A::CM-5 (6).1
- NIST SP 800-53 Revision 4::CM-5 (6)
- NIST SP 800-53 Revision 5::CM-5 (6)

1.18 SQL6-D0-006600 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must limit privileges to change software modules, to include stored procedures, functions and triggers, and links to software external to SQL Server.

```
GROUP ID: V-213951
RULE ID: SV-213951r960960
```

Rationale:

If the system were to allow any user to make changes to software libraries, then those changes might be implemented without undergoing the appropriate testing and approvals that are part of a robust change management process.

Accordingly, only qualified and authorized individuals must be allowed to obtain access to information system components for purposes of initiating changes, including upgrades and modifications.

Unmanaged changes that occur to the database software libraries or configuration can lead to unauthorized or compromised installations.

Audit:

Review server documentation to determine the process by which shared software libraries are monitored for change. Ensure the process alerts for changes in a file's ownership, modification dates, and hash value at a minimum.

If alerts do not at least hash their value, this is a finding.

To determine the location for these instance-specific binaries:

- Launch SQL Server Management Studio (SSMS) >> Connect to the instance to be reviewed >> Right-click server name in Object Explorer >> Click Facets >> Select the Server facet >> Record the value for the `RootDirectory` facet property

TIP: Use the `Get-FileHash cmdlet` shipped with PowerShell 5.0 to get the SHA-2 hash of one or more files.

Remediation:

Implement and document a process by which changes made to software libraries are monitored and alerted.

A PowerShell based hashing solution is one such process. The Get-FileHash command (https://msdn.microsoft.com/en-us/powershell/reference/5.1/microsoft.powershell.utility/get-filehash) can be used to compute the SHA-2 hash of one or more files.

Using the Export-Clixml command (https://msdn.microsoft.com/powershell/reference/5.1/microsoft.powershell.utility/Export-Clixml), a baseline can be established and exported to a file.

Using the Compare-Object command (https://technet.microsoft.com/en-us/library/ee156812.aspx), a comparison of the latest baseline versus the original baseline can be used to expose the differences.

Additional Information:

CCI-001499

Limit privileges to change software resident within software libraries.

- NIST SP 800-53::CM-5 (6)
- NIST SP 800-53A::CM-5 (6).1
- NIST SP 800-53 Revision 4::CM-5 (6)
- NIST SP 800-53 Revision 5::CM-5 (6)

1.19 SQL6-D0-006700 (Manual)

Profile Applicability:

- SEVERITY: CAT I

Description:

SQL Server software installation account must be restricted to authorized users.

```
GROUP ID: V-213952
RULE ID: SV-213952r960960
```

Rationale:

When dealing with change control issues, it should be noted any changes to the hardware, software, and/or firmware components of the information system and/or application can have significant effects on the overall security of the system.

If the system were to allow any user to make changes to software libraries, then those changes might be implemented without undergoing the appropriate testing and approvals that are part of a robust change management process. Accordingly, only qualified and authorized individuals must be allowed access to information system components for purposes of initiating changes, including upgrades and modifications.

DBA and other privileged administrative or application owner accounts are granted privileges that allow actions that can have a great impact on SQL Server security and operation. It is especially important to grant privileged access to only those persons who are qualified and authorized to use them.

Audit:

From the system documentation, obtain the list of accounts authorized to install/update SQL Server. Run the following PowerShell command to list all users who have installed/modified SQL Server 2016 software and compare the list against those persons who are qualified and authorized to use the software.

sl "C:\program files\microsoft sql server\130\setup bootstrap\Log" Get-ChildItem -Recurse | Select-String -Pattern "LogonUser = "

If any accounts are shown that are not authorized in the system documentation, this is a finding.

Remediation:

From a command prompt, open lusrmgr.msc. Navigate to Users and right-click Individual User. Select Properties >> Member Of.

Configure SQL Server and OS settings and access controls to restrict user access to objects and data that the user is authorized to view/use.

Additional Information:

CCI-001499 Limit privileges to change software resident within software libraries.

- NIST SP 800-53::CM-5 (6)
- NIST SP 800-53A::CM-5 (6).1
- NIST SP 800-53 Revision 4::CM-5 (6)
- NIST SP 800-53 Revision 5::CM-5 (6)

1.20 SQL6-D0-006800 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

Database software, including DBMS configuration files, must be stored in dedicated directories, separate from the host OS and other applications.

```
GROUP ID: V-213953
RULE ID: SV-213953r960960
```

Rationale:

When dealing with change control issues, it should be noted any changes to the hardware, software, and/or firmware components of the information system and/or application can potentially have significant effects on the overall security of the system.

Multiple applications can provide a cumulative negative effect. A vulnerability and subsequent exploit to one application can lead to an exploit of other applications sharing the same security context. For example, an exploit to a web server process that leads to unauthorized administrative access to host system directories can most likely lead to a compromise of all applications hosted by the same system. Database software not installed using dedicated directories both threatens and is threatened by other hosted applications. Access controls defined for one application may by default provide access to the other application's database objects or directories. Any method that provides any level of separation of security context assists in the protection between applications.

Audit:

Determine the directory in which SQL Server has been installed:

Using SQL Server Management Studio's Object Explorer:

- Right-click `[SQL Server Instance]`
- Select "Facets"
- Record the value of `RootDirectory`

Determine the Operating System directory:

- Click "Start"
- Type "Run"
- Press "Enter"
- Type `%windir%`
- Click "Ok"
- Record the value in the address bar

Verify the SQL Server `RootDirectory` is not in the Operating System directory.

Compare the SQL `RootDirectory` and the Operating System directory. If the SQL `RootDirectory` is in the same directory as the Operating System, this is a finding.

Verify the SQL Server `RootDirectory` is not in another application's directory.

Navigate to the SQL `RootDirectory` using Windows Explorer.

Examine each directory for evidence another application is stored in it.

If evidence exists the SQL `RootDirectory` is in another application's directory, this is a finding.

If the SQL `RootDirectory` is not in the Operating System directory or another application's directory. This is not a finding.

Examples:

1. The Operating System directory is `C:\Windows`. The SQL `RootDirectory` is `D:\Program Files\MSSQLSERVER\MSSQL`. The `MSSQLSERVER` directory is not living in the Operating System directory or the directory of another application. This is not a finding.
2. The Operating System directory is `C:\Windows`. The SQL `RootDirectory` is `C:\Windows\MSSQLSERVER\MSSQL`. This is a finding.
3. The Operating System directory is `C:\Windows`. The SQL `RootDirectory` is `D:\Program Files\Microsoft Office\MSSQLSERVER\MSSQL`. The `MSSQLSERVER` directory is in the Microsoft Office directory, which indicates Microsoft Office is installed here. This is a finding.

Remediation:

Re-install SQL Server application components using dedicated directories that are separate from the operating system.

Relocate or reinstall other application software that currently shares directories with SQL Server components.

Separate from the operating system and/or temporary storage.

Additional Information:

CCI-001499

Limit privileges to change software resident within software libraries.

- NIST SP 800-53::CM-5 (6)
- NIST SP 800-53A::CM-5 (6).1
- NIST SP 800-53 Revision 4::CM-5 (6)
- NIST SP 800-53 Revision 5::CM-5 (6)

1.21 SQL6-D0-006900 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

Default demonstration and sample databases, database objects, and applications must be removed.

```
GROUP ID: V-213954
RULE ID: SV-213954r960963
```

Rationale:

Information systems are capable of providing a wide variety of functions and services. Some of the functions and services, provided by default, may not be necessary to support essential organizational operations (e.g., key missions, functions).

It is detrimental for software products to provide, or install by default, functionality exceeding requirements or mission objectives. Examples include, but are not limited to, installing advertising software, demonstrations, or browser plugins not related to requirements or providing a wide array of functionality, not required for every mission, that cannot be disabled.

DBMSs must adhere to the principles of least functionality by providing only essential capabilities.

Demonstration and sample database objects and applications present publicly known attack points for malicious users. These demonstration and sample objects are meant to provide simple examples of coding specific functions and are not developed to prevent vulnerabilities from being introduced to SQL Server and host system.

Audit:

Review the server documentation, if this system is identified as a development or test system, this check is Not Applicable.

If this system is identified as production, gather a listing of databases from the server and look for any matching the following general demonstration database names:

```
pubs
Northwind
AdventureWorks
WorldwideImporters
```

If any of these databases exist, this is a finding.

Remediation:

Remove all demonstration or sample databases from production instances.

Additional Information:

CCI-000381

Configure the system to provide only organization-defined mission essential capabilities.

- NIST SP 800-53::CM-7
- NIST SP 800-53A::CM-7.1 (ii)
- NIST SP 800-53 Revision 4::CM-7 a
- NIST SP 800-53 Revision 5::CM-7 a

1.22 SQL6-D0-007000 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

Unused database components, DBMS software, and database objects must be removed.

```
GROUP ID: V-213955
RULE ID: SV-213955r960963
```

Rationale:

Information systems are capable of providing a wide variety of functions and services. Some of the functions and services, provided by default, may not be necessary to support essential organizational operations (e.g., key missions, functions).

It is detrimental for software products to provide, or install by default, functionality exceeding requirements or mission objectives.

DBMSs must adhere to the principles of least functionality by providing only essential capabilities.

Audit:

From the server documentation, obtain a listing of required components.

Generate a listing of components installed on the server.

Click Start >> Type "SQL Server 2016 Installation Center" >> Launch the program >> Click Tools >> Click "Installed SQL Server features discovery report"

Compare the feature listing against the required components listing.

If any features are installed, but are not required, this is a finding.

Remediation:

Remove all features that are not required.

Additional Information:

CCI-000381

Configure the system to provide only organization-defined mission essential capabilities.

- NIST SP 800-53::CM-7
- NIST SP 800-53A::CM-7.1 (ii)
- NIST SP 800-53 Revision 4::CM-7 a
- NIST SP 800-53 Revision 5::CM-7 a

1.23 SQL6-D0-007100 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

Unused database components that are integrated in SQL Server and cannot be uninstalled must be disabled.

```
GROUP ID: V-213956
RULE ID: SV-213956r960963
```

Rationale:

Information systems are capable of providing a wide variety of functions and services. Some of the functions and services, provided by default, may not be necessary to support essential organizational operations (e.g., key missions, functions).

It is detrimental for software products to provide, or install by default, functionality exceeding requirements or mission objectives.

DBMSs must adhere to the principles of least functionality by providing only essential capabilities.

Unused, unnecessary DBMS components increase the attack vector for SQL Server by introducing additional targets for attack. By minimizing the services and applications installed on the system, the number of potential vulnerabilities is reduced. Components of the system that are unused and cannot be uninstalled must be disabled. The techniques available for disabling components will vary by DBMS product, OS, and the nature of the component and may include DBMS configuration settings, OS service settings, OS file access security, and DBMS user/role permissions.

Audit:

From the server documentation, obtain a listing of required components.

Generate a listing of components installed on the server.

Click Start >> Type "SQL Server 2016 Installation Center" >> Launch the program >> Click Tools >> Click "Installed SQL Server features discovery report"

Compare the feature listing against the required components listing. Note any components that are installed, but not required.

Launch SQL Server Configuration Manager.

If any components that are installed but are not required are not disabled, this is a finding.

If any required components are not installed, this is a finding.

Remediation:

Disable any unused components or features that cannot be uninstalled.

Additional Information:

CCI-000381

Configure the system to provide only organization-defined mission essential capabilities.

- NIST SP 800-53::CM-7
- NIST SP 800-53A::CM-7.1 (ii)
- NIST SP 800-53 Revision 4::CM-7 a
- NIST SP 800-53 Revision 5::CM-7 a

1.24 SQL6-D0-007200 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

Access to `xp_cmdshell` must be disabled, unless specifically required and approved.

```
GROUP ID: V-213957
RULE ID: SV-213957r960963
```

Rationale:

Information systems are capable of providing a wide variety of functions and services. Some of the functions and services, provided by default, may not be necessary to support essential organizational operations (e.g., key missions, functions).

It is detrimental for applications to provide, or install by default, functionality exceeding requirements or mission objectives.

Applications must adhere to the principles of least functionality by providing only essential capabilities.

SQL Server may spawn additional external processes to execute procedures that are defined in the SQL Server but stored in external host files (external procedures). The spawned process used to execute the external procedure may operate within a different OS security context than SQL Server and provide unauthorized access to the host system.

The xp_cmdshell extended stored procedure allows execution of host executables outside the controls of database access permissions. This access may be exploited by malicious users who have compromised the integrity of the SQL Server database process to control the host operating system to perpetrate additional malicious activity.

Audit:

The `xp_cmdshell` extended stored procedure allows execution of host executables outside the controls of database access permissions. This access may be exploited by malicious users who have compromised the integrity of the SQL Server database process to control the host operating system to perpetrate additional malicious activity.

To determine if `xp_cmdshell` is enabled, execute the following commands:

```
EXEC SP_CONFIGURE 'show advanced options', '1';
RECONFIGURE WITH OVERRIDE;
EXEC SP_CONFIGURE 'xp_cmdshell';
```

If the value of `config_value` is `0`, this is not a finding.

Review the system documentation to determine whether the use of `xp_cmdshell` is required and approved. If it is not approved, this is a finding.

Remediation:

Disable use of or remove any external application executable object definitions that are not authorized.

To disable the use of `xp_cmdshell`, from the query prompt:

```
EXEC sp_configure 'show advanced options', 1;
GO
RECONFIGURE;
GO
EXEC sp_configure 'xp_cmdshell', 0;
GO
RECONFIGURE;
GO??
```

Additional Information:

CCI-000381 Configure the system to provide only organization-defined mission essential capabilities.

- NIST SP 800-53::CM-7
- NIST SP 800-53A::CM-7.1 (ii)
- NIST SP 800-53 Revision 4::CM-7 a
- NIST SP 800-53 Revision 5::CM-7 a

1.25 SQL6-D0-007300 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

Access to CLR code must be disabled or restricted, unless specifically required and approved.

```
GROUP ID: V-213958
RULE ID: SV-213958r960963
```

Rationale:

Information systems are capable of providing a wide variety of functions and services. Some of the functions and services, provided by default, may not be necessary to support essential organizational operations (e.g., key missions, functions).

It is detrimental for applications to provide, or install by default, functionality exceeding requirements or mission objectives.

Applications must adhere to the principles of least functionality by providing only essential capabilities.

SQL Server may spawn additional external processes to execute procedures that are defined in the SQL Server but stored in external host files (external procedures). The spawned process used to execute the external procedure may operate within a different OS security context than SQL Server and provide unauthorized access to the host system.

The common language runtime (CLR) component of the .NET Framework for Microsoft Windows in SQL Server allows you to write stored procedures, triggers, user-defined types, user-defined functions, user-defined aggregates, and streaming table-valued functions, using any .NET Framework language, including Microsoft Visual Basic .NET and Microsoft Visual C#. CLR packing assemblies can access resources protected by .NET Code Access Security when it runs managed code. Specifying `UNSAFE` enables the code in the assembly complete freedom to perform operations in the SQL Server process space that can potentially compromise the robustness of SQL Server. `UNSAFE` assemblies can also potentially subvert the security system of either SQL Server or the common language runtime.

Audit:

The common language runtime (CLR) component of the .NET Framework for Microsoft Windows in SQL Server allows you to write stored procedures, triggers, user-defined types, user-defined functions, user-defined aggregates, and streaming table-valued functions, using any .NET Framework language, including Microsoft Visual Basic .NET and Microsoft Visual C#. CLR packing assemblies can access resources protected by .NET Code Access Security when it runs managed code. Specifying UNSAFE enables the code in the assembly complete freedom to perform operations in the SQL Server process space that can potentially compromise the robustness of SQL Server. UNSAFE assemblies can also potentially subvert the security system of either SQL Server or the common language runtime.

To determine if CLR is enabled, execute the following commands:

```
EXEC SP_CONFIGURE 'show advanced options', '1';
RECONFIGURE WITH OVERRIDE;
EXEC SP_CONFIGURE 'clr enabled';
```

If the value of config_value is 0, this is not a finding.

If the value of config_value is 1, review the system documentation to determine whether the use of CLR code is approved. If it is not approved, this is a finding.

If CLR code is approved, check the database for UNSAFE assembly permission using the following script:

```
USE [master]
SELECT *
FROM sys.assemblies
WHERE permission_set_desc != 'SAFE'
AND is_user_defined = 1;
```

If any records are returned, review the system documentation to determine if the use of UNSAFE assemblies is approved. If it is not approved, this is a finding.

Remediation:

Disable use of or remove any CLR code that is not authorized.

To disable the use of CLR, from the query prompt:

```
sp_configure 'show advanced options', 1;
GO
RECONFIGURE;
GO
sp_configure 'clr enabled', 0;
GO
RECONFIGURE;
GO
```

For any approved CLR code with Unsafe or External permissions, use the ALTER ASSEMBLY to change the Permission set for the Assembly and ensure a certificate is configured.

Additional Information:

CCI-000381

Configure the system to provide only organization-defined mission essential capabilities.

- NIST SP 800-53::CM-7
- NIST SP 800-53A::CM-7.1 (ii)
- NIST SP 800-53 Revision 4::CM-7 a
- NIST SP 800-53 Revision 5::CM-7 a

1.26 SQL6-D0-007400 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

Access to Non-Standard extended stored procedures must be disabled or restricted, unless specifically required and approved.

```
GROUP ID: V-213959
RULE ID: SV-213959r960963
```

Rationale:

Information systems are capable of providing a wide variety of functions and services. Some of the functions and services, provided by default, may not be necessary to support essential organizational operations (e.g., key missions, functions).

It is detrimental for applications to provide, or install by default, functionality exceeding requirements or mission objectives.

Applications must adhere to the principles of least functionality by providing only essential capabilities.

SQL Server may spawn additional external processes to execute procedures that are defined in the SQL Server but stored in external host files (external procedures). The spawned process used to execute the external procedure may operate within a different OS security context than SQL Server and provide unauthorized access to the host system.

Extended stored procedures are DLLs that an instance of SQL Server can dynamically load and run. Extended stored procedures run directly in the address space of an instance of SQL Server and are programmed by using the SQL Server Extended Stored Procedure API. Non-Standard extended stored procedures can compromise the integrity of the SQL Server process. This feature will be removed in a future version of Microsoft SQL Server. Do not use this feature in new development work, and modify applications that currently use this feature as soon as possible.

Audit:

Extended stored procedures are DLLs that an instance of SQL Server can dynamically load and run. Extended stored procedures run directly in the address space of an instance of SQL Server and are programmed by using the SQL Server Extended Stored Procedure API.

Non-Standard extended stored procedures can compromise the integrity of the SQL Server process. This feature will be removed in a future version of Microsoft SQL Server. Do not use this feature in new development work, and modify applications that currently use this feature as soon as possible.

To determine if non-standard extended stored procedures exist, run the following:

```
--------------------------------------------------------------------------
USE [master]
GO
DECLARE @xplist AS TABLE
(
       xp_name sysname,
       source_dll nvarchar(255)
)
INSERT INTO @xplist
EXEC sp_helpextendedproc

SELECT X.xp_name, X.source_dll, O.is_ms_shipped FROM @xplist X JOIN
sys.all_objects O ON X.xp_name = O.name WHERE O.is_ms_shipped = 0 ORDER BY
X.xp_name
--------------------------------------------------------------------------
```

If any records are returned, review the system documentation to determine whether the use of Non-Standard extended stored procedures are required and approved.

If it is not approved, this is a finding.

Remediation:

Remove any Non-Standard extended stored procedures that are not documented and approved.

```
sp_dropextendedproc 'proc name'
```

Additional Information:

CCI-000381

Configure the system to provide only organization-defined mission essential capabilities.

- NIST SP 800-53::CM-7
- NIST SP 800-53A::CM-7.1 (ii)
- NIST SP 800-53 Revision 4::CM-7 a
- NIST SP 800-53 Revision 5::CM-7 a

1.27 SQL6-D0-007500 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

Access to linked servers must be disabled or restricted, unless specifically required and approved.

```
GROUP ID: V-213960
RULE ID: SV-213960r1018585
```

Rationale:

Information systems are capable of providing a wide variety of functions and services. Some of the functions and services, provided by default, may not be necessary to support essential organizational operations (e.g., key missions, functions). It is detrimental for applications to provide, or install by default, functionality exceeding requirements or mission objectives. Applications must adhere to the principles of least functionality by providing only essential capabilities. SQL Server may spawn additional external processes to execute procedures that are defined in the SQL Server but stored in external host files (external procedures). The spawned process used to execute the external procedure may operate within a different OS security context than SQL Server and provide unauthorized access to the host system. A linked server allows for access to distributed, heterogeneous queries against OLE DB data sources. After a linked server is created, distributed queries can be run against this server, and queries can join tables from more than one data source. If the linked server is defined as an instance of SQL Server, remote stored procedures can be executed. This access may be exploited by malicious users who have compromised the integrity of the SQL Server.

Audit:

A linked server allows for access to distributed, heterogeneous queries against OLE DB data sources. After a linked server is created, distributed queries can be run against this server, and queries can join tables from more than one data source. If the linked server is defined as an instance of SQL Server, remote stored procedures can be executed.

To obtain a list of linked servers, execute the following command:

```
SELECT name
FROM sys.servers s
WHERE s.is_linked = 1
```

Review the system documentation to determine whether the linked servers listed are required and approved. If it is not approved, this is a finding.

Run the following to get a linked server login mapping:

```
SELECT s.name, p.principal_id, l.remote_name
FROM sys.servers s
```

```
JOIN sys.linked_logins l ON s.server_id = l.server_id
LEFT JOIN sys.server_principals p ON l.local_principal_id = p.principal_id
WHERE s.is_linked = 1
```

Review the linked login mapping and check the remote name as it can impersonate sysadmin. If a login in the list is impersonating sysadmin and system documentation does not require this, it is a finding.

Remediation:

Disable use of or remove any linked servers that are not authorized.

To remove a linked server and all associated logins run the following:

sp_dropserver 'LinkedServerName', 'droplogins';

To remove a login from a linked server run the following:

EXEC sp_droplinkedsrvlogin 'LoginName', NULL;

Additional Information:

CCI-000381

Configure the system to provide only organization-defined mission essential capabilities.

- NIST SP 800-53::CM-7
- NIST SP 800-53A::CM-7.1 (ii)
- NIST SP 800-53 Revision 4::CM-7 a
- NIST SP 800-53 Revision 5::CM-7 a

1.28 SQL6-D0-007600 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must be configured to prohibit or restrict the use of organization-defined protocols as defined in the PPSM CAL and vulnerability assessments.

```
GROUP ID: V-213961
RULE ID: SV-213961r1043177
```

Rationale:

In order to prevent unauthorized connection of devices, unauthorized transfer of information, or unauthorized tunneling (i.e., embedding of data types within data types), organizations must disable or restrict unused or unnecessary protocols on information systems.

Applications are capable of providing a wide variety of functions and services. Some of the functions and services provided by default may not be necessary to support essential organizational operations. Additionally, it is sometimes convenient to provide multiple services from a single component (e.g., email and web services); however, doing so increases risk over limiting the services provided by any one component.

To support the requirements and principles of least functionality, the application must support the organizational requirements providing only essential capabilities and limiting the use of protocols to only those required, authorized, and approved to conduct official business or to address authorized quality of life issues.

SQL Server using protocols deemed unsafe is open to attack through those protocols. This can allow unauthorized access to the database and through the database to other components of the information system.

Audit:

To determine the protocol(s) enabled for SQL Server, open SQL Server Configuration Manager. In the left-hand pane, expand SQL Server Network Configuration. Click on the entry for the SQL Server instance under review: "Protocols for ". The right-hand pane displays the protocols enabled for the instance.

If Named Pipes is enabled and not specifically required and authorized, this is a finding.

If any listed protocol is enabled but not authorized, this is a finding.

Remediation:

In SQL Server Configuration Manager >> SQL Server Network Configuration >> Protocols, right-click on each listed protocol that is enabled but not authorized and Select `Disable`.

Additional Information:

CCI-000382

Configure the system to prohibit or restrict the use of organization-defined prohibited or restricted functions, system ports, protocols, software, and/or services.

- NIST SP 800-53::CM-7
- NIST SP 800-53A::CM-7.1 (iii)
- NIST SP 800-53 Revision 4::CM-7 b
- NIST SP 800-53 Revision 5::CM-7 b

1.29 SQL6-D0-007700 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must be configured to prohibit or restrict the use of organization-defined ports, as defined in the PPSM CAL and vulnerability assessments.

```
GROUP ID: V-213962
RULE ID: SV-213962r1043177
```

Rationale:

In order to prevent unauthorized connection of devices, unauthorized transfer of information, or unauthorized tunneling (i.e., embedding of data types within data types), organizations must disable or restrict unused or unnecessary physical and logical ports on information systems.

Applications are capable of providing a wide variety of functions and services. Some of the functions and services provided by default may not be necessary to support essential organizational operations. Additionally, it is sometimes convenient to provide multiple services from a single component (e.g., email and web services); however, doing so increases risk over limiting the services provided by any one component.

To support the requirements and principles of least functionality, the application must support the organizational requirements providing only essential capabilities and limiting the use of ports to only those required, authorized, and approved to conduct official business or to address authorized quality of life issues.

SQL Server using ports deemed unsafe is open to attack through those ports. This can allow unauthorized access to the database and through the database to other components of the information system.

Audit:

Review SQL Server Configuration for the ports used by SQL Server.

To determine whether SQL Server is configured to use a fixed port or dynamic ports, in the right-hand pane double-click on the TCP/IP entry, to open the Properties dialog. (The default fixed port is 1433.)

If these are in conflict with PPSM guidance, and not explained and approved in the system documentation, this is a finding.

Remediation:

Use SQL Server Configuration to change the ports used by SQL Server to comply with PPSM guidance, or document the need for other ports, and obtain written approval. Close ports no longer needed.

Additional Information:

CCI-000382

Configure the system to prohibit or restrict the use of organization-defined prohibited or restricted functions, system ports, protocols, software, and/or services.

- NIST SP 800-53::CM-7
- NIST SP 800-53A::CM-7.1 (iii)
- NIST SP 800-53 Revision 4::CM-7 b
- NIST SP 800-53 Revision 5::CM-7 b

1.30 SQL6-D0-007800 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must uniquely identify and authenticate organizational users (or processes acting on behalf of organizational users).

```
GROUP ID: V-213963
RULE ID: SV-213963r1051115
```

Rationale:

To assure accountability and prevent unauthenticated access, organizational users must be identified and authenticated to prevent potential misuse and compromise of the system.

Organizational users include organizational employees or individuals the organization deems to have equivalent status of employees (e.g., contractors). Organizational users (and any processes acting on behalf of users) must be uniquely identified and authenticated for all accesses, except the following:

- (i) Accesses explicitly identified and documented by the organization. Organizations document specific user actions that can be performed on the information system without identification or authentication; and
- (ii) Accesses that occur through authorized use of group authenticators without individual authentication. Organizations may require unique identification of individuals using shared accounts, for detailed accountability of individual activity.

Audit:

Review SQL Server users to determine whether shared accounts exist. (This does not include the case where SQL Server has a guest or public account that is providing access to publicly available information.)

If accounts are determined to be shared, determine if individuals are first individually authenticated. Where an application connects to SQL Server using a standard, shared account, ensure that it also captures the individual user identification and passes it to SQL Server.

If individuals are not individually authenticated before using the shared account (e.g., by the operating system or possibly by an application making calls to the database), this is a finding.

If accounts are determined to be shared, determine if they are directly accessible to end users. If so, this is a finding.

Remediation:

Remove user-accessible shared accounts and use individual userIDs.

Configure applications to ensure successful individual authentication prior to shared account access.

Ensure each user's identity is received and used in audit data in all relevant circumstances.

Additional Information:

CCI-000764

Uniquely identify and authenticate organizational users and associate that unique identification with processes acting on behalf of those users.

- NIST SP 800-53::IA-2
- NIST SP 800-53A::IA-2.1
- NIST SP 800-53 Revision 4::IA-2
- NIST SP 800-53 Revision 5::IA-2

1.31 SQL6-D0-007900 (Manual)

Profile Applicability:

- SEVERITY: CAT I

Description:

If DBMS authentication using passwords is employed, SQL Server must enforce the DOD standards for password complexity and lifetime.

```
GROUP ID: V-213964
RULE ID: SV-213964r1112499
```

Rationale:

Windows Authentication is the default authentication mode and is much more secure than SQL Server Authentication. Windows Authentication uses Kerberos security protocol, provides password policy enforcement with regard to complexity validation for strong passwords, provides support for account lockout, and supports password expiration. A connection made using Windows Authentication is sometimes called a trusted connection, because SQL Server trusts the credentials provided by Windows.

By using Windows Authentication, Windows groups can be created at the domain level, and a login can be created on SQL Server for the entire group. Managing access at the domain level can simplify account administration.

OS/enterprise authentication and identification must be used (SRG-APP-000023-DB-000001). Native SQL Server authentication may be used only when circumstances make it unavoidable and must be documented and Authorizing Official (AO)-approved.

The DOD standard for authentication is DOD-approved PKI certificates. Authentication based on User ID and Password may be used only when it is not possible to employ a PKI certificate and requires AO approval.

In such cases, the DOD standards for password complexity and lifetime must be implemented. DBMS products that can inherit the rules for these from the operating system or access control program (e.g., Microsoft Active Directory) must be configured to do so. For other DBMSs, the rules must be enforced using available configuration parameters or custom code.

Audit:

Check for use of SQL Server Authentication:

```
SELECT CASE SERVERPROPERTY('IsIntegratedSecurityOnly') WHEN 1 THEN 'Windows
Authentication' WHEN 0 THEN 'SQL Server Authentication' END as
[Authentication Mode]
```

If the returned value in the `Authentication Mode` column is `Windows Authentication`, this is not a finding.

If the returned value in the Authentication Mode column is SQL Server Authentication, SQL Server should be configured to inherit password complexity and password lifetime rules from the operating system.

Review SQL Server to ensure logons are created with respect to the complexity settings and password lifetime rules by running the statement:

```
SELECT [name], is_expiration_checked, is_policy_checked
FROM sys.sql_logins
```

Review any accounts returned by the query other than the disabled SA account, ##MS_PolicyTsqlExecutionLogin##, ##MS_PolicyEventProcessingLogin##, ##MS_SSISServerCleanupJobLogin##, and other internal accounts that start with ##MS.

If any account does not have both is_expiration_checked and is_policy_checked equal to 1, this is a finding.

Review the operating system settings relating to password complexity.

To check the server operating system for password complexity:

Navigate to Start >> All Programs >> Administrative Tools >> Local Security Policy, and to review the local policies on the machine, go to Account Policy >> Password Policy.

Ensure the DISA Windows Password Policy is set on the SQL Server member server. If any are not, this is a finding.

Remediation:

Configure the SQL Server operating system and SQL Server logins for compliance.

Review the Operating System settings relating to password complexity.

Ensure SQL Server is configured to inherit password complexity rules from the operating system for SQL logins. Ensure check of policy and expiration are enforced when SQL logins are created.

```
CREATE LOGIN <login_name> WITH PASSWORD= <enterStrongPasswordHere>,
CHECK_EXPIRATION = ON, CHECK_POLICY = ON;
```

Additional Information:

CCI-004066

For password-based authentication, enforce organization-defined composition and complexity rules.

- NIST SP 800-53 Revision 5::IA-5 (1) (h)

CCI-000192

The information system enforces password complexity by the minimum number of upper case characters used.

- NIST SP 800-53::IA-5 (1) (a)
- NIST SP 800-53A::IA-5 (1).1 (v)
- NIST SP 800-53 Revision 4::IA-5 (1) (a)

1.32 SQL6-D0-008000 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

Contained databases must use Windows principals.

```
GROUP ID: V-213965
RULE ID: SV-213965r1018610
```

Rationale:

OS/enterprise authentication and identification must be used (SRG-APP-000023-DB-000001). Native DBMS authentication may be used only when circumstances make it unavoidable, and must be documented and Authorizing Official (AO)-approved.

The DOD standard for authentication is DOD-approved PKI certificates. Authentication based on User ID and Password may be used only when it is not possible to employ a PKI certificate, and requires AO approval.

In such cases, the DOD standards for password complexity and lifetime must be implemented. DBMS products that can inherit the rules for these from the operating system or access control program (e.g., Microsoft Active Directory) must be configured to do so. For other DBMSs, the rules must be enforced using available configuration parameters or custom code.

Audit:

Execute the following query to determine if contained databases are used:

```
SELECT * FROM sys.databases WHERE containment = 1
```

If any records are returned. Check the server documentation for a list of authorized contained database users. Ensure contained database users are not using SQL Authentication.

```
EXEC sp_MSforeachdb 'USE [?]; SELECT DB_NAME() AS DatabaseName, * FROM
sys.database_principals WHERE authentication_type = 2'
```

If any records are returned, this is a finding.

Remediation:

Configure the SQL Server contained databases to have users originating from Windows principals. Remove any users not created from Windows principals.

Additional Information:

CCI-004066

For password-based authentication, enforce organization-defined composition and complexity rules.

- NIST SP 800-53 Revision 5::IA-5 (1) (h)

CCI-000192

The information system enforces password complexity by the minimum number of upper case characters used.

- NIST SP 800-53::IA-5 (1) (a)
- NIST SP 800-53A::IA-5 (1).1 (v)
- NIST SP 800-53 Revision 4::IA-5 (1) (a)

1.33 SQL6-D0-008200 (Manual)

Profile Applicability:

- SEVERITY: CAT I

Description:

If passwords are used for authentication, SQL Server must transmit only encrypted representations of passwords.

```
GROUP ID: V-213966
RULE ID: SV-213966r1051304
```

Rationale:

The DOD standard for authentication is DOD-approved PKI certificates.

Authentication based on User ID and Password may be used only when it is not possible to employ a PKI certificate and requires AO approval.

In such cases, passwords need to be protected at all times, and encryption is the standard method for protecting passwords during transmission.

SQL Server passwords sent in clear text format across the network are vulnerable to discovery by unauthorized users. Disclosure of passwords may easily lead to unauthorized access to the database.

Audit:

1. Launch SSMS and connect to the SQL Server to be reviewed.
2. Right-click the instance and select "Properties".
3. Navigate to the "Security" tab.

If the value for `Server authentication` is `Windows Authentication mode`, this requirement is Not Applicable.

From a command prompt, open SQL Server Configuration Manager by typing `sqlservermanager13.msc` and pressing "Enter".

Navigate to SQL Server Configuration Manager >> SQL Server Network Configuration. Right-click on "Protocols", where there is a placeholder for the SQL Server instance name, and click on "Properties".

On the `Flags` tab, if `Force Encryption` is set to `NO`, this is a finding.

On the `Flags` tab, if `Force Encryption` is set to `YES`, examine the certificate used on the `Certificate` tab.

If it is not a DOD approved certificate, or if no certificate is listed, this is a finding.

For clustered instances, the Certificate will **NOT** be shown in the SQL Server Configuration Manager.

1. From a command prompt, navigate to the certificate store where the Full Qualified Domain Name (FQDN) certificate is stored by typing `certlm.msc` and pressing "Enter".
2. In the left side of the window, expand the "Personal" folder, and click "Certificates".
3. Verify that the Certificate with the FQDN name is issued by the DOD. Double-click the certificate, click the "Details" tab, and note the value for the Thumbprint.
4. Verify the value for the "Thumbprint" field matches the value in the registry by running regedit and looking at `HKLM\SOFTWARE\Microsoft\Microsoft SQL Server\<instance>\MSSQLServer\SuperSocketNetLib\Certificate`.
5. Run this on each node of the cluster.

If any nodes have a certificate in use by SQL that is not issued or approved by DOD, this is a finding.

Remediation:

Configure SQL Server to encrypt authentication data for remote connections using DOD-approved cryptography.

Deploy encryption to the SQL Server Network Connections.

From a command prompt, open SQL Server Configuration Manager by typing `sqlservermanager13.msc` and pressing "ENTER".

Navigate to SQL Server Configuration Manager >> SQL Server Network Configuration. Right-click on Protocols for, where is a placeholder for the SQL Server instance name, and click on "Properties".

In the "Protocols for Properties" dialog box, on the "Certificate" tab, select the DOD certificate from the drop-down for the Certificate box and then click "OK". On the "Flags" tab, in the "ForceEncryption" box, select "Yes" and then click "OK" to close the dialog box. Restart the SQL Server service.

For clustered instances, install the certificate after setting "Force Encryption" to "Yes" in SQL Server Configuration Manager.

1. Navigate to the certificate store where the FQDN certificate is stored by typing `certlm.msc` and pressing "ENTER".
2. On the "Properties" page for the certificate, go to the "Details" tab and copy the `thumbprint` value of the certificate to a `Notepad` window.
3. Remove the spaces between the hex characters in the `thumbprint` value in `Notepad`.
4. Start regedit, navigate to the following registry key, and copy the value from step 2: `HKLM\SOFTWARE\Microsoft\Microsoft SQL Server\<instance>\MSSQLServer\SuperSocketNetLib\Certificate`

5. If the SQL virtual server is currently on this node, failover to another node in the cluster and then reboot the node where the registry change occurred.
6. Repeat this procedure on all the nodes.

Additional Information:

CCI-000197

For password-based authentication, transmit passwords only over cryptographically-protected channels.

- NIST SP 800-53::IA-5 (1) (c)
- NIST SP 800-53A::IA-5 (1).1 (v)
- NIST SP 800-53 Revision 4::IA-5 (1) (c)
- NIST SP 800-53 Revision 5::IA-5 (1) (c)

1.34 SQL6-D0-008300 (Manual)

Profile Applicability:

- SEVERITY: CAT I

Description:

Confidentiality of information during transmission is controlled through the use of an approved TLS version.

```
GROUP ID: V-213967
RULE ID: SV-213967r961029
```

Rationale:

Transport Layer Security (TLS) encryption is a required security setting as a number of known vulnerabilities have been reported against Secure Sockets Layer (SSL) and earlier versions of TLS. Encryption of private information is essential to ensuring data confidentiality. If private information is not encrypted, it can be intercepted and easily read by an unauthorized party. SQL Server must use a FIPS-approved minimum TLS version 1.2, and all non-FIPS-approved SSL and TLS versions must be disabled. NIST SP 800-52 Rev.2 specifies the preferred configurations for government systems.

References: TLS Support 1.2 for SQL Server: https://support.microsoft.com/en-us/kb/3135244 TLS Registry Settings: https://docs.microsoft.com/en-us/windows-server/security/tls/tls-registry-settings

Audit:

Access the SQL Server.

Access an administrator command prompt.

Type `regedit` to launch the Registry Editor.

Navigate to:

`HKLM\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\TLS 1.2`

If this key does not exist, this is a finding.

Verify a `REG_DWORD` value of `0` for `DisabledByDefault` and a value of `1` for `Enabled` for both Client and Server.

Navigate to:

```
HKLM\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\TLS 1.0
HKLM\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\TLS 1.1
```

```
HKLM\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\SSL 2.0
HKLM\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\SSL 3.0
```

Under each key, verify a `REG_DWORD` value of `1` for `DisabledByDefault` and a value of `0` for `Enabled` for both Client and Server subkeys.

If any of the respective registry paths are non-existent or contain values other than specified above, this is a finding. If Vendor documentation supporting the configuration is provided, reduce this finding to a CAT 3.

Remediation:

Important Note: Incorrectly modifying the Windows Registry can result in serious system errors. Before making any modifications, ensure you have a recent backup of the system and registry settings.

Access the SQL Server.

Access an administrator command prompt.

Type `regedit` to launch the Registry Editor.

Enable TLS 1.2:

1. Navigate to the path `HKLM\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols`.

   a. If the `TLS 1.2` key does not exist, right-click `Protocols`.

   b. Click `New`.

   c. Click `Key`.

   d. Type the name `TLS 1.2`.

2. Navigate to the TLS 1.2 subkey.

   a. If the subkey `Client` does not exist, right-click `TLS 1.2`

   b. Click `New`.

   c. Click `Key`.

   d. Type the name `Client`.

   e. Repeat steps A – D for the `Server` subkey.

3. Navigate to the "Client" subkey.

   a. If the value `Enabled` does not exist, right-click on `Client`.

b. Click New.

c. Click DWORD.

d. Enter Enabled as the name.

e. Repeat steps A-D for the value DisabledByDefault.

4. Double-click Enabled.
5. In Value Data, enter 1.
6. Click "OK".
7. Double-click DisabledByDefault.
8. In Value Data, enter 0.
9. Click "OK".
10. Repeat steps 3 – 9 for the Server subkey.

Disable unwanted SSL/TLS protocol versions:

1. Navigate to the path HKLM\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols.

   a. If the TLS 1.0 key does not exist, right-click Protocols.

   b. Click New.

   c. Click Key.

   d. Type the name TLS 1.0.

2. Navigate to the TLS 1.0 subkey.

   a. If the subkey Client does not exist, right-click TLS 1.0.

   b. Click New.

   c. Click Key.

   d. Type the name Client.

   e. Repeat steps A – D for the Server subkey.

3. Navigate to the Client subkey.

   a. If the value Enabled does not exist, right-click on Client.

   b. Click New.

   c. Click DWORD.

d. Enter `Enabled` as the name.

e. Repeat steps A-D for the value `DisabledByDefault`.

4. Double-click `Enabled`.
5. In Value Data, enter `0`. 6.Click "OK".
6. Double-click `DisabledByDefault`.
7. In Value Data, enter `1`.
8. Click "OK".
9. Repeat steps 3 – 9 for the `Server` subkey.
10. Repeat steps 1 – 10 for `TLS 1.1`, `SSL 2.0`, and `SSL 3.0`.

Additional Information:

CCI-000197

For password-based authentication, transmit passwords only over cryptographically-protected channels.

- NIST SP 800-53::IA-5 (1) (c)
- NIST SP 800-53A::IA-5 (1).1 (v)
- NIST SP 800-53 Revision 4::IA-5 (1) (c)
- NIST SP 800-53 Revision 5::IA-5 (1) (c)

1.35 SQL6-D0-008400 (Manual)

Profile Applicability:

- SEVERITY: CAT I

Description:

SQL Server must enforce authorized access to all PKI private keys stored/utilized by SQL Server.

```
GROUP ID: V-213968
RULE ID: SV-213968r961041
```

Rationale:

The DoD standard for authentication is DoD-approved PKI certificates. PKI certificate-based authentication is performed by requiring the certificate holder to cryptographically prove possession of the corresponding private key.

If the private key is stolen, an attacker can use the private key(s) to impersonate the certificate holder. In cases where SQL Server-stored private keys are used to authenticate SQL Server to the system's clients, loss of the corresponding private keys would allow an attacker to successfully perform undetected man in the middle attacks against SQL Server system and its clients.

Both the holder of a digital certificate and the issuing authority must take careful measures to protect the corresponding private key. Private keys should always be generated and protected in FIPS 140-2 or FIPS 140-3 validated cryptographic modules.

All access to the private key(s) of SQL Server must be restricted to authorized and authenticated users. If unauthorized users have access to one or more of SQL Server's private keys, an attacker could gain access to the key(s) and use them to impersonate the database on the network or otherwise perform unauthorized actions.

Audit:

Review system configuration to determine whether FIPS compliant support has been enabled.

Start >> Control Panel >> Administrative Tools >> Local Security Policy >> Local Policies >> Security Options

Ensure that "System cryptography: Use FIPS-compliant algorithms for encryption, hashing, and signing" is enabled.

If "System cryptography: Use FIPS-compliant algorithms for encryption, hashing, and signing" is not enabled, this is a finding.

For more information, see https://support.microsoft.com/en-us/kb/3141890.

Remediation:

Enable use of FIPS-compliant algorithms.

Start >> Control Panel >> Administrative Tools >> Local Security Policy >> Local Policies >> Security Options

Double-click "System cryptography: Use FIPS-compliant algorithms for encryption, hashing, and signing."

Click Enabled >> Apply.

Additional Information:

CCI-000186

For public key-based authentication, enforce authorized access to the corresponding private key.

- NIST SP 800-53::IA-5 (2)
- NIST SP 800-53A::IA-5 (2).1
- NIST SP 800-53 Revision 4::IA-5 (2) (b)
- NIST SP 800-53 Revision 5::IA-5 (2) (a) (1)

1.36 SQL6-D0-008700 (Manual)

Profile Applicability:

- SEVERITY: CAT I

Description:

SQL Server must use NIST FIPS 140-2 or 140-3 validated cryptographic modules for cryptographic operations.

```
GROUP ID: V-213969
RULE ID: SV-213969r961050
```

Rationale:

Use of weak or not validated cryptographic algorithms undermines the purposes of utilizing encryption and digital signatures to protect data. Weak algorithms can be easily broken, and not validated cryptographic modules may not implement algorithms correctly. Unapproved cryptographic modules or algorithms should not be relied on for authentication, confidentiality, or integrity. Weak cryptography could allow an attacker to gain access to, and modify data stored in, the database as well as the administration settings of SQL Server.

Applications, including DBMSs, utilizing cryptography are required to use approved NIST FIPS 140-2 or 140-3 validated cryptographic modules that meet the requirements of applicable federal laws, Executive Orders, directives, policies, regulations, standards, and guidance.

NSA Type- (where =1, 2, 3, 4) products are NSA-certified, hardware-based encryption modules.

The standard for validating cryptographic modules will transition to the NIST FIPS 140-3 publication.

FIPS 140-2 modules can remain active for up to five years after validation or until September 21, 2026, when the FIPS 140-2 validations will be moved to the historical list. Even on the historical list, CMVP supports the purchase and use of these modules for existing systems. While Federal Agencies decide when they move to FIPS 140-3 only modules, purchasers are reminded that for several years there may be a limited selection of FIPS 140-3 modules from which to choose. CMVP recommends purchasers consider all modules that appear on the Validated Modules Search Page: https://csrc.nist.gov/projects/cryptographic-module-validation-program/validated-modules

More information on the FIPS 140-3 transition can be found here: https://csrc.nist.gov/Projects/fips-140-3-transition-effort/

Audit:

In Windows, open Administrative Tools >> Local Security Policy. Expand Local Policies >> Security Options. In the right-side pane, find "System cryptography: Use FIPS compliant algorithms for encryption, hashing, and signing".

If, in the "Security Setting" column, the value is "Disabled," this is a finding.

https://support.microsoft.com/en-us/kb/955720

Remediation:

In Windows, open Administrative Tools >> Local Security Policy. Expand Local Policies >> Security Options. In the right-side pane, double-click on "System cryptography: Use FIPS compliant algorithms for encryption, hashing, and signing".

In the dialog box that appears, if the radio buttons are active, click "Enabled", and then click "Apply". If the radio buttons are grayed out, use Group Policy Management (on the appropriate server for this domain) to enforce the Enabled policy, and deploy it to the server(s) running SQL Server.

Additional Information:

CCI-000803

Implement mechanisms for authentication to a cryptographic module that meet the requirements of applicable laws, Executive Orders, directives, policies, regulations, standards, and guidance for such authentication.

- NIST SP 800-53::IA-7
- NIST SP 800-53A::IA-7.1
- NIST SP 800-53 Revision 4::IA-7
- NIST SP 800-53 Revision 5::IA-7

1.37 SQL6-D0-008800 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must uniquely identify and authenticate non-organizational users (or processes acting on behalf of non-organizational users).

```
GROUP ID: V-213970
RULE ID: SV-213970r961053
```

Rationale:

Non-organizational users include all information system users other than organizational users, which include organizational employees or individuals the organization deems to have equivalent status of employees (e.g., contractors, guest researchers, individuals from allied nations).

Non-organizational users must be uniquely identified and authenticated for all accesses other than those accesses explicitly identified and documented by the organization when related to the use of anonymous access, such as accessing a web server.

Accordingly, a risk assessment is used in determining the authentication needs of the organization.

Scalability, practicality, and security are simultaneously considered in balancing the need to ensure ease of use for access to federal information and information systems with the need to protect and adequately mitigate risk to organizational operations, organizational assets, individuals, other organizations, and the Nation.

Audit:

Review documentation, SQL Server settings, and authentication system settings to determine if non-organizational users are individually identified and authenticated when logging onto the system.

Execute the following query to obtain a list of logins on the SQL Server and ensure all accounts are uniquely identifiable:

```
SELECT name, type_desc FROM sys.server_principals WHERE type in ('S','U')
```

If accounts are determined to be shared, determine if individuals are first individually authenticated. Where an application connects to SQL Server using a standard, shared account, ensure that it also captures the individual user identification and passes it to SQL Server.

If the documentation indicates that this is a public-facing, read-only (from the point of view of public users) database that does not require individual authentication, this is not a finding.

If non-organizational users are not uniquely identified and authenticated, this is a finding.

Remediation:

Ensure all logins are uniquely identifiable and authenticate all non-organizational users who log onto the system. This likely would be done via a combination of the operating system with unique accounts and the SQL Server by ensuring mapping to individual accounts. Verify server documentation to ensure accounts are documented and unique.

Additional Information:

CCI-000804

Uniquely identify and authenticate non-organizational users or processes acting on behalf of non-organizational users.

- NIST SP 800-53::IA-8
- NIST SP 800-53A::IA-8.1
- NIST SP 800-53 Revision 4::IA-8
- NIST SP 800-53 Revision 5::IA-8

1.38 SQL6-D0-009200 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must maintain the authenticity of communications sessions by guarding against man-in-the-middle attacks that guess at Session ID values.

```
GROUP ID: V-213971
RULE ID: SV-213971r1043181
```

Rationale:

One class of man-in-the-middle, or session hijacking, attack involves the adversary guessing at valid session identifiers based on patterns in identifiers already known.

The preferred technique for thwarting guesses at Session IDs is the generation of unique session identifiers using a FIPS 140-2 or FIPS 140-3 approved random number generator.

However, it is recognized that available DBMS products do not all implement the preferred technique yet may have other protections against session hijacking. Therefore, other techniques are acceptable, provided they are demonstrated to be effective.

Audit:

Verify that Windows is configured to require the use of FIPS compliant algorithms.

Click Start >> Type "Local Security Policy" >> Press Enter >> Expand "Local Policies" >> Select "Security Options" >> Locate "System Cryptography: Use FIPS compliant algorithms for encryption, hashing, and signing."

If the Security Setting for this option is `Disabled`, this is a finding.

Remediation:

Configure Windows to require the use of FIPS compliant algorithms.

Click Start >> Type "Local Security Policy" >> Press Enter >> Expand "Local Policies" >> Select "Security Options" >> Locate "System Cryptography: Use FIPS compliant algorithms for encryption, hashing, and signing." >> Change the Setting option to `Enabled` >> Restart Windows

Additional Information:

CCI-001188

Generate a unique session identifier for each session with organization-defined randomness requirements.

- NIST SP 800-53::SC-23 (4)
- NIST SP 800-53A::SC-23 (4).1 (ii)
- NIST SP 800-53 Revision 4::SC-23 (3)
- NIST SP 800-53 Revision 5::SC-23 (3)

1.39 SQL6-D0-009500 (Manual)

Profile Applicability:

- SEVERITY: CAT I

Description:

SQL Server must protect the confidentiality and integrity of all information at rest.

```
GROUP ID: V-213972
RULE ID: SV-213972r961128
```

Rationale:

This control is intended to address the confidentiality and integrity of information at rest in non-mobile devices and covers user information and system information. Information at rest refers to the state of information when it is located on a secondary storage device (e.g., disk drive, tape drive) within an organizational information system. Applications and application users generate information throughout the course of their application use.

User data generated, as well as application-specific configuration data, needs to be protected. Organizations may choose to employ different mechanisms to achieve confidentiality and integrity protections, as appropriate.

If the confidentiality and integrity of SQL Server data is not protected, the data will be open to compromise and unauthorized modification.

Audit:

Review system documentation to determine whether the system handles classified information. If the system does not handle classified information, the severity of this check should be downgraded to Category II.

If the application owner and Authorizing Official have determined that encryption of data at rest is required, ensure the data on secondary devices is encrypted.

If full-disk encryption is being used, this is not a finding.

If data encryption is required, ensure the data is encrypted before being put on the secondary device by executing:

```
SELECT
d.name AS [Database Name],
CASE e.encryption_state
WHEN 0 THEN 'No database encryption key present, no encryption'
WHEN 1 THEN 'Unencrypted'
WHEN 2 THEN 'Encryption in progress'
WHEN 3 THEN 'Encrypted'
WHEN 4 THEN 'Key change in progress'
WHEN 5 THEN 'Decryption in progress'
WHEN 6 THEN 'Protection change in progress'
```

```
END AS [Encryption State]
FROM sys.dm_database_encryption_keys e
RIGHT JOIN sys.databases d ON DB_NAME(e.database_id) = d.name
WHERE d.name NOT IN ('master','model','msdb')
ORDER BY [Database Name];
```

For each user database where encryption is required, verify that encryption is in effect. If not, this is a finding.

Verify that there are physical security measures, operating system access control lists and organizational controls appropriate to the sensitivity level of the data in the database(s). If not, this is a finding.

Remediation:

Apply appropriate controls to protect the confidentiality and integrity of data on a secondary device. Where encryption is required, this can be done by full-disk encryption or by database encryption.

To enable database encryption, create a master key, create a database encryption key, and protect it by using mechanisms tied to the master key, and then set encryption on.

Implement physical security measures, operating system access control lists and organizational controls appropriate to the sensitivity level of the data in the database(s).

Additional Information:

CCI-001199

Protects the confidentiality and/or integrity of organization-defined information at rest.

- NIST SP 800-53::SC-28
- NIST SP 800-53A::SC-28.1
- NIST SP 800-53 Revision 4::SC-28
- NIST SP 800-53 Revision 5::SC-28

1.40 SQL6-D0-009600 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

The Service Master Key must be backed up and stored in a secure location that is not on the SQL Server.

```
GROUP ID: V-213973
RULE ID: SV-213973r961128
```

Rationale:

Backup and recovery of the Service Master Key may be critical to the complete recovery of the database. Creating this backup should be one of the first administrative actions performed on the server. Not having this key can lead to loss of data during recovery.

Audit:

Review procedures for and evidence of backup of the Server Service Master Key in the System Security Plan.

If the procedures or evidence does not exist, this is a finding.

If the procedures do not indicate that a backup of the Service Master Key is stored in a secure location that is not on the SQL Server, this is a finding.

If procedures do not indicate access restrictions to the Service Master Key backup, this is a finding.

Remediation:

Document and implement procedures to safely back up and store the Service Master Key in a secure location that is not on the SQL Server. Include in the procedures methods to establish evidence of backup and storage, and careful, restricted access and restoration of the Service Master Key.

```
BACKUP SERVICE MASTER KEY TO FILE = 'path_to_file'
ENCRYPTION BY PASSWORD = 'password';
```

As this requires a password, take care to ensure it is not exposed to unauthorized persons or stored as plain text.

Additional Information:

CCI-001199

Protects the confidentiality and/or integrity of organization-defined information at rest.

- NIST SP 800-53::SC-28
- NIST SP 800-53A::SC-28.1
- NIST SP 800-53 Revision 4::SC-28
- NIST SP 800-53 Revision 5::SC-28

1.41 SQL6-D0-009700 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

The Master Key must be backed up and stored in a secure location that is not on the SQL Server.

```
GROUP ID: V-213974
RULE ID: SV-213974r961128
```

Rationale:

Backup and recovery of the Master Key may be critical to the complete recovery of the database. Not having this key can lead to loss of data during recovery.

Audit:

If the application owner and authorizing official have determined that encryption of data at rest is not required, this is not a finding.

Review procedures for and evidence of backup of the Master Key in the System Security Plan.

If the procedures or evidence does not exist, this is a finding.

If the procedures do not indicate that a backup of the Master Key is stored in a secure location that is not on the SQL Server, this is a finding.

If procedures do not indicate access restrictions to the Master Key backup, this is a finding.

Remediation:

Document and implement procedures to safely back up and store the Master Key in a secure location that is not on the SQL Server. Include in the procedures methods to establish evidence of backup and storage, and careful, restricted access and restoration of the Master Key.

```
BACKUP MASTER KEY TO FILE = 'path_to_file'
ENCRYPTION BY PASSWORD = 'password';
```

As this requires a password, take care to ensure it is not exposed to unauthorized persons or stored as plain text.

Additional Information:

CCI-001199

Protects the confidentiality and/or integrity of organization-defined information at rest.

- NIST SP 800-53::SC-28
- NIST SP 800-53A::SC-28.1
- NIST SP 800-53 Revision 4::SC-28
- NIST SP 800-53 Revision 5::SC-28

1.42 SQL6-D0-009800 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must prevent unauthorized and unintended information transfer via shared system resources.

```
GROUP ID: V-213975
RULE ID: SV-213975r961149
```

Rationale:

The purpose of this control is to prevent information, including encrypted representations of information, produced by the actions of a prior user/role (or the actions of a process acting on behalf of a prior user/role) from being available to any current user/role (or current process) that obtains access to a shared system resource (e.g., registers, main memory, secondary storage) after the resource has been released back to the information system. Control of information in shared resources is also referred to as object reuse.

Audit:

Review system documentation to determine if Common Criteria Compliance is not required due to potential impact on system performance.

SQL Server Residual Information Protection (RIP) requires a memory allocation to be overwritten with a known pattern of bits before memory is reallocated to a new resource. Meeting the RIP standard can contribute to improved security; however, overwriting the memory allocation can slow performance. After the common criteria compliance enabled option is enabled, the overwriting occurs.

Review the Instance configuration:

```
SELECT value_in_use
FROM sys.configurations
WHERE name = 'common criteria compliance enabled'
```

If `value_in_use` is set to `1` this is not a finding. If `value_in_use` is set to `0` this is a finding.

NOTE: Enabling this feature may impact performance on highly active SQL Server instances. If an exception justifying setting SQL Server Residual Information Protection (RIP) to disabled (value_in_use set to "0") has been documented and approved, then this may be downgraded to a CAT III finding.

Remediation:

Configure SQL Server to effectively protect the private resources of one process or user from unauthorized access by another user or process.

```
sp_configure 'show advanced options', 1;
GO
RECONFIGURE;
GO
sp_configure 'common criteria compliance enabled', 1;
GO
RECONFIGURE
GO
```

Additional Information:

CCI-001090

Prevent unauthorized and unintended information transfer via shared system resources.

- NIST SP 800-53::SC-4
- NIST SP 800-53A::SC-4.1
- NIST SP 800-53 Revision 4::SC-4
- NIST SP 800-53 Revision 5::SC-4

1.43 SQL6-D0-009900 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must prevent unauthorized and unintended information transfer via Instant File Initialization (IFI).

```
GROUP ID: V-213976
RULE ID: SV-213976r961149
```

Rationale:

The purpose of this control is to prevent information, including encrypted representations of information, produced by the actions of a prior user/role (or the actions of a process acting on behalf of a prior user/role) from being available to any current user/role (or current process) that obtains access to a shared system resource (e.g., registers, main memory, secondary storage) after the resource has been released back to the information system. Control of information in shared resources is also referred to as object reuse.

When Instant File Initialization (IFI) is in use, the deleted disk content is overwritten only as new data is written to the files. For this reason, the deleted content might be accessed by an unauthorized principal until some other data writes on that specific area of the data file.

Audit:

Review system configuration to determine whether IFI support has been enabled (by default in SQL Server 2016).

Start >> Control Panel >> System and Security >> Administrative Tools >> Local Security Policy >> Local Policies >> User Rights Assignment >> Perform volume maintenance tasks

The default SQL service account for a default instance is `NT SERVICE\MSSQLSERVER` or for a named instance is NT `SERVICE\MSSQL$InstanceName`.

If the SQL service account or SQL service SID has been granted "Perform volume maintenance tasks" Local Rights Assignment, this means that Instant File Initialization (IFI) is enabled.

Review the system documentation to determine if Instant File Initialization (IFI) is required.

If IFI is enabled but not documented as required, this is a finding.

If IFI is not enabled, this is not a finding.

Remediation:

If IFI is not documented as being required, disable instant file initialization for the instance of SQL Server by removing the SQL Service SID and/or service account from the "Perform volume maintenance tasks" Local Rights Assignment.

Additional Information:

CCI-001090

Prevent unauthorized and unintended information transfer via shared system resources.

- NIST SP 800-53::SC-4
- NIST SP 800-53A::SC-4.1
- NIST SP 800-53 Revision 4::SC-4
- NIST SP 800-53 Revision 5::SC-4

1.44 SQL6-D0-010000 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

Access to database files must be limited to relevant processes and to authorized, administrative users.

```
GROUP ID: V-213977
RULE ID: SV-213977r961149
```

Rationale:

SQL Server must prevent unauthorized and unintended information transfer via shared system resources. Permitting only SQL Server processes and authorized, administrative users to have access to the files where the database resides helps ensure that those files are not shared inappropriately and are not open to backdoor access and manipulation.

Audit:

Review the permissions granted to users by the operating system/file system on the database files, database log files, and database backup files.

To obtain the location of SQL Server data, transaction log, and backup files, open and execute the supplemental file "Get SQL Data and Backup `Directories.sql`".

For each of the directories returned by the above script, verify whether the correct permissions have been applied.

1. Launch Windows Explorer.
2. Navigate to the folder.
3. Right-click the folder and click "Properties".
4. Navigate to the "Security" tab.
5. Review the listing of principals and permissions.

```
Account Type                  Directory Type         Permission
------------------------------------------------------------------------
-----------------
Database Administrators       ALL                    Full Control
SQL Server Service SID        Data; Log; Backup;     Full Control
SQL Server Agent Service SID  Backup                 Full Control
SYSTEM                        ALL                    Full Control
CREATOR OWNER                 ALL                    Full Control
```

For information on how to determine a "Service SID", go to:

https://aka.ms/sql-service-sids

Additional permission requirements, including full directory permissions and operating system rights for SQL Server, are documented at:

https://aka.ms/sqlservicepermissions

If any additional permissions are granted but not documented as authorized, this is a finding.

Remediation:

Remove any unauthorized permission grants from SQL Server data, log, and backup directories.

1. On the "Security" tab, highlight the user entry.
2. Click "Remove".

Additional Information:

CCI-001090

Prevent unauthorized and unintended information transfer via shared system resources.

- NIST SP 800-53::SC-4
- NIST SP 800-53A::SC-4.1
- NIST SP 800-53 Revision 4::SC-4
- NIST SP 800-53 Revision 5::SC-4

1.45 SQL6-D0-010100 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must reveal detailed error messages only to documented and approved individuals or roles.

```
GROUP ID: V-213978
RULE ID: SV-213978r1067807
```

Rationale:

If SQL Server provides too much information in error logs and administrative messages to the screen, this could lead to compromise. The structure and content of error messages need to be carefully considered by the organization and development team. The extent to which the information system is able to identify and handle error conditions is guided by organizational policy and operational requirements.

Some default DBMS error messages can contain information that could aid an attacker in, among others things, identifying the database type, host address, or state of the database. Custom errors may contain sensitive customer information.

It is important that detailed error messages be visible only to those who are authorized to view them; that general users receive only generalized acknowledgment that errors have occurred; and that these generalized messages appear only when relevant to the user's task. For example, a message along the lines of, "An error has occurred. Unable to save your changes. If this problem persists, please contact your help desk." would be relevant. A message such as "Warning: your transaction generated a large number of page splits" would likely not be relevant. "ABGQ is not a valid widget code." would be appropriate; but "The `INSERT` statement conflicted with the `FOREIGN KEY` constraint `WidgetTransactionFK`. The conflict occurred in database `DB7`, table `dbo.WidgetMaster`, column `WidgetCode`" would not, as it reveals too much about the database structure.

Audit:

Error messages within applications, custom database code (stored procedures, triggers) must be enforced by guidelines and code reviews practices.

SQL Server generates certain system events and user-defined events to the SQL Server error log. The SQL Server error log can be viewed using SQL Server Management Studio GUI. All users granted the security admin or sysadmin level of permission are able to view the logs. Review the users returned in the following script:

```
USE master
GO
SELECT Name
```

```
FROM syslogins
WHERE (sysadmin = 1 or securityadmin = 1)
and hasaccess = 1;
```

If any nonauthorized users have access to the SQL Server Error Log located at `Program Files\Microsoft SQL Server\MSSQL.n\MSSQL\LOG`, this is a finding.

In addition, the SQL Server Error Log is also located at `Program Files\Microsoft SQL Server\MSSQL.n\MSSQL\LOG\`. Review the permissions on this folder to ensure that only authorized users are listed.

If any nonauthorized users have access to the SQL Server Error Log in SQL Server Management Studio or if documentation does not exist stating that full error messages must be returned, this is a finding.

Otherwise, verify if trace flag `3625` is enabled to mask certain system-level error information returned to nonadministrative users.

Launch SQL Server Configuration Manager:

- Select SQL Server Services >> SQL Server. Select the SQL Server, then right-click and select "Properties". Select "Startup Parameters" tab and verify `-T3625` exists in the dialogue window.

OR

Run the query:

```
DBCC TRACESTATUS;
```

If `TraceFlag 3625` does not return with `Status = 1` and if documentation does not exist stating that full error messages must be returned, this is a finding.

Remediation:

Configure audit logging, tracing and/or custom code in the database or application to record detailed error messages generated by SQL Server, for review by authorized personnel.

If any nonauthorized users have access to the SQL Server Error Log in SQL Server Management Studio, use the `REVOKE` or `DENY` commands to remove them from the security admin or sysadmin roles.

If any nonauthorized users have access to the SQL Server Error Log located at `Program Files\Microsoft SQL Server\MSSQL.n\MSSQL\LOG`, remove their permissions.

Consider enabling trace flag `3625` to mask certain system-level error information returned to nonadministrative users.

Configure audit logging, tracing and/or custom code in the database or application to record detailed error messages generated by SQL Server, for review by authorized personnel.

If any nonauthorized users have access to the SQL Server Error Log in SQL Server Management Studio, use the `REVOKE` or `DENY` commands to remove them from the security admin or sysadmin roles.

If any nonauthorized users have access to the SQL Server Error Log located at `Program Files\Microsoft SQL Server\MSSQL.n\MSSQL\LOG`, remove their permissions.

Enable trace flag `3625` to mask certain system-level error information returned to nonadministrative users.

To launch SQL Server Configuration Manager, select "SQL Server Services", select the "SQL Server", then right-click and select "Properties". Select "Startup Parameters" tab, enter `-T3625`, and then click "Add". Click "OK", then restart SQL instance.

To launch SQL Server Configuration Manager, click "SQL Services", open the instance properties, click the "Service Parameters" tab, enter `-T3625`. Click "Add", click "OK", then restart SQL instance.

Additional Information:

CCI-001314

Reveal error messages only to organization-defined personnel or roles.

- NIST SP 800-53::SI-11 c
- NIST SP 800-53A::SI-11.1 (iv)
- NIST SP 800-53 Revision 4::SI-11 b
- NIST SP 800-53 Revision 5::SI-11 b

1.46 SQL6-D0-010400 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must prevent non-privileged users from executing privileged functions, to include disabling, circumventing, or altering implemented security safeguards/countermeasures.

```
GROUP ID: V-213979
RULE ID: SV-213979r961353
```

Rationale:

Preventing non-privileged users from executing privileged functions mitigates the risk that unauthorized individuals or processes may gain unnecessary access to information or privileges.

System documentation should include a definition of the functionality considered privileged.

Depending on circumstances, privileged functions can include, for example, establishing accounts, performing system integrity checks, or administering cryptographic key management activities. Non-privileged users are individuals that do not possess appropriate authorizations. Circumventing intrusion detection and prevention mechanisms or malicious code protection mechanisms are examples of privileged functions that require protection from non-privileged users.

A privileged function in SQL Server/database context is any operation that modifies the structure of the database, its built-in logic, or its security settings. This would include all Data Definition Language (DDL) statements and all security-related statements. In an SQL environment, it encompasses, but is not necessarily limited to:

- CREATE
- ALTER
- DROP
- GRANT
- REVOKE
- DENY

There may also be Data Manipulation Language (DML) statements that, subject to context, should be regarded as privileged. Possible examples include:

```
TRUNCATE TABLE;
DELETE, or
DELETE affecting more than n rows, for some n, or
DELETE without a WHERE clause;
```

```
UPDATE or
UPDATE affecting more than n rows, for some n, or
UPDATE without a WHERE clause;
```

Any SELECT, INSERT, UPDATE, or DELETE to an application-defined security table executed by other than a security principal.

Depending on the capabilities of SQL Server and the design of the database and associated applications, the prevention of unauthorized use of privileged functions may be achieved by means of DBMS security features, database triggers, other mechanisms, or a combination of these.

Audit:

Review server-level securables and built-in role membership to ensure only authorized users have privileged access and the ability to create server-level objects and grant permissions to themselves or others.

Review the system documentation to determine the required levels of protection for DBMS server securables, by type of login.

Review the permissions in place on the server. If the actual permissions do not match the documented requirements, this is a finding.

Get all permission assignments to logins and roles:

```
SELECT DISTINCT
    CASE
        WHEN SP.class_desc IS NOT NULL THEN
            CASE
                WHEN SP.class_desc = 'SERVER' AND S.is_linked = 0 THEN
'SERVER'
                WHEN SP.class_desc = 'SERVER' AND S.is_linked = 1 THEN
'SERVER (linked)'
                ELSE SP.class_desc
            END
        WHEN E.name IS NOT NULL THEN 'ENDPOINT'
        WHEN S.name IS NOT NULL AND S.is_linked = 0 THEN 'SERVER'
        WHEN S.name IS NOT NULL AND S.is_linked = 1 THEN 'SERVER (linked)'
        WHEN P.name IS NOT NULL THEN 'SERVER_PRINCIPAL'
        ELSE '???'
    END                     AS [Securable Class],
    CASE
        WHEN E.name IS NOT NULL THEN E.name
        WHEN S.name IS NOT NULL THEN S.name
        WHEN P.name IS NOT NULL THEN P.name
        ELSE '???'
    END                     AS [Securable],
    P1.name                 AS [Grantee],
    P1.type_desc            AS [Grantee Type],
    sp.permission_name      AS [Permission],
    sp.state_desc           AS [State],
    P2.name                 AS [Grantor],
    P2.type_desc            AS [Grantor Type]
```

```
FROM
    sys.server_permissions SP
    INNER JOIN sys.server_principals P1
        ON P1.principal_id = SP.grantee_principal_id
    INNER JOIN sys.server_principals P2
        ON P2.principal_id = SP.grantor_principal_id

    FULL OUTER JOIN sys.servers S
        ON  SP.class_desc = 'SERVER'
        AND S.server_id = SP.major_id

    FULL OUTER JOIN sys.endpoints E
        ON  SP.class_desc = 'ENDPOINT'
        AND E.endpoint_id = SP.major_id

    FULL OUTER JOIN sys.server_principals P
        ON  SP.class_desc = 'SERVER_PRINCIPAL'
        AND P.principal_id = SP.major_id
```

Get all server role memberships:

```
SELECT
    R.name    AS [Role],
    M.name    AS [Member]
FROM
    sys.server_role_members X
    INNER JOIN sys.server_principals R ON R.principal_id =
X.role_principal_id
    INNER JOIN sys.server_principals M ON M.principal_id =
X.member_principal_id
```

The `CONTROL SERVER` permission is similar but not identical to the sysadmin fixed server role. Permissions do not imply role memberships and role memberships do not grant permissions. (e.g., `CONTROL SERVER` does not imply membership in the sysadmin fixed server role.)

Ensure only the documented and approved logins have privileged functions in SQL Server.

If the current configuration does not match the documented baseline, this is a finding.

Remediation:

Restrict the granting of permissions to server-level securables to only those authorized. Most notably, members of sysadmin and securityadmin built-in instance-level roles, `CONTROL SERVER` permission, and use of the GRANT with GRANT permission.

Additional Information:

CCI-002235

Prevent non-privileged users from executing privileged functions.

- NIST SP 800-53 Revision 4::AC-6 (10)
- NIST SP 800-53 Revision 5::AC-6 (10)

1.47 SQL6-D0-010500 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

Use of credentials and proxies must be restricted to necessary cases only.

```
GROUP ID: V-213980
RULE ID: SV-213980r961359
```

Rationale:

In certain situations, to provide required functionality, a DBMS needs to execute internal logic (stored procedures, functions, triggers, etc.) and/or external code modules with elevated privileges. However, if the privileges required for execution are at a higher level than the privileges assigned to organizational users invoking the functionality applications/programs, those users are indirectly provided with greater privileges than assigned by organizations.

Privilege elevation must be utilized only where necessary and protected from misuse.

Audit:

Review the server documentation to obtain a listing of accounts used for executing external processes. Execute the following query to obtain a listing of accounts currently configured for use by external processes.

```
SELECT C.name AS credential_name, C.credential_identity
FROM sys.credentials C
GO

SELECT P.name AS proxy_name, C.name AS credential_name, C.credential_identity
FROM sys.credentials C
JOIN msdb.dbo.sysproxies P ON C.credential_id = P.credential_id
WHERE P.enabled = 1
GO
```

If any Credentials or SQL Agent Proxy accounts are returned that are not documented and authorized, this is a finding.

Remediation:

Remove any SQL Agent Proxy accounts and credentials that are not authorized.

```
DROP CREDENTIAL <Credential Name>
GO

USE [msdb]
EXEC sp_delete_proxy @proxy_name = '<Proxy Name>'
GO
```

Additional Information:

CCI-002233

Prevent the organization-defined software from executing at higher privilege levels than users executing the software.

- NIST SP 800-53 Revision 4::AC-6 (8)
- NIST SP 800-53 Revision 5::AC-6 (8)

1.48 SQL6-D0-010900 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must allocate audit record storage capacity in accordance with organization-defined audit record storage requirements.

```
GROUP ID: V-213983
RULE ID: SV-213983r1018595
```

Rationale:

In order to ensure sufficient storage capacity for the audit logs, SQL Server must be able to allocate audit record storage capacity. Although another requirement (SRG-APP-000515-DB-000318) mandates that audit data be off-loaded to a centralized log management system, it remains necessary to provide space on the database server to serve as a buffer against outages and capacity limits of the off-loading mechanism.

The task of allocating audit record storage capacity is usually performed during initial installation of SQL Server and is closely associated with the DBA and system administrator roles. The DBA or system administrator will usually coordinate the allocation of physical drive space with the application owner/installer and the application will prompt the installer to provide the capacity information, the physical location of the disk, or both.

In determining the capacity requirements, consider such factors as: total number of users; expected number of concurrent users during busy periods; number and type of events being monitored; types and amounts of data being captured; the frequency/speed with which audit records are off-loaded to the central log management system; and any limitations that exist on SQL Server's ability to reuse the space formerly occupied by off-loaded records.

Audit:

If the database is setup to write audit logs using `APPLICATION` or `SECURITY` event logs rather than writing to a file, this is Not Applicable.

Check the server documentation for the SQL Audit file size configurations. Locate the Audit file path and drive.

```
SELECT max_file_size, max_rollover_files, log_file_path AS "Audit Path"
FROM sys.server_file_audits
```

Calculate the space needed as the maximum file size and number of files from the SQL Audit File properties.

If the calculated product of the `max_file_size` times the `max_rollover_files` exceeds the size of the storage location, this is a finding;

OR if `max_file_size` is set to `0` (Unlimited), this is a finding;

OR if `max_rollover_files` are set to `0` (None) or `2147483647` (Unlimited), this is a finding.

Remediation:

Review the SQL Audit file location; ensure the destination has enough space available to accommodate the maximum total size of all files that could be written.

Configure the maximum number of audit log files that are to be generated, staying within the number of logs the system was sized to support.

Update the `max_files` or `max_rollover_files` parameter of the audits to ensure the correct number of files is defined.

If writing to application event logs or security logs, space considerations are covered in the Windows Server STIGs. Be sure to reference these depending on the OS in use.

Additional Information:

CCI-001849

Allocate audit log storage capacity to accommodate organization-defined audit log retention requirements.

- NIST SP 800-53 Revision 4::AU-4
- NIST SP 800-53 Revision 5::AU-4

1.49 SQL6-D0-011000 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must provide a warning to appropriate support staff when allocated audit record storage volume reaches 75% of maximum audit record storage capacity.

```
GROUP ID: V-213984
RULE ID: SV-213984r961398
```

Rationale:

Organizations are required to use a central log management system, so, under normal conditions, the audit space allocated to SQL Server on its own server will not be an issue. However, space will still be required on the server for SQL Server audit records in transit, and, under abnormal conditions, this could fill up. Since a requirement exists to halt processing upon audit failure, a service outage would result.

If support personnel are not notified immediately upon storage volume utilization reaching 75%, they are unable to plan for storage capacity expansion.

The appropriate support staff include, at a minimum, the ISSO and the DBA/SA.

Monitoring of free space can be accomplished using Microsoft System Center or a third-party monitoring tool.

Audit:

The operating system and SQL Server offer a number of methods for checking the drive or volume free space. Locate the destination drive where SQL Audits are stored and review system configuration.

If no alert exist to notify support staff in the event the SQL Audit drive reaches 75%, this is a finding.

Remediation:

Utilize operating system alerting mechanisms, SQL Agent, Operations Management tools, and/or third-party tools to configure the system to notify appropriate support staff immediately upon storage volume utilization reaching 75%.

Additional Information:

CCI-001855

Provide a warning to organization-defined personnel, roles, and/or locations within an organization-defined time period when allocated audit log storage volume reaches an organization-defined percentage of repository maximum audit log storage capacity.

- NIST SP 800-53 Revision 4::AU-5 (1)
- NIST SP 800-53 Revision 5::AU-5 (1)

1.50 SQL6-D0-011100 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must provide an immediate real-time alert to appropriate support staff of all audit log failures.

```
GROUP ID: V-213985
RULE ID: SV-213985r961401
```

Rationale:

It is critical for the appropriate personnel to be aware if a system is at risk of failing to process audit logs as required. Without a real-time alert, security personnel may be unaware of an impending failure of the audit capability, and system operation may be adversely affected.

The appropriate support staff include, at a minimum, the ISSO and the DBA/SA.

A failure of database auditing will result in either the database continuing to function without auditing or in a complete halt to database operations. When audit processing fails, appropriate personnel must be alerted immediately to avoid further downtime or unaudited transactions

Alerts provide organizations with urgent messages. Real-time alerts provide these messages immediately (i.e., the time from event detection to alert occurs in seconds or less). Alerts can be generated using tools like the SQL Server Agent Alerts and Database Mail.

Audit:

Review SQL Server settings, OS, or third-party logging software settings to determine whether a real-time alert will be sent to the appropriate personnel when auditing fails for any reason.

If real-time alerts are not sent upon auditing failure, this is a finding.

Remediation:

Configure the system to provide immediate real-time alerts to appropriate support staff when an audit log failure occurs.

Additional Information:

CCI-001858

Provide an alert in an organization-defined real-time-period to organization-defined personnel, roles, and/or locations when organization-defined audit failure events requiring real-time alerts occur.

- NIST SP 800-53 Revision 4::AU-5 (2)
- NIST SP 800-53 Revision 5::AU-5 (2)

1.51 SQL6-D0-011200 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must record time stamps in audit records and application data that can be mapped to Coordinated Universal Time (UTC, formerly GMT).

```
GROUP ID: V-213986
RULE ID: SV-213986r961443
```

Rationale:

If time stamps are not consistently applied and there is no common time reference, it is difficult to perform forensic analysis.

Time stamps generated by SQL Server must include date and time. Time is commonly expressed in Coordinated Universal Time (UTC), a modern continuation of Greenwich Mean Time (GMT), or local time with an offset from UTC.

Audit:

SQL Server audits store the timestamp in UTC time.

Determine if the computer is joined to a domain.

```
SELECT DEFAULT_DOMAIN()[DomainName]
```

If this is not `NULL`, this is not a finding.

If the computer is not joined to a domain, determine what the time source is. (Run the following command in an elevated PowerShell session.)

```
w32tm /query /source
```

If the results of the command return `Local CMOS Clock` and is not documented with justification and AO authorization, this is a finding.

If the OS does not synchronize with a time server, review the procedure for maintaining accurate time on the system.

If such a procedure does not exist, this is a finding.

If the procedure exists, review evidence that the correct time is actually maintained.

If the evidence indicates otherwise, this is a finding.

Remediation:

Where possible, configure the operating system to automatic synchronize with an official time server, using NTP.

Where there is reason not to implement automatic synchronization with an official time server, using NTP, document the reason, and the procedure for maintaining the correct time, and obtain AO approval. Enforce the procedure.

Additional Information:

CCI-001890

Record time stamps for audit records that use Coordinated Universal Time, have a fixed local time offset from Coordinated Universal Time, or that include the local time offset as part of the time stamp.

- NIST SP 800-53 Revision 4::AU-8 b
- NIST SP 800-53 Revision 5::AU-8 b

1.52 SQL6-D0-011400 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must enforce access restrictions associated with changes to the configuration of the instance.

```
GROUP ID: V-213987
RULE ID: SV-213987r961461
```

Rationale:

Failure to provide logical access restrictions associated with changes to configuration may have significant effects on the overall security of the system.

When dealing with access restrictions pertaining to change control, it should be noted that any changes to the hardware, software, and/or firmware components of the information system can potentially have significant effects on the overall security of the system.

Accordingly, only qualified and authorized individuals should be allowed to obtain access to system components for the purposes of initiating changes, including upgrades and modifications.

Audit:

Obtain a list of logins who have privileged permissions and role memberships in SQL.

Execute the following query to obtain a list of logins and roles and their respective permissions assignment:

```
SELECT p.name AS Principal,
p.type_desc AS Type,
sp.permission_name AS Permission,
sp.state_desc AS State
FROM sys.server_principals p
INNER JOIN sys.server_permissions sp ON p.principal_id =
sp.grantee_principal_id
WHERE sp.permission_name = 'CONTROL SERVER'
OR sp.state = 'W'
```

Execute the following query to obtain a list of logins and their role memberships.

```
SELECT m.name AS Member,
m.type_desc AS Type,
r.name AS Role
FROM sys.server_principals m
INNER JOIN sys.server_role_members rm ON m.principal_id =
rm.member_principal_id
INNER JOIN sys.server_principals r ON rm.role_principal_id = r.principal_id
```

```
WHERE r.name IN ('sysadmin','securityadmin','serveradmin')
```

Check the server documentation to verify the logins and roles returned are authorized. If the logins and/or roles are not documented and authorized, this is a finding.

Remediation:

Revoke unauthorized permissions from principals.

https://msdn.microsoft.com/en-us/library/ms186308.aspx

Remove unauthorized logins from roles.

```
ALTER SERVER ROLE DROP MEMBER login;
```

https://technet.microsoft.com/en-us/library/ee677634.aspx

Additional Information:

CCI-001813

Enforce access restrictions using organization-defined mechanisms.

- NIST SP 800-53 Revision 4::CM-5 (1)
- NIST SP 800-53 Revision 5::CM-5 (1) (a)

1.53 SQL6-D0-011500 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

Windows must enforce access restrictions associated with changes to the configuration of the SQL Server instance.

```
GROUP ID: V-213988
RULE ID: SV-213988r961461
```

Rationale:

Failure to provide logical access restrictions associated with changes to configuration may have significant effects on the overall security of the system.

When dealing with access restrictions pertaining to change control, it should be noted that any changes to the hardware, software, and/or firmware components of the information system can potentially have significant effects on the overall security of the system.

Accordingly, only qualified and authorized individuals should be allowed to obtain access to system components for the purposes of initiating changes, including upgrades and modifications.

Audit:

Obtain a list of users who have privileged access to the server via the local Administrators group.

1. Launch `lusrmgr.msc`
2. Select Groups
3. Double-click Administrators

Alternatively, execute the following command in PowerShell:

```
net localgroup administrators
```

Check the server documentation to verify the users returned are authorized.

If the users are not documented and authorized, this is a finding.

Remediation:

Remove users from the local Administrators group who are not authorized.

Additional Information:

CCI-001813

Enforce access restrictions using organization-defined mechanisms.

- NIST SP 800-53 Revision 4::CM-5 (1)
- NIST SP 800-53 Revision 5::CM-5 (1) (a)

1.54 SQL6-D0-011800 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must produce audit records of its enforcement of access restrictions associated with changes to the configuration of SQL Server or database(s).

```
GROUP ID: V-213989
RULE ID: SV-213989r1018611
```

Rationale:

Without auditing the enforcement of access restrictions against changes to configuration, it would be difficult to identify attempted attacks and an audit trail would not be available for forensic investigation for after-the-fact actions.

Enforcement actions are the methods or mechanisms used to prevent unauthorized changes to configuration settings. Enforcement action methods may be as simple as denying access to a file based on the application of file permissions (access restriction). Audit items may consist of lists of actions blocked by access restrictions or changes identified after the fact.

Audit:

Determine if an audit is configured to capture denied actions and started by executing the following query:

```
SELECT name AS 'Audit Name',
status_desc AS 'Audit Status',
audit_file_path AS 'Current Audit File'
FROM sys.dm_server_audit_status
```

If no records are returned, this is a finding.

Execute the following query to verify the following events are included in the server audit specification:

```
APPLICATION_ROLE_CHANGE_PASSWORD_GROUP,
AUDIT_CHANGE_GROUP,
BACKUP_RESTORE_GROUP,
DATABASE_CHANGE_GROUP,
DATABASE_OBJECT_ACCESS_GROUP,
DATABASE_OBJECT_CHANGE_GROUP,
DATABASE_OBJECT_OWNERSHIP_CHANGE_GROUP,
DATABASE_OBJECT_PERMISSION_CHANGE_GROUP,
DATABASE_OWNERSHIP_CHANGE_GROUP,
DATABASE_OPERATION_GROUP,
DATABASE_PERMISSION_CHANGE_GROUP,
DATABASE_PRINCIPAL_CHANGE_GROUP,
DATABASE_PRINCIPAL_IMPERSONATION_GROUP,
```

```
DATABASE_ROLE_MEMBER_CHANGE_GROUP,
DBCC_GROUP,
LOGIN_CHANGE_PASSWORD_GROUP,
SCHEMA_OBJECT_CHANGE_GROUP,
SCHEMA_OBJECT_OWNERSHIP_CHANGE_GROUP,
SCHEMA_OBJECT_PERMISSION_CHANGE_GROUP,
SERVER_OBJECT_CHANGE_GROUP,
SERVER_OBJECT_OWNERSHIP_CHANGE_GROUP,
SERVER_OBJECT_PERMISSION_CHANGE_GROUP,
SERVER_OPERATION_GROUP,
SERVER_PERMISSION_CHANGE_GROUP,
SERVER_PRINCIPAL_IMPERSONATION_GROUP,
SERVER_ROLE_MEMBER_CHANGE_GROUP,
SERVER_STATE_CHANGE_GROUP,
TRACE_CHANGE_GROUP

SELECT a.name AS 'AuditName',
s.name AS 'SpecName',
d.audit_action_name AS 'ActionName',
d.audited_result AS 'Result'
FROM sys.server_audit_specifications s
JOIN sys.server_audits a ON s.audit_guid = a.audit_guid
JOIN sys.server_audit_specification_details d ON s.server_specification_id =
d.server_specification_id
WHERE a.is_state_enabled = 1
AND d.audit_action_name IN (
'APPLICATION_ROLE_CHANGE_PASSWORD_GROUP',
'AUDIT_CHANGE_GROUP',
'BACKUP_RESTORE_GROUP',
'DATABASE_CHANGE_GROUP',
'DATABASE_OBJECT_ACCESS_GROUP',
'DATABASE_OBJECT_CHANGE_GROUP',
'DATABASE_OBJECT_OWNERSHIP_CHANGE_GROUP',
'DATABASE_OBJECT_PERMISSION_CHANGE_GROUP',
'DATABASE_OWNERSHIP_CHANGE_GROUP',
'DATABASE_OPERATION_GROUP',
'DATABASE_PERMISSION_CHANGE_GROUP',
'DATABASE_PRINCIPAL_CHANGE_GROUP',
'DATABASE_PRINCIPAL_IMPERSONATION_GROUP',
'DATABASE_ROLE_MEMBER_CHANGE_GROUP',
'DBCC_GROUP',
'LOGIN_CHANGE_PASSWORD_GROUP',
'SCHEMA_OBJECT_CHANGE_GROUP',
'SCHEMA_OBJECT_OWNERSHIP_CHANGE_GROUP',
'SCHEMA_OBJECT_PERMISSION_CHANGE_GROUP',
'SERVER_OBJECT_CHANGE_GROUP',
'SERVER_OBJECT_OWNERSHIP_CHANGE_GROUP',
'SERVER_OBJECT_PERMISSION_CHANGE_GROUP',
'SERVER_OPERATION_GROUP',
'SERVER_PERMISSION_CHANGE_GROUP',
'SERVER_PRINCIPAL_IMPERSONATION_GROUP',
'SERVER_ROLE_MEMBER_CHANGE_GROUP',
'SERVER_STATE_CHANGE_GROUP',
'TRACE_CHANGE_GROUP'
)
Order by d.audit_action_name
```

If the identified groups are not returned, this is a finding.

Remediation:

Add the required events to the server audit specification to audit denied actions.

```
USE [master];
GO

ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION WITH (STATE
= OFF);
GO

ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(APPLICATION_ROLE_CHANGE_PASSWORD_GROUP );
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(AUDIT_CHANGE_GROUP );
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(BACKUP_RESTORE_GROUP );
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_CHANGE_GROUP );
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_OBJECT_ACCESS_GROUP );
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_OBJECT_CHANGE_GROUP );
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_OBJECT_OWNERSHIP_CHANGE_GROUP );
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_OBJECT_PERMISSION_CHANGE_GROUP );
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_OWNERSHIP_CHANGE_GROUP );
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_OPERATION_GROUP );
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_PERMISSION_CHANGE_GROUP );
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_PRINCIPAL_CHANGE_GROUP );
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_PRINCIPAL_IMPERSONATION_GROUP );
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_ROLE_MEMBER_CHANGE_GROUP );
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DBCC_GROUP );
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(LOGIN_CHANGE_PASSWORD_GROUP );
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SCHEMA_OBJECT_CHANGE_GROUP );
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SCHEMA_OBJECT_OWNERSHIP_CHANGE_GROUP );
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SCHEMA_OBJECT_PERMISSION_CHANGE_GROUP );
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SERVER_OBJECT_CHANGE_GROUP );
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SERVER_OBJECT_OWNERSHIP_CHANGE_GROUP );
```

```
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SERVER_OBJECT_PERMISSION_CHANGE_GROUP );
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SERVER_OPERATION_GROUP );
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SERVER_PERMISSION_CHANGE_GROUP );
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SERVER_PRINCIPAL_IMPERSONATION_GROUP );
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SERVER_ROLE_MEMBER_CHANGE_GROUP );
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SERVER_STATE_CHANGE_GROUP );
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(TRACE_CHANGE_GROUP );
GO

ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION WITH (STATE
= ON);
GO
```

Additional Information:

CCI-003938

Automatically generate audit records of the enforcement actions.

- NIST SP 800-53 Revision 5::CM-5 (1) (b)

CCI-001814

The Information system supports auditing of the enforcement actions.

- NIST SP 800-53 Revision 4::CM-5 (1)

1.55 SQL6-D0-011900 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must disable network functions, ports, protocols, and services deemed by the organization to be nonsecure, in accord with the Ports, Protocols, and Services Management (PPSM) guidance.

```
GROUP ID: V-213990
RULE ID: SV-213990r961470
```

Rationale:

Use of nonsecure network functions, ports, protocols, and services exposes the system to avoidable threats.

Audit:

SQL Server must only use approved network communication libraries, ports, and protocols.

Obtain a list of all approved network libraries, communication ports, and protocols from the server documentation.

Verify that the protocols are enabled for the instance.

If any ports or protocols are used that are not specifically approved in the server documentation, this is a finding.

Remediation:

Assign the approved TCP/IP port number to the SQL Server Database Engine.

1. In SQL Server Configuration Manager, in the console pane, expand SQL Server Network Configuration, expand Protocols for `<instance name>`, and then double-click `TCP/IP`.
2. In the "TCP/IP Properties" dialog box, on the "IP Addresses" tab, several IP addresses appear in the format IP1, IP2, up to IPAll. One of these is for the IP address of the loopback adapter, `127.0.0.1`. Additional IP addresses appear for each IP Address on the computer. (You will probably see both IP version 4 and IP version 6 addresses.) Right-click each address, and then click "Properties" to identify the IP address that you want to configure.
3. If the "TCP Dynamic Ports" dialog box contains "0", indicating the Database Engine is listening on dynamic ports, delete the `0`.
4. In the "IPn Properties area" box, in the "TCP Port" box, type the port number you want this IP address to listen on, and then click "OK".

5. In the console pane, click "SQL Server Services".
6. In the details pane, right-click "SQL Server ()" and then click "Restart", to stop and restart SQL Server.

To disable a server network protocol for an instance:

1. In SQL Server Configuration Manager, in the console pane, expand SQL Server Network Configuration.
2. In the console pane, click "Protocols" for .
3. In the details pane, right-click the protocol you want to change, and then click "Enable" or "Disable".
4. In the console pane, click "SQL Server Services".
5. In the details pane, right-click "SQL Server ()", and then click "Restart", to stop and restart the SQL Server service

Additional Information:

CCI-001762

Disable or remove organization-defined functions, ports, protocols, software, and services within the system deemed to be unnecessary and/or nonsecure.

- NIST SP 800-53 Revision 4::CM-7 (1) (b)
- NIST SP 800-53 Revision 5::CM-7 (1) (b)

1.56 SQL6-D0-012300 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must maintain a separate execution domain for each executing process.

```
GROUP ID: V-213991
RULE ID: SV-213991r961608
```

Rationale:

Database management systems can maintain separate execution domains for each executing process by assigning each process a separate address space.

Each process has a distinct address space so that communication between processes is controlled through the security functions, and one process cannot modify the executing code of another process.

Maintaining separate execution domains for executing processes can be achieved, for example, by implementing separate address spaces.

Audit:

Review the server documentation to determine whether use of CLR assemblies is required. Run the following query to determine whether CLR is enabled for the instance:

```
SELECT name, value, value_in_use
FROM sys.configurations
WHERE name = 'clr enabled'
```

If `value_in_use` is a `1` and CLR is not required, this is a finding.

Remediation:

Disable CLR support in SQL Server by executing the following query:

```
EXEC sp_configure 'clr enabled', 0
GO

RECONFIGURE
GO
```

Additional Information:

CCI-002530

Maintain a separate execution domain for each executing system process.

- NIST SP 800-53 Revision 4::SC-39

- NIST SP 800-53 Revision 5::SC-39

1.57 SQL6-D0-012400 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server services must be configured to run under unique dedicated user accounts.

```
GROUP ID: V-213992
RULE ID: SV-213992r961608
```

Rationale:

Database management systems can maintain separate execution domains for each executing process by assigning each process a separate address space. Each process has a distinct address space so that communication between processes is controlled through the security functions, and one process cannot modify the executing code of another process. Maintaining separate execution domains for executing processes can be achieved, for example, by implementing separate address spaces.

Audit:

Review the server documentation to obtain a listing of required service accounts. Review the accounts configured for all SQL Server services installed on the server.

Click Start >> Type "SQL Server Configuration Manager" >> Launch the program >> Click SQL Server Services tree node. Review the "Log On As" column for each service.

If any services are configured with the same service account or are configured with an account that is not documented and authorized, this is a finding.

Remediation:

Configure SQL Server services to have a documented, dedicated account.

For non-domain servers, consider using virtual service accounts (VSA). See https://msdn.microsoft.com/en-us/library/ms143504.aspx#VA_Desc for more information.

For standalone, domain-joined servers, consider using managed service accounts. See https://msdn.microsoft.com/en-us/library/ms143504.aspx#MSA for more information.

For clustered instances, consider using group managed service accounts. See https://msdn.microsoft.com/en-us/library/ms143504.aspx#GMSA or https://blogs.msdn.microsoft.com/markweberblog/2016/05/25/group-managed-service-accounts-gmsa-and-sql-server-2016/ for more information.

Additional Information:

CCI-002530

Maintain a separate execution domain for each executing system process.

- NIST SP 800-53 Revision 4::SC-39
- NIST SP 800-53 Revision 5::SC-39

1.58 SQL6-D0-012700 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

When updates are applied to SQL Server software, any software components that have been replaced or made unnecessary must be removed.

```
GROUP ID: V-213993
RULE ID: SV-213993r961677
```

Rationale:

Previous versions of DBMS components that are not removed from the information system after updates have been installed may be exploited by adversaries.

Some DBMSs' installation tools may remove older versions of software automatically from the information system. In other cases, manual review and removal will be required. In planning installations and upgrades, organizations must include steps (automated, manual, or both) to identify and remove the outdated modules.

A transition period may be necessary when both the old and the new software are required. This should be taken into account in the planning.

Audit:

From the server documentation, obtain a listing of required components.

Generate a listing of components installed on the server.

Click Start >> Type "SQL Server 2016 Installation Center" >> Launch the program >> Click Tools >> Click "Installed SQL Server features discovery report"

Compare the feature listing against the required components listing. If any features are installed, but are not required, this is a finding.

Remediation:

Remove all features that are not required.

Additional Information:

CCI-002617

Remove previous versions of organization-defined software components after updated versions have been installed.

- NIST SP 800-53 Revision 4::SI-2 (6)
- NIST SP 800-53 Revision 5::SI-2 (6)

1.59 SQL6-D0-012800 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

Security-relevant software updates to SQL Server must be installed within the time period directed by an authoritative source (e.g. IAVM, CTOs, DTMs, and STIGs).

```
GROUP ID: V-213994
RULE ID: SV-213994r1001008
```

Rationale:

Security flaws with software applications, including database management systems, are discovered daily. Vendors are constantly updating and patching their products to address newly discovered security vulnerabilities. Organizations (including any contractor to the organization) are required to promptly install security-relevant software updates (e.g., patches, service packs, and hot fixes). Flaws discovered during security assessments, continuous monitoring, incident response activities, or information system error handling must also be addressed expeditiously.

Organization-defined time periods for updating security-relevant software may vary based on a variety of factors including, for example, the security category of the information system or the criticality of the update (i.e., severity of the vulnerability related to the discovered flaw).

This requirement will apply to software patch management solutions that are used to install patches across the enclave and also to applications themselves that are not part of that patch management solution. For example, many browsers today provide the capability to install their own patch software. Patch criticality, as well as system criticality, will vary. Therefore, the tactical situations regarding the patch management process will also vary. This means that the time period utilized must be a configurable parameter. Time frames for application of security-relevant software updates may be dependent upon the Information Assurance Vulnerability Management (IAVM) process.

SQL Server will be configured to check for and install security-relevant software updates within an identified time period from the availability of the update. The specific time period will be defined by an authoritative source (e.g. IAVM, CTOs, DTMs, and STIGs).

Audit:

Obtain evidence that software patches are consistently applied to SQL Server within the time frame defined for each patch. To be considered supported, Microsoft must report that the version is supported by security patches to known vulnerability. Review the Support dates at: https://learn.microsoft.com/en-us/troubleshoot/sql/releases/download-and-install-latest-updates

Check the SQL Server version by running the following script: `Print @@version`

If the SQL Server version is not shown as supported, this is a finding.

If such evidence cannot be obtained, or the evidence that is obtained indicates a pattern of noncompliance, this is a finding.

Remediation:

Upgrade SQL Server to the Microsoft-supported version. Institute and adhere to policies and procedures to ensure that patches are consistently applied to SQL Server within the time allowed.

Additional Information:

CCI-002605

Install security-relevant software updates within an organization-defined time period of the release of the updates.

- NIST SP 800-53 Revision 4::SI-2 c
- NIST SP 800-53 Revision 5::SI-2 c

1.60 SQL6-D0-012900 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must be able to generate audit records when successful and unsuccessful attempts to access security objects occur.

```
GROUP ID: V-213995
RULE ID: SV-213995r961791
```

Rationale:

Changes to the security configuration must be tracked.

This requirement applies to situations where security data is retrieved or modified via data manipulation operations, as opposed to via specialized security functionality.

In an SQL environment, types of access include, but are not necessarily limited to:

- SELECT
- INSERT
- UPDATE
- DELETE
- EXECUTE

To aid in diagnosis, it is necessary to keep track of failed attempts in addition to the successful ones.

Satisfies: SRG-APP-000492-DB-000332, SRG-APP-000492-DB-000333

Audit:

Review the system documentation to determine if SQL Server is required to audit the retrieval of when security objects are accessed.

If this is not required, this is not a finding.

If the documentation does not exist, this is a finding.

Determine if an audit is configured and started by executing the following query.

```
SELECT name AS 'Audit Name',
status_desc AS 'Audit Status',
audit_file_path AS 'Current Audit File'
FROM sys.dm_server_audit_status
```

If no records are returned, this is a finding.

Execute the following query to verify the SCHEMA_OBJECT_ACCESS_GROUP is included in the server audit specification.

```
SELECT a.name AS 'AuditName',
s.name AS 'SpecName',
d.audit_action_name AS 'ActionName',
d.audited_result AS 'Result'
FROM sys.server_audit_specifications s
JOIN sys.server_audits a ON s.audit_guid = a.audit_guid
JOIN sys.server_audit_specification_details d ON s.server_specification_id =
d.server_specification_id
WHERE a.is_state_enabled = 1 AND d.audit_action_name =
'SCHEMA_OBJECT_ACCESS_GROUP'
```

If the SCHEMA_OBJECT_ACCESS_GROUP is not returned in an active audit, this is a finding.

Remediation:

Deploy an audit to audit the retrieval of privilege/permission/role membership information when successful and unsuccessful attempts to access security objects occur.

See the supplemental file SQL 2016 Audit.sql.

Additional Information:

CCI-000172

Generate audit records for the event types defined in AU-2 c that include the audit record content defined in AU-3.

- NIST SP 800-53::AU-12 c
- NIST SP 800-53A::AU-12.1 (iv)
- NIST SP 800-53 Revision 4::AU-12 c
- NIST SP 800-53 Revision 5::AU-12 c

1.61 SQL6-D0-013200 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must generate audit records when successful and unsuccessful attempts to access categorized information (e.g., classification levels/security levels) occur.

```
GROUP ID: V-213998
RULE ID: SV-213998r961797
```

Rationale:

Changes in categorized information must be tracked. Without an audit trail, unauthorized access to protected data could go undetected.

For detailed information on categorizing information, refer to FIPS Publication 199, Standards for Security Categorization of Federal Information and Information Systems, and FIPS Publication 200, Minimum Security Requirements for Federal Information and Information Systems.

To aid in diagnosis, it is necessary to keep track of failed attempts in addition to the successful ones.

Satisfies: SRG-APP-000494-DB-000344

Audit:

Review the system documentation to determine if SQL Server is required to audit when data classifications are both successfully and unsuccessfully retrieved.

If this is not required, this is not a finding.

If the documentation does not exist, this is a finding.

Determine if an audit is configured and started by executing the following query.

```
SELECT name AS 'Audit Name',
  status_desc AS 'Audit Status',
  audit_file_path AS 'Current Audit File'
FROM sys.dm_server_audit_status
```

If no records are returned, this is a finding.

If the auditing the retrieval of privilege/permission/role membership information is required, execute the following query to verify the SCHEMA_OBJECT_ACCESS_GROUP is included in the server audit specification.

```
SELECT a.name AS 'AuditName',
  s.name AS 'SpecName',
  d.audit_action_name AS 'ActionName',
```

```
  d.audited_result AS 'Result'
FROM sys.server_audit_specifications s
JOIN sys.server_audits a ON s.audit_guid = a.audit_guid
JOIN sys.server_audit_specification_details d ON s.server_specification_id =
d.server_specification_id
WHERE a.is_state_enabled = 1 AND d.audit_action_name =
'SCHEMA_OBJECT_ACCESS_GROUP
```

If the SCHEMA_OBJECT_ACCESS_GROUP is not returned in an active audit, this is a finding.

Remediation:

Deploy an audit to audit when data classifications are both successfully and unsuccessfully retrieved. See the supplemental file SQL 2016 Audit.sql.

Additional Information:

CCI-000172

Generate audit records for the event types defined in AU-2 c that include the audit record content defined in AU-3.

- NIST SP 800-53::AU-12 c
- NIST SP 800-53A::AU-12.1 (iv)
- NIST SP 800-53 Revision 4::AU-12 c
- NIST SP 800-53 Revision 5::AU-12 c

1.62 SQL6-D0-013400 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must generate audit records when successful and unsuccessful attempts to add privileges/permissions occur.

```
GROUP ID: V-214000
RULE ID: SV-214000r961800
```

Rationale:

Changes in the permissions, privileges, and roles granted to users and roles must be tracked. Without an audit trail, unauthorized elevation or restriction of privileges could go undetected. Elevated privileges give users access to information and functionality that they should not have; restricted privileges wrongly deny access to authorized users.

In an SQL environment, adding permissions is typically done via the GRANT command, or, in the negative, the DENY command.

To aid in diagnosis, it is necessary to keep track of failed attempts in addition to the successful ones.

Satisfies: SRG-APP-000495-DB-000326

Audit:

Check that SQL Server Audit is being used for the STIG compliant audit.

Determine if an audit is configured and started by executing the following query:

```
SELECT name AS 'Audit Name',
status_desc AS 'Audit Status',
audit_file_path AS 'Current Audit File'
FROM sys.dm_server_audit_status
```

Execute the following query to verify the required audit actions are included in the server audit specification:

```
SELECT a.name AS 'AuditName',
s.name AS 'SpecName',
d.audit_action_name AS 'ActionName',
d.audited_result AS 'Result'
FROM sys.server_audit_specifications s
JOIN sys.server_audits a ON s.audit_guid = a.audit_guid
JOIN sys.server_audit_specification_details d ON s.server_specification_id =
d.server_specification_id
WHERE a.is_state_enabled = 1
AND d.audit_action_name IN ('DATABASE_OBJECT_OWNERSHIP_CHANGE_GROUP'
,'DATABASE_OBJECT_PERMISSION_CHANGE_GROUP'
```

```
,'DATABASE_OWNERSHIP_CHANGE_GROUP'
,'DATABASE_PERMISSION_CHANGE_GROUP'
,'DATABASE_ROLE_MEMBER_CHANGE_GROUP'
,'SCHEMA_OBJECT_OWNERSHIP_CHANGE_GROUP'
,'SCHEMA_OBJECT_PERMISSION_CHANGE_GROUP'
,'SERVER_OBJECT_OWNERSHIP_CHANGE_GROUP'
,'SERVER_OBJECT_PERMISSION_CHANGE_GROUP'
,'SERVER_PERMISSION_CHANGE_GROUP'
,'SERVER_ROLE_MEMBER_CHANGE_GROUP')
```

If the any of the following audit actions are not returned in an active audit, this is a finding.

```
DATABASE_OBJECT_OWNERSHIP_CHANGE_GROUP
DATABASE_OBJECT_PERMISSION_CHANGE_GROUP
DATABASE_OWNERSHIP_CHANGE_GROUP
DATABASE_PERMISSION_CHANGE_GROUP
DATABASE_ROLE_MEMBER_CHANGE_GROUP
SCHEMA_OBJECT_OWNERSHIP_CHANGE_GROUP
SCHEMA_OBJECT_PERMISSION_CHANGE_GROUP
SERVER_OBJECT_OWNERSHIP_CHANGE_GROUP
SERVER_OBJECT_PERMISSION_CHANGE_GROUP
SERVER_PERMISSION_CHANGE_GROUP
SERVER_ROLE_MEMBER_CHANGE_GROUP
```

If no records are returned, this is a finding.

Remediation:

Add the following events to the SQL Server Audit that is being used for the STIG compliant audit.

```
DATABASE_OBJECT_OWNERSHIP_CHANGE_GROUP
DATABASE_OBJECT_PERMISSION_CHANGE_GROUP
DATABASE_OWNERSHIP_CHANGE_GROUP
DATABASE_PERMISSION_CHANGE_GROUP
DATABASE_ROLE_MEMBER_CHANGE_GROUP
SCHEMA_OBJECT_OWNERSHIP_CHANGE_GROUP
SCHEMA_OBJECT_PERMISSION_CHANGE_GROUP
SERVER_OBJECT_OWNERSHIP_CHANGE_GROUP
SERVER_OBJECT_PERMISSION_CHANGE_GROUP
SERVER_PERMISSION_CHANGE_GROUP
SERVER_ROLE_MEMBER_CHANGE_GROUP
```

See the supplemental file `SQL 2016 Audit.sql`.

Reference: https://msdn.microsoft.com/en-us/library/cc280663.aspx

Additional Information:

CCI-000172

Generate audit records for the event types defined in AU-2 c that include the audit record content defined in AU-3.

- NIST SP 800-53::AU-12 c

- NIST SP 800-53A::AU-12.1 (iv)
- NIST SP 800-53 Revision 4::AU-12 c
- NIST SP 800-53 Revision 5::AU-12 c

1.63 SQL6-D0-013600 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must generate audit records when successful and unsuccessful attempts to modify privileges/permissions occur.

```
GROUP ID: V-214002
RULE ID: SV-214002r961800
```

Rationale:

Changes in the permissions, privileges, and roles granted to users and roles must be tracked. Without an audit trail, unauthorized elevation or restriction of privileges could go undetected. Elevated privileges give users access to information and functionality that they should not have; restricted privileges wrongly deny access to authorized users.

In an SQL environment, modifying permissions is typically done via the GRANT, REVOKE, and DENY commands.

To aid in diagnosis, it is necessary to keep track of failed attempts in addition to the successful ones.

Satisfies: SRG-APP-000495-DB-000328

Audit:

Check that SQL Server Audit is being used for the STIG compliant audit.

Determine if an audit is configured and started by executing the following query:

```
SELECT name AS 'Audit Name',
status_desc AS 'Audit Status',
audit_file_path AS 'Current Audit File'
FROM sys.dm_server_audit_status
```

Execute the following query to verify the required audit actions are included in the server audit specification:

```
SELECT a.name AS 'AuditName',
s.name AS 'SpecName',
d.audit_action_name AS 'ActionName',
d.audited_result AS 'Result'
FROM sys.server_audit_specifications s
JOIN sys.server_audits a ON s.audit_guid = a.audit_guid
JOIN sys.server_audit_specification_details d ON s.server_specification_id =
d.server_specification_id
WHERE a.is_state_enabled = 1
AND d.audit_action_name IN ('DATABASE_OBJECT_OWNERSHIP_CHANGE_GROUP'
,'DATABASE_OBJECT_PERMISSION_CHANGE_GROUP'
```

```
,'DATABASE_OWNERSHIP_CHANGE_GROUP'
,'DATABASE_PERMISSION_CHANGE_GROUP'
,'DATABASE_ROLE_MEMBER_CHANGE_GROUP'
,'SCHEMA_OBJECT_OWNERSHIP_CHANGE_GROUP'
,'SCHEMA_OBJECT_PERMISSION_CHANGE_GROUP'
,'SERVER_OBJECT_OWNERSHIP_CHANGE_GROUP'
,'SERVER_OBJECT_PERMISSION_CHANGE_GROUP'
,'SERVER_PERMISSION_CHANGE_GROUP'
,'SERVER_ROLE_MEMBER_CHANGE_GROUP')
```

If the any of the following audit actions are not returned in an active audit, this is a finding.

```
DATABASE_OBJECT_OWNERSHIP_CHANGE_GROUP
DATABASE_OBJECT_PERMISSION_CHANGE_GROUP
DATABASE_OWNERSHIP_CHANGE_GROUP
DATABASE_PERMISSION_CHANGE_GROUP
DATABASE_ROLE_MEMBER_CHANGE_GROUP
SCHEMA_OBJECT_OWNERSHIP_CHANGE_GROUP
SCHEMA_OBJECT_PERMISSION_CHANGE_GROUP
SERVER_OBJECT_OWNERSHIP_CHANGE_GROUP
SERVER_OBJECT_PERMISSION_CHANGE_GROUP
SERVER_PERMISSION_CHANGE_GROUP
SERVER_ROLE_MEMBER_CHANGE_GROUP
```

If no records are returned, this is a finding.

Remediation:

Add the following events to the SQL Server Audit that is being used for the STIG compliant audit.

```
DATABASE_OBJECT_OWNERSHIP_CHANGE_GROUP
DATABASE_OBJECT_PERMISSION_CHANGE_GROUP
DATABASE_OWNERSHIP_CHANGE_GROUP
DATABASE_PERMISSION_CHANGE_GROUP
DATABASE_ROLE_MEMBER_CHANGE_GROUP
SCHEMA_OBJECT_OWNERSHIP_CHANGE_GROUP
SCHEMA_OBJECT_PERMISSION_CHANGE_GROUP
SERVER_OBJECT_OWNERSHIP_CHANGE_GROUP
SERVER_OBJECT_PERMISSION_CHANGE_GROUP
SERVER_PERMISSION_CHANGE_GROUP
SERVER_ROLE_MEMBER_CHANGE_GROUP
```

See the supplemental file `SQL 2016 Audit.sql`.

Reference:

https://msdn.microsoft.com/en-us/library/cc280663.aspx

Additional Information:

CCI-000172

Generate audit records for the event types defined in AU-2 c that include the audit record content defined in AU-3.

- NIST SP 800-53::AU-12 c
- NIST SP 800-53A::AU-12.1 (iv)
- NIST SP 800-53 Revision 4::AU-12 c
- NIST SP 800-53 Revision 5::AU-12 c

1.64 SQL6-D0-013800 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must generate audit records when successful and unsuccessful attempts to modify security objects occur.

```
GROUP ID: V-214004
RULE ID: SV-214004r961803
```

Rationale:

Changes in the database objects (tables, views, procedures, functions) that record and control permissions, privileges, and roles granted to users and roles must be tracked. Without an audit trail, unauthorized changes to the security subsystem could go undetected. The database could be severely compromised or rendered inoperative.

To aid in diagnosis, it is necessary to keep track of failed attempts in addition to the successful ones.

Satisfies: SRG-APP-000496-DB-000334

Audit:

Determine if an audit is configured and started by executing the following query:

```
SELECT name AS 'Audit Name',
  status_desc AS 'Audit Status',
  audit_file_path AS 'Current Audit File'
FROM sys.dm_server_audit_status
```

If no records are returned, this is a finding.

Execute the following query to verify the SCHEMA_OBJECT_CHANGE_GROUP is included in the server audit specification.

```
SELECT a.name AS 'AuditName',
  s.name AS 'SpecName',
  d.audit_action_name AS 'ActionName',
  d.audited_result AS 'Result'
FROM sys.server_audit_specifications s
JOIN sys.server_audits a ON s.audit_guid = a.audit_guid
JOIN sys.server_audit_specification_details d ON s.server_specification_id =
d.server_specification_id
WHERE a.is_state_enabled = 1 AND d.audit_action_name =
'SCHEMA_OBJECT_CHANGE_GROUP'
```

If the "SCHEMA_OBJECT_CHANGE_GROUP" is not returned in an active audit, this is a finding.

Remediation:

Add the SCHEMA_OBJECT_CHANGE_GROUP to the server audit specification

```
USE [master];
GO

ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION WITH (STATE
= OFF);
GO

ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SCHEMA_OBJECT_CHANGE_GROUP);
GO

ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION WITH (STATE
= ON);
GO
```

See supplemental script SQL 2016 Audit.sql.

Additional Information:

CCI-000172

Generate audit records for the event types defined in AU-2 c that include the audit record content defined in AU-3.

- NIST SP 800-53::AU-12 c
- NIST SP 800-53A::AU-12.1 (iv)
- NIST SP 800-53 Revision 4::AU-12 c
- NIST SP 800-53 Revision 5::AU-12 c

1.65 SQL6-D0-014000 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must generate audit records when successful and unsuccessful attempts to modify categorized information (e.g., classification levels/security levels) occur.

```
GROUP ID: V-214006
RULE ID: SV-214006r961809
```

Rationale:

Changes in categories of information must be tracked. Without an audit trail, unauthorized access to protected data could go undetected.

To aid in diagnosis, it is necessary to keep track of failed attempts in addition to the successful ones.

For detailed information on categorizing information, refer to FIPS Publication 199, Standards for Security Categorization of Federal Information and Information Systems, and FIPS Publication 200, Minimum Security Requirements for Federal Information and Information Systems.

Satisfies: SRG-APP-000498-DB-000346

Audit:

Review the system documentation to determine if SQL Server is required to audit when data classifications are successfully and unsuccessfully modified.

If this is not required, this is not a finding.

If the documentation does not exist, this is a finding.

Determine if an audit is configured and started by executing the following query:

```
SELECT name AS 'Audit Name',
  status_desc AS 'Audit Status',
  audit_file_path AS 'Current Audit File'
FROM  sys.dm_server_audit_status
```

If no records are returned, this is a finding.

If the auditing the retrieval of privilege/permission/role membership information is required, execute the following query to verify the SCHEMA_OBJECT_ACCESS_GROUP is included in the server audit specification.

```
SELECT a.name AS 'AuditName',
  s.name AS 'SpecName',
  d.audit_action_name AS 'ActionName',
```

```
  d.audited_result AS 'Result'
FROM sys.server_audit_specifications s
JOIN sys.server_audits a ON s.audit_guid = a.audit_guid
JOIN sys.server_audit_specification_details d ON s.server_specification_id =
d.server_specification_id
WHERE a.is_state_enabled = 1 AND d.audit_action_name =
'SCHEMA_OBJECT_ACCESS_GROUP'
```

If the SCHEMA_OBJECT_ACCESS_GROUP is not returned in an active audit, this is a finding.

Remediation:

Deploy an audit to audit when data classifications are unsuccessfully modified. See the supplemental file SQL 2016 Audit.sql.

Additional Information:

CCI-000172

Generate audit records for the event types defined in AU-2 c that include the audit record content defined in AU-3.

- NIST SP 800-53::AU-12 c
- NIST SP 800-53A::AU-12.1 (iv)
- NIST SP 800-53 Revision 4::AU-12 c
- NIST SP 800-53 Revision 5::AU-12 c

1.66 SQL6-D0-014200 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must generate audit records when successful and unsuccessful attempts to delete privileges/permissions occur.

```
GROUP ID: V-214008
RULE ID: SV-214008r961812
```

Rationale:

Changes in the permissions, privileges, and roles granted to users and roles must be tracked. Without an audit trail, unauthorized elevation or restriction of privileges could go undetected. Elevated privileges give users access to information and functionality that they should not have; restricted privileges wrongly deny access to authorized users.

In an SQL environment, deleting permissions is typically done via the REVOKE or DENY command.

To aid in diagnosis, it is necessary to keep track of failed attempts in addition to the successful ones.

Audit:

Check that SQL Server Audit is being used for the STIG compliant audit. Determine if an audit is configured and started by executing the following query. If no records are returned, this is a finding.

```
SELECT name AS 'Audit Name',
status_desc AS 'Audit Status',
audit_file_path AS 'Current Audit File'
FROM sys.dm_server_audit_status
```

Execute the following query to verify the required audit actions are included in the server audit specification:

```
SELECT a.name AS 'AuditName',
s.name AS 'SpecName',
d.audit_action_name AS 'ActionName',
d.audited_result AS 'Result'
FROM sys.server_audit_specifications s
JOIN sys.server_audits a ON s.audit_guid = a.audit_guid
JOIN sys.server_audit_specification_details d ON s.server_specification_id =
d.server_specification_id
WHERE a.is_state_enabled = 1
AND d.audit_action_name IN ('DATABASE_OBJECT_OWNERSHIP_CHANGE_GROUP'
,'DATABASE_OBJECT_PERMISSION_CHANGE_GROUP'
,'DATABASE_OWNERSHIP_CHANGE_GROUP'
,'DATABASE_PERMISSION_CHANGE_GROUP'
```

```
,'DATABASE_ROLE_MEMBER_CHANGE_GROUP'
,'SCHEMA_OBJECT_OWNERSHIP_CHANGE_GROUP'
,'SCHEMA_OBJECT_PERMISSION_CHANGE_GROUP'
,'SERVER_OBJECT_OWNERSHIP_CHANGE_GROUP'
,'SERVER_OBJECT_PERMISSION_CHANGE_GROUP'
,'SERVER_PERMISSION_CHANGE_GROUP'
,'SERVER_ROLE_MEMBER_CHANGE_GROUP')
```

If the any of the following audit actions are not returned in an active audit, this is a finding.

```
DATABASE_OBJECT_OWNERSHIP_CHANGE_GROUP
DATABASE_OBJECT_PERMISSION_CHANGE_GROUP
DATABASE_OWNERSHIP_CHANGE_GROUP
DATABASE_PERMISSION_CHANGE_GROUP
DATABASE_ROLE_MEMBER_CHANGE_GROUP
SCHEMA_OBJECT_OWNERSHIP_CHANGE_GROUP
SCHEMA_OBJECT_PERMISSION_CHANGE_GROUP
SERVER_OBJECT_OWNERSHIP_CHANGE_GROUP
SERVER_OBJECT_PERMISSION_CHANGE_GROUP
SERVER_PERMISSION_CHANGE_GROUP
SERVER_ROLE_MEMBER_CHANGE_GROUP
```

Reference: https://msdn.microsoft.com/en-us/library/cc280663.aspx

Remediation:

Add the following events to the SQL Server Audit that is being used for the STIG compliant audit.

```
DATABASE_OBJECT_OWNERSHIP_CHANGE_GROUP
DATABASE_OBJECT_PERMISSION_CHANGE_GROUP
DATABASE_OWNERSHIP_CHANGE_GROUP
DATABASE_PERMISSION_CHANGE_GROUP
DATABASE_ROLE_MEMBER_CHANGE_GROUP
SCHEMA_OBJECT_OWNERSHIP_CHANGE_GROUP
SCHEMA_OBJECT_PERMISSION_CHANGE_GROUP
SERVER_OBJECT_OWNERSHIP_CHANGE_GROUP
SERVER_OBJECT_PERMISSION_CHANGE_GROUP
SERVER_PERMISSION_CHANGE_GROUP
SERVER_ROLE_MEMBER_CHANGE_GROUP
```

See the supplemental file `SQL 2016 Audit.sql`.

Reference: https://msdn.microsoft.com/en-us/library/cc280663.aspx

Additional Information:

CCI-000172

Generate audit records for the event types defined in AU-2 c that include the audit record content defined in AU-3.

- NIST SP 800-53::AU-12 c
- NIST SP 800-53A::AU-12.1 (iv)
- NIST SP 800-53 Revision 4::AU-12 c

- NIST SP 800-53 Revision 5::AU-12 c

1.67 SQL6-D0-014400 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must generate audit records when successful and unsuccessful attempts to delete security objects occur.

```
GROUP ID: V-214010
RULE ID: SV-214010r961818
```

Rationale:

The removal of security objects from the database/DBMS would seriously degrade a system's information assurance posture. If such an action is attempted, it must be logged.

To aid in diagnosis, it is necessary to keep track of failed attempts in addition to the successful ones.

Satisfies: SRG-APP-000501-DB-000336

Audit:

Determine if an audit is configured and started by executing the following query:

```
SELECT name AS 'Audit Name',
status_desc AS 'Audit Status',
audit_file_path AS 'Current Audit File'
FROM sys.dm_server_audit_status
```

If no records are returned, this is a finding.

Execute the following query to verify the SCHEMA_OBJECT_CHANGE_GROUP is included in the server audit specification.

```
SELECT a.name AS 'AuditName',
s.name AS 'SpecName',
d.audit_action_name AS 'ActionName',
d.audited_result AS 'Result'
FROM sys.server_audit_specifications s
JOIN sys.server_audits a ON s.audit_guid = a.audit_guid
JOIN sys.server_audit_specification_details d ON s.server_specification_id =
d.server_specification_id
WHERE a.is_state_enabled = 1 AND d.audit_action_name =
'SCHEMA_OBJECT_CHANGE_GROUP'
```

If the SCHEMA_OBJECT_CHANGE_GROUP is not returned in an active audit, this is a finding.

Remediation:

Add the SCHEMA_OBJECT_CHANGE_GROUP to the server audit specification

```
USE [master];
GO

ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION WITH (STATE
= OFF);
GO

ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SCHEMA_OBJECT_CHANGE_GROUP);
GO

ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION WITH (STATE
= ON);
GO
```

See the supplemental script "SQL 2016 Audit.sql" for complete script.

Additional Information:

CCI-000172

Generate audit records for the event types defined in AU-2 c that include the audit record content defined in AU-3.

- NIST SP 800-53::AU-12 c
- NIST SP 800-53A::AU-12.1 (iv)
- NIST SP 800-53 Revision 4::AU-12 c
- NIST SP 800-53 Revision 5::AU-12 c

1.68 SQL6-D0-014600 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must generate audit records when successful and unsuccessful attempts to delete categorized information (e.g., classification levels/security levels) occur.

```
GROUP ID: V-214012
RULE ID: SV-214012r1018597
```

Rationale:

Changes in categorized information must be tracked. Without an audit trail, unauthorized access to protected data could go undetected.

To aid in diagnosis, it is necessary to keep track of failed attempts in addition to the successful ones.

For detailed information on categorizing information, refer to FIPS Publication 199, Standards for Security Categorization of Federal Information and Information Systems, and FIPS Publication 200, Minimum Security Requirements for Federal Information and Information Systems.

Satisfies: SRG-APP-000502-DB-000348

Audit:

Review the system documentation to determine if SQL Server is required to audit when data classifications are successfully and unsuccessfully deleted.

If this is not required, this is not a finding.

If the documentation does not exist, this is a finding.

Determine if an audit is configured and started by executing the following query:

```
SELECT name AS 'Audit Name',
  status_desc AS 'Audit Status',
  audit_file_path AS 'Current Audit File'
FROM  sys.dm_server_audit_status
```

If no records are returned, this is a finding.

If the auditing the retrieval of privilege/permission/role membership information is required, execute the following query to verify the SCHEMA_OBJECT_ACCESS_GROUP is included in the server audit specification.

```
SELECT a.name AS 'AuditName',
 s.name AS 'SpecName',
 d.audit_action_name AS 'ActionName',
```

```
 d.audited_result AS 'Result'
FROM sys.server_audit_specifications s
JOIN sys.server_audits a ON s.audit_guid = a.audit_guid
JOIN sys.server_audit_specification_details d ON s.server_specification_id =
d.server_specification_id
WHERE a.is_state_enabled = 1 AND d.audit_action_name =
'SCHEMA_OBJECT_ACCESS_GROUP'
```

If the SCHEMA_OBJECT_ACCESS_GROUP is not returned in an active audit, this is a finding.

Remediation:

Deploy an audit to audit when data classifications are unsuccessfully deleted. See the supplemental file "SQL 2016 Audit.sql".

Additional Information:

CCI-000172

Generate audit records for the event types defined in AU-2 c that include the audit record content defined in AU-3.

- NIST SP 800-53::AU-12 c
- NIST SP 800-53A::AU-12.1 (iv)
- NIST SP 800-53 Revision 4::AU-12 c
- NIST SP 800-53 Revision 5::AU-12 c

1.69 SQL6-D0-014800 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must generate audit records when successful and unsuccessful logons or connection attempts occur.

```
GROUP ID: V-214014
RULE ID: SV-214014r961824
```

Rationale:

For completeness of forensic analysis, it is necessary to track who/what (a user or other principal) logs on to SQL Server. It is also necessary to track failed attempts to log on to SQL Server. While positive identification may not be possible in a case of failed authentication, as much information as possible about the incident must be captured.

Satisfies: SRG-APP-000503-DB-000350

Audit:

Determine if an audit is configured and started by executing the following query:

```
SELECT name AS 'Audit Name',
  status_desc AS 'Audit Status',
  audit_file_path AS 'Current Audit File'
FROM sys.dm_server_audit_status
```

Execute the following query to verify the SUCCESSFUL_LOGIN_GROUP and FAILED_LOGIN_GROUP are included in the server audit specification.

```
SELECT a.name AS 'AuditName',
  s.name AS 'SpecName',
 d.audit_action_name AS 'ActionName',
  d.audited_result AS 'Result'
FROM sys.server_audit_specifications s
JOIN sys.server_audits a ON s.audit_guid = a.audit_guid
JOIN sys.server_audit_specification_details d ON s.server_specification_id =
d.server_specification_id
WHERE a.is_state_enabled = 1 AND d.audit_action_name IN
('SUCCESSFUL_LOGIN_GROUP', 'FAILED_LOGIN_GROUP')
```

If both SUCCESSFUL_LOGIN_GROUP and FAILED_LOGIN_GROUP are returned in an active audit, this is not a finding.

If both SUCCESSFUL_LOGIN_GROUP and FAILED_LOGIN_GROUP are not in the active audit, determine whether "Both failed and successful logins" is enabled.

In SQL Management Studio

1. Right-click on the instance
2. >> Select "Properties"
3. >> Select "Security" on the left hand side
4. >> Check the setting for "Login auditing"

If "Both failed and successful logins" is not selected, this is a finding.

Remediation:

Add both SUCCESSFUL_LOGIN_GROUP and FAILED_LOGIN_GROUP to the server audit specification.

```
USE [master];
GO

ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION WITH (STATE
= OFF);
GO

ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SUCCESSFUL_LOGIN_GROUP);
GO

ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(FAILED_LOGIN_GROUP);
GO

ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION WITH (STATE
= ON);
GO
```

Alternatively, enable "Both failed and successful logins".

In SQL Management Studio: Right-click on the instance.

- Select "Properties".
- Select "Security" on the left-hand side.
- Select "Both failed and successful logins".
- Click "OK".

Additional Information:

CCI-000172

Generate audit records for the event types defined in AU-2 c that include the audit record content defined in AU-3.

- NIST SP 800-53::AU-12 c
- NIST SP 800-53A::AU-12.1 (iv)
- NIST SP 800-53 Revision 4::AU-12 c
- NIST SP 800-53 Revision 5::AU-12 c

1.70 SQL6-D0-014900 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must generate audit records for all privileged activities or other system-level access.

```
GROUP ID: V-214015
RULE ID: SV-214015r961827
```

Rationale:

Without tracking privileged activity, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

System documentation should include a definition of the functionality considered privileged.

A privileged function in this context is any operation that modifies the structure of the database, its built-in logic, or its security settings. This would include all Data Definition Language (DDL) statements and all security-related statements. In an SQL environment, it encompasses, but is not necessarily limited to:

- `CREATE`
- `ALTER`
- `DROP`
- `GRANT`
- `REVOKE`
- `DENY`

There may also be Data Manipulation Language (DML) statements that, subject to context, should be regarded as privileged. Possible examples in SQL include:\

- `TRUNCATE TABLE;`
- `DELETE`, or
- `DELETE` affecting more than *n* rows, for some *n*, or
- `DELETE` without a `WHERE` clause;
- `UPDATE` or
- `UPDATE` affecting more than *n* rows, for some *n*, or
- `UPDATE` without a `WHERE` clause;

any `SELECT`, `INSERT`, `UPDATE`, or `DELETE` to an application-defined security table executed by other than a security principal.

Depending on the capabilities of SQL Server and the design of the database and associated applications, audit logging may be achieved by means of DBMS auditing features, database triggers, other mechanisms, or a combination of these.

Note that it is particularly important to audit, and tightly control, any action that weakens the implementation of this requirement itself, since the objective is to have a complete audit trail of all administrative activity.

Audit:

Determine if an audit is configured and started by executing the following query:

```
SELECT name AS 'Audit Name',
status_desc AS 'Audit Status',
audit_file_path AS 'Current Audit File'
FROM sys.dm_server_audit_status
```

If no records are returned, this is a finding.

Execute the following query to verify the following events are included in the server audit specification:

```
APPLICATION_ROLE_CHANGE_PASSWORD_GROUP
AUDIT_CHANGE_GROUP
BACKUP_RESTORE_GROUP
DATABASE_CHANGE_GROUP
DATABASE_OBJECT_CHANGE_GROUP
DATABASE_OBJECT_OWNERSHIP_CHANGE_GROUP
DATABASE_OBJECT_PERMISSION_CHANGE_GROUP
DATABASE_OPERATION_GROUP
DATABASE_OWNERSHIP_CHANGE_GROUP
DATABASE_PERMISSION_CHANGE_GROUP
DATABASE_PRINCIPAL_CHANGE_GROUP
DATABASE_PRINCIPAL_IMPERSONATION_GROUP
DATABASE_ROLE_MEMBER_CHANGE_GROUP
DBCC_GROUP
LOGIN_CHANGE_PASSWORD_GROUP
SCHEMA_OBJECT_CHANGE_GROUP
SCHEMA_OBJECT_OWNERSHIP_CHANGE_GROUP
SCHEMA_OBJECT_PERMISSION_CHANGE_GROUP
SERVER_OBJECT_CHANGE_GROUP
SERVER_OBJECT_OWNERSHIP_CHANGE_GROUP
SERVER_OBJECT_PERMISSION_CHANGE_GROUP
SERVER_OPERATION_GROUP
SERVER_PERMISSION_CHANGE_GROUP
SERVER_PRINCIPAL_CHANGE_GROUP
SERVER_PRINCIPAL_IMPERSONATION_GROUP
SERVER_ROLE_MEMBER_CHANGE_GROUP
SERVER_STATE_CHANGE_GROUP
TRACE_CHANGE_GROUP
USER_CHANGE_PASSWORD_GROUP

SELECT a.name AS 'AuditName',
s.name AS 'SpecName',
d.audit_action_name AS 'ActionName',
d.audited_result AS 'Result'
```

```
FROM sys.server_audit_specifications s
JOIN sys.server_audits a ON s.audit_guid = a.audit_guid
JOIN sys.server_audit_specification_details d ON s.server_specification_id =
d.server_specification_id
WHERE a.is_state_enabled = 1
AND d.audit_action_name IN
('APPLICATION_ROLE_CHANGE_PASSWORD_GROUP','AUDIT_CHANGE_GROUP','BACKUP_RESTOR
E_GROUP','DATABASE_CHANGE_GROUP','DATABASE_OBJECT_CHANGE_GROUP','DATABASE_OBJ
ECT_OWNERSHIP_CHANGE_GROUP','DATABASE_OBJECT_PERMISSION_CHANGE_GROUP','DATABA
SE_OPERATION_GROUP','DATABASE_OWNERSHIP_CHANGE_GROUP','DATABASE_PERMISSION_CH
ANGE_GROUP','DATABASE_PRINCIPAL_CHANGE_GROUP','DATABASE_PRINCIPAL_IMPERSONATI
ON_GROUP','DATABASE_ROLE_MEMBER_CHANGE_GROUP','DBCC_GROUP','LOGIN_CHANGE_PASS
WORD_GROUP','SCHEMA_OBJECT_CHANGE_GROUP','SCHEMA_OBJECT_OWNERSHIP_CHANGE_GROU
P','SCHEMA_OBJECT_PERMISSION_CHANGE_GROUP','SERVER_OBJECT_CHANGE_GROUP','SERV
ER_OBJECT_OWNERSHIP_CHANGE_GROUP','SERVER_OBJECT_PERMISSION_CHANGE_GROUP','SE
RVER_OPERATION_GROUP','SERVER_PERMISSION_CHANGE_GROUP','SERVER_PRINCIPAL_CHAN
GE_GROUP','SERVER_PRINCIPAL_IMPERSONATION_GROUP','SERVER_ROLE_MEMBER_CHANGE_G
ROUP','SERVER_STATE_CHANGE_GROUP','TRACE_CHANGE_GROUP','USER_CHANGE_PASSWORD_
GROUP')
```

If the identified groups are not returned, this is a finding.

Remediation:

Add the required events to the server audit specification

```
USE [master];
GO

ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION WITH (STATE
= OFF);
GO

ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(APPLICATION_ROLE_CHANGE_PASSWORD_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(AUDIT_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(BACKUP_RESTORE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_OBJECT_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_OBJECT_OWNERSHIP_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_OBJECT_PERMISSION_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_OPERATION_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_OWNERSHIP_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_PERMISSION_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_PRINCIPAL_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_PRINCIPAL_IMPERSONATION_GROUP);
```

```
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_ROLE_MEMBER_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DBCC_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(LOGIN_CHANGE_PASSWORD_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SCHEMA_OBJECT_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SCHEMA_OBJECT_OWNERSHIP_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SCHEMA_OBJECT_PERMISSION_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SERVER_OBJECT_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SERVER_OBJECT_OWNERSHIP_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SERVER_OBJECT_PERMISSION_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SERVER_OPERATION_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SERVER_PERMISSION_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SERVER_PRINCIPAL_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SERVER_PRINCIPAL_IMPERSONATION_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SERVER_ROLE_MEMBER_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SERVER_STATE_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(TRACE_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(USER_CHANGE_PASSWORD_GROUP);
GO

ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION WITH (STATE
= ON);
GO
```

Additional Information:

CCI-000172

Generate audit records for the event types defined in AU-2 c that include the audit record content defined in AU-3.

- NIST SP 800-53::AU-12 c
- NIST SP 800-53A::AU-12.1 (iv)
- NIST SP 800-53 Revision 4::AU-12 c
- NIST SP 800-53 Revision 5::AU-12 c

1.71 SQL6-D0-015000 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must generate audit records when unsuccessful attempts to execute privileged activities or other system-level access occur.

```
GROUP ID: V-214016
RULE ID: SV-214016r961827
```

Rationale:

Without tracking privileged activity, it would be difficult to establish, correlate, and investigate the events relating to an incident or identify those responsible for one.

System documentation should include a definition of the functionality considered privileged.

A privileged function in this context is any operation that modifies the structure of the database, its built-in logic, or its security settings. This would include all Data Definition Language (DDL) statements and all security-related statements. In an SQL environment, it encompasses, but is not necessarily limited to:

- CREATE
- ALTER
- DROP
- GRANT
- REVOKE
- DENY

Note that it is particularly important to audit, and tightly control, any action that weakens the implementation of this requirement itself, since the objective is to have a complete audit trail of all administrative activity.

To aid in diagnosis, it is necessary to keep track of failed attempts in addition to the successful ones.

Audit:

Determine if an audit is configured and started by executing the following query.

```
SELECT name AS 'Audit Name',
status_desc AS 'Audit Status',
audit_file_path AS 'Current Audit File'
FROM sys.dm_server_audit_status
```

If no records are returned, this is a finding.

Execute the following query to verify the following events are included in the server audit specification:

```
APPLICATION_ROLE_CHANGE_PASSWORD_GROUP
AUDIT_CHANGE_GROUP
BACKUP_RESTORE_GROUP
DATABASE_CHANGE_GROUP
DATABASE_OBJECT_CHANGE_GROUP
DATABASE_OBJECT_OWNERSHIP_CHANGE_GROUP
DATABASE_OBJECT_PERMISSION_CHANGE_GROUP
DATABASE_OPERATION_GROUP
DATABASE_OWNERSHIP_CHANGE_GROUP
DATABASE_PERMISSION_CHANGE_GROUP
DATABASE_PRINCIPAL_CHANGE_GROUP
DATABASE_PRINCIPAL_IMPERSONATION_GROUP
DATABASE_ROLE_MEMBER_CHANGE_GROUP
DBCC_GROUP
LOGIN_CHANGE_PASSWORD_GROUP
LOGOUT_GROUP
SCHEMA_OBJECT_CHANGE_GROUP
SCHEMA_OBJECT_OWNERSHIP_CHANGE_GROUP
SCHEMA_OBJECT_PERMISSION_CHANGE_GROUP
SERVER_OBJECT_CHANGE_GROUP
SERVER_OBJECT_OWNERSHIP_CHANGE_GROUP
SERVER_OBJECT_PERMISSION_CHANGE_GROUP
SERVER_OPERATION_GROUP
SERVER_PERMISSION_CHANGE_GROUP
SERVER_PRINCIPAL_CHANGE_GROUP
SERVER_PRINCIPAL_IMPERSONATION_GROUP
SERVER_ROLE_MEMBER_CHANGE_GROUP
SERVER_STATE_CHANGE_GROUP
TRACE_CHANGE_GROUP
USER_CHANGE_PASSWORD_GROUP

SELECT a.name AS 'AuditName',
s.name AS 'SpecName',
d.audit_action_name AS 'ActionName',
d.audited_result AS 'Result'
FROM sys.server_audit_specifications s
JOIN sys.server_audits a ON s.audit_guid = a.audit_guid
JOIN sys.server_audit_specification_details d ON s.server_specification_id =
d.server_specification_id
WHERE a.is_state_enabled = 1
AND d.audit_action_name IN ('APPLICATION_ROLE_CHANGE_PASSWORD_GROUP',
'AUDIT_CHANGE_GROUP',
'BACKUP_RESTORE_GROUP',
'DATABASE_CHANGE_GROUP',
'DATABASE_OBJECT_CHANGE_GROUP',
'DATABASE_OBJECT_OWNERSHIP_CHANGE_GROUP',
'DATABASE_OBJECT_PERMISSION_CHANGE_GROUP',
'DATABASE_OPERATION_GROUP',
'DATABASE_OWNERSHIP_CHANGE_GROUP',
'DATABASE_PERMISSION_CHANGE_GROUP',
'DATABASE_PRINCIPAL_CHANGE_GROUP',
'DATABASE_PRINCIPAL_IMPERSONATION_GROUP',
'DATABASE_ROLE_MEMBER_CHANGE_GROUP',
```

```
'DBCC_GROUP',
'LOGIN_CHANGE_PASSWORD_GROUP',
'LOGOUT_GROUP',
'SCHEMA_OBJECT_CHANGE_GROUP',
'SCHEMA_OBJECT_OWNERSHIP_CHANGE_GROUP',
'SCHEMA_OBJECT_PERMISSION_CHANGE_GROUP',
'SERVER_OBJECT_CHANGE_GROUP',
'SERVER_OBJECT_OWNERSHIP_CHANGE_GROUP',
'SERVER_OBJECT_PERMISSION_CHANGE_GROUP',
'SERVER_OPERATION_GROUP',
'SERVER_PERMISSION_CHANGE_GROUP',
'SERVER_PRINCIPAL_CHANGE_GROUP',
'SERVER_PRINCIPAL_IMPERSONATION_GROUP',
'SERVER_ROLE_MEMBER_CHANGE_GROUP',
'SERVER_STATE_CHANGE_GROUP',
'TRACE_CHANGE_GROUP',
'USER_CHANGE_PASSWORD_GROUP'
)
Order by d.audit_action_name
```

If the identified groups are not returned, this is a finding.

Remediation:

Add the required events to the server audit specification

```
USE [master];
GO

ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION WITH (STATE
= OFF);
GO

ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(APPLICATION_ROLE_CHANGE_PASSWORD_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(AUDIT_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(BACKUP_RESTORE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_OBJECT_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_OBJECT_OWNERSHIP_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_OBJECT_PERMISSION_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_OPERATION_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_OWNERSHIP_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_PERMISSION_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_PRINCIPAL_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_PRINCIPAL_IMPERSONATION_GROUP);
```

```
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DATABASE_ROLE_MEMBER_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(DBCC_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(LOGIN_CHANGE_PASSWORD_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(LOGOUT_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SCHEMA_OBJECT_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SCHEMA_OBJECT_OWNERSHIP_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SCHEMA_OBJECT_PERMISSION_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SERVER_OBJECT_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SERVER_OBJECT_OWNERSHIP_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SERVER_OBJECT_PERMISSION_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SERVER_OPERATION_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SERVER_PERMISSION_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SERVER_PRINCIPAL_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SERVER_PRINCIPAL_IMPERSONATION_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SERVER_ROLE_MEMBER_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SERVER_STATE_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(TRACE_CHANGE_GROUP);
ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(USER_CHANGE_PASSWORD_GROUP);
GO

ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION WITH (STATE
= ON);
GO
```

Additional Information:

CCI-000172

Generate audit records for the event types defined in AU-2 c that include the audit record content defined in AU-3.

- NIST SP 800-53::AU-12 c
- NIST SP 800-53A::AU-12.1 (iv)
- NIST SP 800-53 Revision 4::AU-12 c
- NIST SP 800-53 Revision 5::AU-12 c

1.72 SQL6-D0-015100 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must generate audit records showing starting and ending time for user access to the database(s).

```
GROUP ID: V-214017
RULE ID: SV-214017r961830
```

Rationale:

For completeness of forensic analysis, it is necessary to know how long a user's (or other principal's) connection to SQL Server lasts. This can be achieved by recording disconnections, in addition to logons/connections, in the audit logs.

Disconnection may be initiated by the user or forced by the system (as in a timeout) or result from a system or network failure. To the greatest extent possible, all disconnections must be logged.

Audit:

Determine if an audit is configured and started by executing the following query:

```
SELECT name AS 'Audit Name',
status_desc AS 'Audit Status',
audit_file_path AS 'Current Audit File'
FROM sys.dm_server_audit_status
```

If no records are returned, this is a finding.

Execute the following query to verify the following events are included in the server audit specification:

```
APPLICATION_ROLE_CHANGE_PASSWORD_GROUP
AUDIT_CHANGE_GROUP
BACKUP_RESTORE_GROUP
DATABASE_CHANGE_GROUP
DATABASE_OBJECT_CHANGE_GROUP
DATABASE_OBJECT_OWNERSHIP_CHANGE_GROUP
DATABASE_OBJECT_PERMISSION_CHANGE_GROUP
DATABASE_OPERATION_GROUP
DATABASE_OWNERSHIP_CHANGE_GROUP
DATABASE_PERMISSION_CHANGE_GROUP
DATABASE_PRINCIPAL_CHANGE_GROUP
DATABASE_PRINCIPAL_IMPERSONATION_GROUP
DATABASE_ROLE_MEMBER_CHANGE_GROUP
DBCC_GROUP
LOGIN_CHANGE_PASSWORD_GROUP
LOGOUT_GROUP
```

```
SCHEMA_OBJECT_CHANGE_GROUP
SCHEMA_OBJECT_OWNERSHIP_CHANGE_GROUP
SCHEMA_OBJECT_PERMISSION_CHANGE_GROUP
SERVER_OBJECT_CHANGE_GROUP
SERVER_OBJECT_OWNERSHIP_CHANGE_GROUP
SERVER_OBJECT_PERMISSION_CHANGE_GROUP
SERVER_OPERATION_GROUP
SERVER_PERMISSION_CHANGE_GROUP
SERVER_PRINCIPAL_CHANGE_GROUP
SERVER_PRINCIPAL_IMPERSONATION_GROUP
SERVER_ROLE_MEMBER_CHANGE_GROUP
SERVER_STATE_CHANGE_GROUP
TRACE_CHANGE_GROUP
USER_CHANGE_PASSWORD_GROUP

SELECT a.name AS 'AuditName',
s.name AS 'SpecName',
d.audit_action_name AS 'ActionName',
d.audited_result AS 'Result'
FROM sys.server_audit_specifications s
JOIN sys.server_audits a ON s.audit_guid = a.audit_guid
JOIN sys.server_audit_specification_details d ON s.server_specification_id =
d.server_specification_id
WHERE a.is_state_enabled = 1
AND d.audit_action_name IN ('APPLICATION_ROLE_CHANGE_PASSWORD_GROUP',
'AUDIT_CHANGE_GROUP',
'BACKUP_RESTORE_GROUP',
'DATABASE_CHANGE_GROUP',
'DATABASE_OBJECT_CHANGE_GROUP',
'DATABASE_OBJECT_OWNERSHIP_CHANGE_GROUP',
'DATABASE_OBJECT_PERMISSION_CHANGE_GROUP',
'DATABASE_OPERATION_GROUP',
'DATABASE_OWNERSHIP_CHANGE_GROUP',
'DATABASE_PERMISSION_CHANGE_GROUP',
'DATABASE_PRINCIPAL_CHANGE_GROUP',
'DATABASE_PRINCIPAL_IMPERSONATION_GROUP',
'DATABASE_ROLE_MEMBER_CHANGE_GROUP',
'DBCC_GROUP',
'LOGIN_CHANGE_PASSWORD_GROUP',
'LOGOUT_GROUP',
'SCHEMA_OBJECT_CHANGE_GROUP',
'SCHEMA_OBJECT_OWNERSHIP_CHANGE_GROUP',
'SCHEMA_OBJECT_PERMISSION_CHANGE_GROUP',
'SERVER_OBJECT_CHANGE_GROUP',
'SERVER_OBJECT_OWNERSHIP_CHANGE_GROUP',
'SERVER_OBJECT_PERMISSION_CHANGE_GROUP',
'SERVER_OPERATION_GROUP',
'SERVER_PERMISSION_CHANGE_GROUP',
'SERVER_PRINCIPAL_CHANGE_GROUP',
'SERVER_PRINCIPAL_IMPERSONATION_GROUP',
'SERVER_ROLE_MEMBER_CHANGE_GROUP',
'SERVER_STATE_CHANGE_GROUP',
'TRACE_CHANGE_GROUP',
'USER_CHANGE_PASSWORD_GROUP'
)
Order by d.audit_action_name
```

If the identified groups are not returned, this is a finding.

Remediation:

Add the LOGOUT_GROUP to the server audit specification

```
USE [master];
GO

ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION WITH (STATE
= OFF);
GO

ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(LOGOUT_GROUP);
GO

ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION WITH (STATE
= ON);
GO
```

Additional Information:

CCI-000172

Generate audit records for the event types defined in AU-2 c that include the audit record content defined in AU-3.

- NIST SP 800-53::AU-12 c
- NIST SP 800-53A::AU-12.1 (iv)
- NIST SP 800-53 Revision 4::AU-12 c
- NIST SP 800-53 Revision 5::AU-12 c

1.73 SQL6-D0-015200 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must generate audit records when concurrent logons/connections by the same user from different workstations occur.

```
GROUP ID: V-214018
RULE ID: SV-214018r961833
```

Rationale:

For completeness of forensic analysis, it is necessary to track who logs on to SQL Server.

Concurrent connections by the same user from multiple workstations may be valid use of the system; or such connections may be due to improper circumvention of the requirement to use the CAC for authentication; or they may indicate unauthorized account sharing; or they may be because an account has been compromised.

(If the fact of multiple, concurrent logons by a given user can be reliably reconstructed from the log entries for other events (logons/connections; voluntary and involuntary disconnections), then it is not mandatory to create additional log entries specifically for this.)

Audit:

Determine if an audit is configured and started by executing the following query.

```
SELECT name AS 'Audit Name',
  status_desc AS 'Audit Status',
  audit_file_path AS 'Current Audit File'
FROM sys.dm_server_audit_status
```

If no records are returned, this is a finding.

Execute the following query to verify the SUCCESSFUL_LOGIN_GROUP is included in the server audit specification.

```
SELECT a.name AS 'AuditName',
s.name AS 'SpecName',
d.audit_action_name AS 'ActionName',
d.audited_result AS 'Result'
FROM sys.server_audit_specifications s
JOIN sys.server_audits a ON s.audit_guid = a.audit_guid
JOIN sys.server_audit_specification_details d ON s.server_specification_id =
d.server_specification_id
WHERE a.is_state_enabled = 1 AND d.audit_action_name =
'SUCCESSFUL_LOGIN_GROUP'
```

If the SUCCESSFUL_LOGIN_GROUP is returned in an active audit, this is not a finding.

If SUCCESSFUL_LOGIN_GROUP is not in the active audit, determine whether "Both failed and successful logins" is enabled.

In SQL Management Studio:

Right-click on the instance >> Select "Properties" >> Select "Security" on the left hand side >> Check the setting for "Login auditing"

If "Both failed and successful logins" is not selected, this is a finding.

Remediation:

Add the SUCCESSFUL_LOGIN_GROUP to the server audit specification.

```
USE [master];
GO

ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION WITH (STATE
= OFF);
GO

ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION ADD
(SUCCESSFUL_LOGIN_GROUP);
GO

ALTER SERVER AUDIT SPECIFICATION STIG_AUDIT_SERVER_SPECIFICATION WITH (STATE
= ON);
GO
```

Alternatively, enable "Both failed and successful logins"

In SQL Management Studio:

Right-click on the instance >> Select "Properties" >> Select "Security" on the left hand side >> Select "Both failed and successful logins" >> Click "OK"

Additional Information:

CCI-000172

Generate audit records for the event types defined in AU-2 c that include the audit record content defined in AU-3.

- NIST SP 800-53::AU-12 c
- NIST SP 800-53A::AU-12.1 (iv)
- NIST SP 800-53 Revision 4::AU-12 c
- NIST SP 800-53 Revision 5::AU-12 c

1.74 SQL6-D0-015400 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must generate audit records when successful and unsuccessful accesses to objects occur.

```
GROUP ID: V-214020
RULE ID: SV-214020r961836
```

Rationale:

Without tracking all or selected types of access to all or selected objects (tables, views, procedures, functions, etc.), it would be difficult to establish, correlate, and investigate the events relating to an incident, or identify those responsible for one.

In an SQL environment, types of access include, but are not necessarily limited to:

- SELECT
- INSERT
- UPDATE
- DELETE
- EXECUTE

To aid in diagnosis, it is necessary to keep track of failed attempts in addition to the successful ones.

Satisfies: SRG-APP-000507-DB-000356

Audit:

Review the system documentation to determine if SQL Server is required to audit when successful and unsuccessful accesses to objects occur.

If this is not required, this is not a finding.

If the documentation does not exist, this is a finding.

Determine if an audit is configured and started by executing the following query:

```
SELECT name AS 'Audit Name',
  status_desc AS 'Audit Status',
  audit_file_path AS 'Current Audit File'
FROM sys.dm_server_audit_status
```

If no records are returned, this is a finding.

If the auditing the retrieval of privilege/permission/role membership information is required, execute the following query to verify the SCHEMA_OBJECT_ACCESS_GROUP is included in the server audit specification.

```
SELECT a.name AS 'AuditName',
 s.name AS 'SpecName',
 d.audit_action_name AS 'ActionName',
 d.audited_result AS 'Result'
FROM sys.server_audit_specifications s
JOIN sys.server_audits a ON s.audit_guid = a.audit_guid
JOIN sys.server_audit_specification_details d ON s.server_specification_id =
d.server_specification_id
WHERE a.is_state_enabled = 1 AND d.audit_action_name =
'SCHEMA_OBJECT_ACCESS_GROUP'
```

If the SCHEMA_OBJECT_ACCESS_GROUP is not returned in an active audit, this is a finding.

Remediation:

Deploy an audit to audit when successful and unsuccessful accesses to objects occur. See the supplemental file SQL 2016 Audit.sql.

Additional Information:

CCI-000172

Generate audit records for the event types defined in AU-2 c that include the audit record content defined in AU-3.

- NIST SP 800-53::AU-12 c
- NIST SP 800-53A::AU-12.1 (iv)
- NIST SP 800-53 Revision 4::AU-12 c
- NIST SP 800-53 Revision 5::AU-12 c

1.75 SQL6-D0-015500 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must generate audit records for all direct access to the database(s).

```
GROUP ID: V-214021
RULE ID: SV-214021r961839
```

Rationale:

In this context, direct access is any query, command, or call to SQL Server that comes from any source other than the application(s) that it supports. Examples would be the command line or a database management utility program. The intent is to capture all activity from administrative and non-standard sources.

Audit:

Determine whether any Server Audits are configured to filter records. From SQL Server Management Studio execute the following query:

```
SELECT name AS AuditName, predicate AS AuditFilter
FROM sys.server_audits
WHERE predicate IS NOT NULL
```

If any audits are returned, review the associated filters to determine whether administrative activities are being excluded.

If any audits are configured to exclude administrative activities, this is a finding.

Remediation:

Check the system documentation for required SQL Server Audits. Remove any Audit filters that exclude or reduce required auditing. Update filters to ensure administrative activity is not excluded.

Additional Information:

CCI-000172

Generate audit records for the event types defined in AU-2 c that include the audit record content defined in AU-3.

- NIST SP 800-53::AU-12 c
- NIST SP 800-53A::AU-12.1 (iv)
- NIST SP 800-53 Revision 4::AU-12 c
- NIST SP 800-53 Revision 5::AU-12 c

1.76 SQL6-D0-015600 (Manual)

Profile Applicability:

- SEVERITY: CAT I

Description:

SQL Server must implement NIST FIPS 140-2 or 140-3 validated cryptographic modules to provision digital signatures.

```
GROUP ID: V-214022
RULE ID: SV-214022r961857
```

Rationale:

Use of weak or untested encryption algorithms undermines the purposes of utilizing encryption to protect data. The application must implement cryptographic modules adhering to the higher standards approved by the federal government since this provides assurance they have been tested and validated.

For detailed information, refer to NIST FIPS Publication 140-2 or Publication 140-3, Security Requirements For Cryptographic Modules. Note that the product's cryptographic modules must be validated and certified by NIST as FIPS-compliant.

Audit:

Verify that Windows is configured to require the use of FIPS compliant algorithms.

Click Start >> Type "Local Security Policy" >> Press Enter >> Expand "Local Policies" >> Select "Security Options" >> Locate "System Cryptography: Use FIPS compliant algorithms for encryption, hashing, and signing."

If the Security Setting for this option is "Disabled" this is a finding.

Remediation:

In Windows, open Administrative Tools >> Local Security Policy.

Expand Local Policies >> Security Options.

In the right-side pane, double-click on "System cryptography: Use FIPS compliant algorithms for encryption, hashing, and signing."

Additional Information:

CCI-002450

Implement organization-defined types of cryptography for each specified cryptography use.

- NIST SP 800-53 Revision 4::SC-13

- NIST SP 800-53 Revision 5::SC-13 b

1.77 SQL6-D0-015700 (Manual)

Profile Applicability:

- SEVERITY: CAT I

Description:

SQL Server must implement NIST FIPS 140-2 or 140-3 validated cryptographic modules to generate and validate cryptographic hashes.

```
GROUP ID: V-214023
RULE ID: SV-214023r961857
```

Rationale:

Use of weak or untested encryption algorithms undermines the purposes of utilizing encryption to protect data. The application must implement cryptographic modules adhering to the higher standards approved by the federal government since this provides assurance they have been tested and validated.

For detailed information, refer to NIST FIPS Publication 140-2 or Publication 140-3, Security Requirements For Cryptographic Modules. Note that the product's cryptographic modules must be validated and certified by NIST as FIPS-compliant.

Audit:

Verify that Windows is configured to require the use of FIPS-compliant algorithms.

Click Start >> Type "Local Security Policy" >> Press Enter >> Expand "Local Policies" >> Select "Security Options" >> Locate "System Cryptography: Use FIPS compliant algorithms for encryption, hashing, and signing."

If "System Cryptography: Use FIPS compliant algorithms for encryption, hashing, and signing" is not enabled, this is a finding.

Remediation:

Configure Windows to require the use of FIPS compliant algorithms.

Click Start >> Type "Local Security Policy" >> Press Enter >> Expand "Local Policies" >> Select "Security Options" >> Locate "System Cryptography: Use FIPS compliant algorithms for encryption, hashing, and signing." >> Change the Setting option to `Enabled` >> Restart Windows

Additional Information:

CCI-002450

Implement organization-defined types of cryptography for each specified cryptography use.

- NIST SP 800-53 Revision 4::SC-13
- NIST SP 800-53 Revision 5::SC-13 b

1.78 SQL6-D0-015800 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must implement NIST FIPS 140-2 or 140-3 validated cryptographic modules to protect unclassified information requiring confidentiality and cryptographic protection, in accordance with the data owners requirements.

```
GROUP ID: V-214024
RULE ID: SV-214024r961857
```

Rationale:

Use of weak or untested encryption algorithms undermines the purposes of utilizing encryption to protect data. The application must implement cryptographic modules adhering to the higher standards approved by the federal government since this provides assurance they have been tested and validated.

It is the responsibility of the data owner to assess the cryptography requirements in light of applicable federal laws, Executive Orders, directives, policies, regulations, and standards.

For detailed information, refer to NIST FIPS Publication 140-2 or Publication 140-3, Security Requirements For Cryptographic Modules. Note that the product's cryptographic modules must be validated and certified by NIST as FIPS-compliant.

Audit:

Review the server documentation, if this system does not contain data that must be encrypted, this finding is NA.

Verify that Windows is configured to require the use of FIPS compliant algorithms for the unclassified information that requires it.

Click Start >> Type "Local Security Policy" >> Press Enter >> Expand "Local Policies" >> Select "Security Options" >> Locate "System Cryptography: Use FIPS compliant algorithms for encryption, hashing, and signing."

If "System Cryptography: Use FIPS compliant algorithms for encryption, hashing, and signing" is not enabled, this is a finding.

Remediation:

Configure Windows to require the use of FIPS compliant algorithms for the unclassified information that requires it.

Click Start >> Type "Local Security Policy" >> Press Enter >> Expand "Local Policies" >> Select "Security Options" >> Locate "System Cryptography: Use FIPS compliant algorithms for encryption, hashing, and signing." >> Change the Setting option to `Enabled` >> Restart Windows

Additional Information:

CCI-002450

Implement organization-defined types of cryptography for each specified cryptography use.

- NIST SP 800-53 Revision 4::SC-13
- NIST SP 800-53 Revision 5::SC-13 b

1.79 SQL6-D0-015900 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

The system SQL Server must off-load audit data to a separate log management facility; this must be continuous and in near real time for systems with a network connection to the storage facility and weekly or more often for stand-alone systems.

```
GROUP ID: V-214025
RULE ID: SV-214025r961860
```

Rationale:

Information stored in one location is vulnerable to accidental or incidental deletion or alteration.

Off-loading is a common process in information systems with limited audit storage capacity.

The system SQL Server may write audit records to database tables, to files in the file system, to other kinds of local repository, or directly to a centralized log management system. Whatever the method used, it must be compatible with off-loading the records to the centralized system.

Audit:

Review the system documentation for a description of how audit records are off-loaded.

If the system has a continuous network connection to the centralized log management system, but the DBMS audit records are not written directly to the centralized log management system or transferred in near-real-time, this is a finding.

If the system does not have a continuous network connection to the centralized log management system, and the DBMS audit records are not transferred to the centralized log management system weekly or more often, this is a finding.

Remediation:

Configure the system or deploy and configure software tools to transfer audit records to a centralized log management system, continuously and in near-real time where a continuous network connection to the log management system exists, or at least weekly in the absence of such a connection.

Additional Information:

CCI-001851

Transfer audit logs per organization-defined frequency to a different system, system component, or media than the system or system component conducting the logging.

- NIST SP 800-53 Revision 4::AU-4 (1)
- NIST SP 800-53 Revision 5::AU-4 (1)

1.80 SQL6-D0-016000 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must configure Customer Feedback and Error Reporting.

```
GROUP ID: V-214026
RULE ID: SV-214026r961863
```

Rationale:

By default, Microsoft SQL Server enables participation in the customer experience improvement program (CEIP). This program collects information about how its customers are using the product. Specifically, SQL Server collects information about the installation experience, feature usage, and performance. This information helps Microsoft improve the product to better meet customer needs.

Audit:

Launch "Registry Editor"

Navigate to `HKEY_LOCAL_MACHINE\Software\Microsoft\Microsoft SQL Server\[InstanceId]\CPE`

Review the following values: `CustomerFeedback`, `EnableErrorReporting`

Navigate to `HKEY_LOCAL_MACHINE\Software\Microsoft\Microsoft SQL Server\130`

Review the following values: `CustomerFeedback`, `EnableErrorReporting`

If this is a classified system, and any of the above values are not zero (`0`), this is a finding.

If this is an unclassified system, review the server documentation to determine whether CEIP participation is authorized.

If CEIP participation is not authorized, and any of the above values are one (`1`), this is a finding.

Remediation:

To disable participation in the CEIP program, change the value of the following registry keys to zero (`0`).

To enable participation in the CEIP program, change the value of the following registry keys to one (`1`).

```
HKEY_LOCAL_MACHINE\Software\Microsoft\Microsoft SQL
Server\[InstanceId]\CPE\CustomerFeedback
HKEY_LOCAL_MACHINE\Software\Microsoft\Microsoft SQL
Server\[InstanceId]\CPE\EnableErrorReporting
HKEY_LOCAL_MACHINE\Software\Microsoft\Microsoft SQL
Server\130\CustomerFeedback
HKEY_LOCAL_MACHINE\Software\Microsoft\Microsoft SQL
Server\130\EnableErrorReporting
```

Additional Information:

CCI-000366

Implement the security configuration settings.

- NIST SP 800-53::CM-6 b
- NIST SP 800-53A::CM-6.1 (iv)
- NIST SP 800-53 Revision 4::CM-6 b
- NIST SP 800-53 Revision 5::CM-6 b

1.81 SQL6-D0-016100 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server must configure SQL Server Usage and Error Reporting Auditing.

```
GROUP ID: V-214027
RULE ID: SV-214027r961863
```

Rationale:

By default, Microsoft SQL Server enables participation in the customer experience improvement program (CEIP). This program collects information about how its customers are using the product. Specifically, SQL Server collects information about the installation experience, feature usage, and performance. This information helps Microsoft improve the product to better meet customer needs. The Local Audit component of SQL Server Usage Feedback collection writes data collected by the service to a designated folder, representing the data (logs) that will be sent to Microsoft. The purpose of the Local Audit is to allow customers to see all data Microsoft collects with this feature, for compliance, regulatory or privacy validation reasons.

Audit:

Review the server documentation to determine if auditing of the telemetry data is required. If auditing of telemetry data is not required, this is not a finding.

If auditing of telemetry data is required, determine the telemetry service user name by executing the following query:

```
SELECT name
FROM sys.server_principals
WHERE name LIKE '%SQLTELEMETRY%'
```

Review the values of the following registry key:

Note: InstanceId refers to the type and instance of the feature. (e.g., `MSSQL13.SqlInstance`, `MSAS13.SSASInstance`, `MSRS13.SSRSInstance`)

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Microsoft SQL
Server\[InstanceId]\CPE\UserRequestedLocalAuditDirectory
```

If the registry key do not exist or the value is blank, this is a finding.

Navigate the path defined in the `UserRequestedLocalAuditDirectory` registry key in file explorer.

Right-click on the folder and choose "Properties".

Open the "Security" tab.

Verify the SQLTELEMETRY account has the following permissions:

- List folder contents
- Read
- Write

If the permissions are not set properly on the folder, this is a finding.

Open services.msc and find the telemetry service.

- For Database Engine, use SQL Server CEIP service (<*INSTANCENAME*>).
- For Analysis Services, use SQL Server Analysis Services CEIP (<*INSTANCENAME*>).

Right-click on the service and choose "Properties". Verify the "Startup type" is "Automatic."

If the service is not configured to automatically start, this is a finding.

Review the processes and procedures for reviewing the telemetry data. If there is evidence that the telemetry data is periodically reviewed in accordance with the processes and procedures, this is not a finding.

If no processes and procedures exist for reviewing telemetry data, this is a finding.

Remediation:

Configure the instance to audit telemetry data. More information about auditing telemetry data can be found at https://msdn.microsoft.com/en-us/library/mt743085.aspx.

Create a folder to store the telemetry audit data in.

Grant the SQLTELEMETRY service the following permissions on the folder:

- List folder contents
- Read
- Write

Create and configure the following registry key:

Note: InstanceId refers to the type and instance of the feature. (e.g., MSSQL13.SqlInstance, MSAS13.SSASInstance, MSRS13.SSRSInstance)

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Microsoft SQL
Server\[InstanceId]\CPE\UserRequestedLocalAuditDirectory [string]
```

Set the UserRequestedLocalAuditDirectory key value to the path of the telemetry audit folder.

Set the telemetry service to start automatically. Restart the service.

- For Database Engine, use SQL Server CEIP service (<*INSTANCENAME*>).
- For Analysis Services, use SQL Server Analysis Services CEIP (<*INSTANCENAME*>).

Additional Information:

CCI-000366

Implement the security configuration settings.

- NIST SP 800-53::CM-6 b
- NIST SP 800-53A::CM-6.1 (iv)
- NIST SP 800-53 Revision 4::CM-6 b
- NIST SP 800-53 Revision 5::CM-6 b

1.82 SQL6-D0-016200 (Manual)

Profile Applicability:

- SEVERITY: CAT I

Description:

The SQL Server default account [sa] must be disabled.

```
GROUP ID: V-214028
RULE ID: SV-214028r960792
```

Rationale:

SQL Server's [sa] account has special privileges required to administer the database. The [sa] account is a well-known SQL Server account and is likely to be targeted by attackers and thus more prone to providing unauthorized access to the database.

This [sa] default account is administrative and could lead to catastrophic consequences, including the complete loss of control over SQL Server. If the [sa] default account is not disabled, an attacker might be able to gain access through the account. SQL Server by default disables the [sa] account at installation.

Some applications that run on SQL Server require the [sa] account to be enabled for the application to function properly. These applications that require the [sa] account to be enabled are usually legacy systems.

Audit:

Check SQL Server settings to determine if the `[sa]` (system administrator) account has been disabled by executing the following query:

```
USE master;
GO
SELECT name, is_disabled
FROM sys.sql_logins
WHERE principal_id = 1;
GO
```

Verify that the "name" column contains the current name of the [sa] database server account.

If the `is_disabled` column is not set to `1`, this is a finding.

Remediation:

Modify the enabled flag of SQL Server's [sa] (system administrator) account by running the following script.

```
USE master;
GO
ALTER LOGIN [sa] DISABLE;
```

GO

Additional Information:

CCI-000213

Enforce approved authorizations for logical access to information and system resources in accordance with applicable access control policies.

- NIST SP 800-53::AC-3
- NIST SP 800-53A::AC-3.1
- NIST SP 800-53 Revision 4::AC-3
- NIST SP 800-53 Revision 5::AC-3

1.83 SQL6-D0-016300 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server default account [sa] must have its name changed.

```
GROUP ID: V-214029
RULE ID: SV-214029r960963
```

Rationale:

SQL Server's [sa] account has special privileges required to administer the database. The [sa] account is a well-known SQL Server account name and is likely to be targeted by attackers, and is thus more prone to providing unauthorized access to the database.

Since the SQL Server [sa] is administrative in nature, the compromise of a default account can have catastrophic consequences, including the complete loss of control over SQL Server. Since SQL Server needs for this account to exist and it should not be removed, one way to mitigate this risk is to change the [sa] account name.

Audit:

Verify the SQL Server default [sa] (system administrator) account name has been changed by executing the following query:

```
USE master;
GO
SELECT *
FROM sys.sql_logins
WHERE [name] = 'sa' OR [principal_id] = 1;
GO
```

If the login account name SA or sa appears in the query output, this is a finding.

Remediation:

Modify the SQL Server's [sa] (system administrator) account by running the following script:

```
USE master;
GO
ALTER LOGIN [sa] WITH NAME = <new name>
GO
```

Additional Information:

CCI-000381

Configure the system to provide only organization-defined mission essential capabilities.

- NIST SP 800-53::CM-7
- NIST SP 800-53A::CM-7.1 (ii)
- NIST SP 800-53 Revision 4::CM-7 a
- NIST SP 800-53 Revision 5::CM-7 a

1.84 SQL6-D0-016400 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

Execution of startup stored procedures must be restricted to necessary cases only.

```
GROUP ID: V-214030
RULE ID: SV-214030r961359
```

Rationale:

In certain situations, to provide required functionality, a DBMS needs to execute internal logic (stored procedures, functions, triggers, etc.) and/or external code modules with elevated privileges. However, if the privileges required for execution are at a higher level than the privileges assigned to organizational users invoking the functionality applications/programs, those users are indirectly provided with greater privileges than assigned by organizations.

When 'Scan for startup procs' is enabled, SQL Server scans for and runs all automatically run stored procedures defined on the server. The execution of start-up stored procedures will be done under a high privileged context, therefore it is a commonly used post-exploitation vector.

Audit:

Review the system documentation to obtain a listing of documented stored procedures used by SQL Server during start up. Execute the following query:

```
Select [name] as StoredProc
From sys.procedures
Where OBJECTPROPERTY(OBJECT_ID, 'ExecIsStartup') = 1
```

If any stored procedures are returned that are not documented, this is a finding.

Remediation:

To disable start up stored procedure(s), run the following in Master for each undocumented procedure:

```
sp_procoption @procname = '<procedure name>', @OptionName = 'Startup',
@optionValue = 'Off'
```

Additional Information:

CCI-002233

Prevent the organization-defined software from executing at higher privilege levels than users executing the software.

- NIST SP 800-53 Revision 4::AC-6 (8)
- NIST SP 800-53 Revision 5::AC-6 (8)

1.85 SQL6-D0-016500 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server Mirroring endpoint must utilize AES encryption.

```
GROUP ID: V-214031
RULE ID: SV-214031r961863
```

Rationale:

Information can be either unintentionally or maliciously disclosed or modified during preparation for transmission, including, for example, during aggregation, at protocol transformation points, and during packing/unpacking. These unauthorized disclosures or modifications compromise the confidentiality or integrity of the information.

Use of this requirement will be limited to situations where the data owner has a strict requirement for ensuring data integrity and confidentiality is maintained at every step of the data transfer and handling process.

SQL Mirroring endpoints support different encryption algorithms, including no-encryption. Using a weak encryption algorithm or plaintext in communication protocols can lead to data loss, data manipulation and/or connection hijacking.

Audit:

If the data owner does not have a strict requirement for ensuring data integrity and confidentiality is maintained at every step of the data transfer and handling process, and the requirement is documented and authorized, this is not a finding.

If Database Mirroring is in use, run the following to check for encrypted transmissions:

```
SELECT name, type_desc, encryption_algorithm_desc
FROM sys.database_mirroring_endpoints
WHERE encryption_algorithm != 2
```

If any records are returned, this is a finding.

Remediation:

Run the following to enable encryption on the mirroring endpoint:

```
ALTER ENDPOINT <Endpoint Name>
FOR DATABASE_MIRRORING
(ENCRYPTION = REQUIRED ALGORITHM AES)
```

Additional Information:

CCI-000366

Implement the security configuration settings.

- NIST SP 800-53::CM-6 b
- NIST SP 800-53A::CM-6.1 (iv)
- NIST SP 800-53 Revision 4::CM-6 b
- NIST SP 800-53 Revision 5::CM-6 b

1.86 SQL6-D0-016600 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server Service Broker endpoint must utilize AES encryption.

```
GROUP ID: V-214032
RULE ID: SV-214032r961863
```

Rationale:

Information can be either unintentionally or maliciously disclosed or modified during preparation for transmission, including, for example, during aggregation, at protocol transformation points, and during packing/unpacking. These unauthorized disclosures or modifications compromise the confidentiality or integrity of the information.

Use of this requirement will be limited to situations where the data owner has a strict requirement for ensuring data integrity and confidentiality is maintained at every step of the data transfer and handling process.

SQL Server Service Broker endpoints support different encryption algorithms, including no-encryption. Using a weak encryption algorithm or plaintext in communication protocols can lead to data loss, data manipulation and/or connection hijacking.

Audit:

If the data owner does not have a strict requirement for ensuring data integrity and confidentiality is maintained at every step of the data transfer and handling process, and the requirement is documented and authorized, this is not a finding.

If SQL Service Broker is in use, run the following to check for encrypted transmissions:

```
SELECT name, type_desc, encryption_algorithm_desc
FROM sys.service_broker_endpoints
WHERE encryption_algorithm != 2
```

If any records are returned, this is a finding.

Remediation:

Run the following to enable encryption on the Service Broker endpoint:

```
ALTER ENDPOINT <EndpointName>
FOR SERVICE_BROKER
(ENCRYPTION = REQUIRED ALGORITHM AES)
```

Additional Information:

CCI-000366

Implement the security configuration settings.

- NIST SP 800-53::CM-6 b
- NIST SP 800-53A::CM-6.1 (iv)
- NIST SP 800-53 Revision 4::CM-6 b
- NIST SP 800-53 Revision 5::CM-6 b

1.87 SQL6-D0-016700 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server execute permissions to access the registry must be revoked, unless specifically required and approved.

```
GROUP ID: V-214033
RULE ID: SV-214033r960963
```

Rationale:

Information systems are capable of providing a wide variety of functions and services. Some of the functions and services, provided by default, may not be necessary to support essential organizational operations (e.g., key missions, functions).

It is detrimental for applications to provide, or install by default, functionality exceeding requirements or mission objectives.

Applications must adhere to the principles of least functionality by providing only essential capabilities.

SQL Server may spawn additional external processes to execute procedures that are defined in the SQL Server but stored in external host files (external procedures). The spawned process used to execute the external procedure may operate within a different OS security context than SQL Server and provide unauthorized access to the host system.

The registry contains sensitive information, including password hashes as well as clear text passwords. Registry extended stored procedures allow Microsoft SQL Server to access the machine's registry. The sensitivity of these procedures are exacerbated if Microsoft SQL Server is run under the Windows account LocalSystem. LocalSystem can read and write nearly all values in the registry, even those not accessible by the Administrator. Unlike the `xp_cmdshell` extended stored procedure, which runs under a separate context if executed by a login not in the sysadmin role, the registry extended stored procedures always execute under the security context of the MSSQLServer service. Because the sensitive information is stored in the registry, it is essential that access to that information be properly guarded.

Audit:

To determine if permissions to execute registry extended stored procedures have been revoked from all users (other than dbo), execute the following command:

```
SELECT OBJECT_NAME(major_id) AS [Stored Procedure]
,dpr.NAME AS [Principal]
FROM sys.database_permissions AS dp
```

```
INNER JOIN sys.database_principals AS dpr ON dp.grantee_principal_id =
dpr.principal_id
WHERE major_id IN (
 OBJECT_ID('xp_regaddmultistring')
,OBJECT_ID('xp_regdeletekey')
,OBJECT_ID('xp_regdeletevalue')
,OBJECT_ID('xp_regenumvalues')
,OBJECT_ID('xp_regenumkeys')
,OBJECT_ID('xp_regremovemultistring')
,OBJECT_ID('xp_regwrite')
,OBJECT_ID('xp_instance_regaddmultistring')
,OBJECT_ID('xp_instance_regdeletekey')
,OBJECT_ID('xp_instance_regdeletevalue')
,OBJECT_ID('xp_instance_regenumkeys')
,OBJECT_ID('xp_instance_regenumvalues')
,OBJECT_ID('xp_instance_regremovemultistring')
,OBJECT_ID('xp_instance_regwrite')
)
AND dp.[type] = 'EX'
ORDER BY dpr.NAME;
```

If any records are returned, review the system documentation to determine whether the accessing of the registry via extended stored procedures are required and authorized. If it is not authorized, this is a finding.

Remediation:

Remove execute permissions to any registry extended stored procedure from all users (other than dbo).

```
USE master
GO
REVOKE EXECUTE ON [<procedureName>] FROM [<principal>]
GO
```

Additional Information:

CCI-000381

Configure the system to provide only organization-defined mission essential capabilities.

- NIST SP 800-53::CM-7
- NIST SP 800-53A::CM-7.1 (ii)
- NIST SP 800-53 Revision 4::CM-7 a
- NIST SP 800-53 Revision 5::CM-7 a

1.88 SQL6-D0-016800 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

Filestream must be disabled, unless specifically required and approved.

```
GROUP ID: V-214034
RULE ID: SV-214034r960963
```

Rationale:

Information systems are capable of providing a wide variety of functions and services. Some of the functions and services, provided by default, may not be necessary to support essential organizational operations (e.g., key missions, functions).

It is detrimental for applications to provide, or install by default, functionality exceeding requirements or mission objectives.

Applications must adhere to the principles of least functionality by providing only essential capabilities.

The most significant potential for attacking an instance is through the use of features that expose an external interface or ad hoc execution capability. `FILESTREAM` integrates the SQL Server Database Engine with an NTFS file system by storing varbinary(max) binary large object (BLOB) data as files on the file system. Transact-SQL statements can insert, update, query, search, and back up `FILESTREAM` data.

Audit:

Review the system documentation to see if FileStream is in use. If in use authorized, this is not a finding.

If FileStream is not documented as being authorized, execute the following query.

```
EXEC sp_configure 'filestream access level'
```

If `run_value` is greater than `0`, this is a finding.

This rule checks that Filestream SQL specific option is disabled.

```
SELECT CASE
        WHEN EXISTS (SELECT *
                     FROM sys.configurations
                     WHERE Name = 'filestream access level'
                           AND Cast(value AS INT) = 0) THEN 'No'
        ELSE 'Yes'
      END AS TSQLFileStreamAccess;
```

If the above query returns `Yes` in the `FileStreamEnabled` field, this is a finding.

Remediation:

Disable the use of `Filestream`.

1. Delete all `FILESTREAM` columns from all tables.

```
ALTER TABLE <name> DROP COLUMN <column name>
```

2. Disassociate tables from the FILESTREAM filegroups.

```
ALTER TABLE <name> SET (FILESTREAM_ON = 'NULL'
```

3. Remove all `FILESTREAM` data containers.

```
ALTER DATABASE <name> REMOVE FILE <file name>
```

4. Remove all `FILESTREAM` filegroups.

```
ALTER DATABASE <name> REMOVE FILEGROUP <file name>
```

5. Disable `FILESTREAM`.

```
EXEC sp_configure filestream_access_level, 0
    RECONFIGURE
```

6. Restart the SQL Service

Additional Information:

CCI-000381

Configure the system to provide only organization-defined mission essential capabilities.

- NIST SP 800-53::CM-7
- NIST SP 800-53A::CM-7.1 (ii)
- NIST SP 800-53 Revision 4::CM-7 a
- NIST SP 800-53 Revision 5::CM-7 a

1.89 SQL6-D0-017000 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

Ole Automation Procedures feature must be disabled, unless specifically required and approved.

```
GROUP ID: V-214035
RULE ID: SV-214035r960963
```

Rationale:

Information systems are capable of providing a wide variety of functions and services. Some of the functions and services, provided by default, may not be necessary to support essential organizational operations (e.g., key missions, functions).

It is detrimental for applications to provide, or install by default, functionality exceeding requirements or mission objectives.

Applications must adhere to the principles of least functionality by providing only essential capabilities.

SQL Server may spawn additional external processes to execute procedures that are defined in the SQL Server but stored in external host files (external procedures). The spawned process used to execute the external procedure may operate within a different OS security context than SQL Server and provide unauthorized access to the host system.

SQL Server is capable of providing a wide range of features and services. Some of the features and services, provided by default, may not be necessary, and enabling them could adversely affect the security of the system.

The Ole Automation Procedures option controls whether OLE Automation objects can be instantiated within Transact-SQL batches. These are extended stored procedures that allow SQL Server users to execute functions external to SQL Server in the security context of SQL Server.

The Ole Automation Procedures extended stored procedure allows execution of host executables outside the controls of database access permissions. This access may be exploited by malicious users who have compromised the integrity of the SQL Server database process to control the host operating system to perpetrate additional malicious activity.

Audit:

To determine if "Ole Automation Procedures" option is enabled, execute the following query:

```
EXEC SP_CONFIGURE 'show advanced options', '1';
RECONFIGURE WITH OVERRIDE;
EXEC SP_CONFIGURE 'Ole Automation Procedures';
```

If the value of `config_value` is `0`, this is not a finding.

If the value of `config_value` is `1`, review the system documentation to determine whether the use of "Ole Automation Procedures" is required and authorized. If it is not authorized, this is a finding.

Remediation:

Disable use of or remove any external application executable object definitions that are not authorized. To disable the use of "Ole Automation Procedures" option, from the query prompt:

```
sp_configure 'show advanced options', 1;
GO
RECONFIGURE;
GO
sp_configure 'Ole Automation Procedures', 0;
GO
RECONFIGURE;
GO
```

Additional Information:

CCI-000381

Configure the system to provide only organization-defined mission essential capabilities.

- NIST SP 800-53::CM-7
- NIST SP 800-53A::CM-7.1 (ii)
- NIST SP 800-53 Revision 4::CM-7 a
- NIST SP 800-53 Revision 5::CM-7 a

1.90 SQL6-D0-017100 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server User Options feature must be disabled, unless specifically required and approved.

```
GROUP ID: V-214036
RULE ID: SV-214036r960963
```

Rationale:

SQL Server is capable of providing a wide range of features and services. Some of the features and services, provided by default, may not be necessary, and enabling them could adversely affect the security of the system.

The user options option specifies global defaults for all users. A list of default query processing options is established for the duration of a user's work session. The user options option allows you to change the default values of the SET options (if the server's default settings are not appropriate).

Audit:

To determine if "User Options" option is enabled, execute the following query:

```
EXEC SP_CONFIGURE 'show advanced options', '1';
RECONFIGURE WITH OVERRIDE;
EXEC SP_CONFIGURE 'user options';
```

If the value of `config_value` is `0`, this is not a finding.

If the value of `config_value` is `1`, review the system documentation to determine whether the use of "user options" is required and authorized. If it is not authorized, this is a finding.

Remediation:

Disable use of or remove any external application executable object definitions that are not authorized. To disable the use of "User Options" option, from the query prompt:

```
sp_configure 'show advanced options', 1;
GO
RECONFIGURE;
GO
sp_configure 'user options', 0;
GO
RECONFIGURE;
GO
```

Additional Information:

CCI-000381

Configure the system to provide only organization-defined mission essential capabilities.

- NIST SP 800-53::CM-7
- NIST SP 800-53A::CM-7.1 (ii)
- NIST SP 800-53 Revision 4::CM-7 a
- NIST SP 800-53 Revision 5::CM-7 a

1.91 SQL6-D0-017200 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

Remote Access feature must be disabled, unless specifically required and approved.

```
GROUP ID: V-214037
RULE ID: SV-214037r960963
```

Rationale:

Information systems are capable of providing a wide variety of functions and services. Some of the functions and services, provided by default, may not be necessary to support essential organizational operations (e.g., key missions, functions).

It is detrimental for applications to provide, or install by default, functionality exceeding requirements or mission objectives.

Applications must adhere to the principles of least functionality by providing only essential capabilities.

SQL Server may spawn additional external processes to execute procedures that are defined in the SQL Server but stored in external host files (external procedures). The spawned process used to execute the external procedure may operate within a different OS security context than SQL Server and provide unauthorized access to the host system.

SQL Server is capable of providing a wide range of features and services. Some of the features and services, provided by default, may not be necessary, and enabling them could adversely affect the security of the system.

The Remote Access option controls the execution of local stored procedures on remote servers or remote stored procedures on local server. 'Remote access' functionality can be abused to launch a Denial-of-Service (DoS) attack on remote servers by off-loading query processing to a target.

Audit:

To determine if "Remote Access" option is enabled, execute the following query:

```
EXEC SP_CONFIGURE 'show advanced options', '1';
RECONFIGURE WITH OVERRIDE;
EXEC SP_CONFIGURE 'remote access';
```

If the value of `config_value` is `0`, this is not a finding.

If the value of `config_value` is `1`, review the system documentation to determine whether the use of "Remote Access" is required (linked servers) and authorized. If it is not authorized, this is a finding.

Remediation:

Disable use of or remove any external application executable object definitions that are not authorized. To disable the use of "Remote Access" option, from the query prompt:

```
sp_configure 'show advanced options', 1;
GO
RECONFIGURE;
GO
sp_configure 'remote access', 0;
GO
RECONFIGURE;
GO
```

Additional Information:

CCI-000381

Configure the system to provide only organization-defined mission essential capabilities.

- NIST SP 800-53::CM-7
- NIST SP 800-53A::CM-7.1 (ii)
- NIST SP 800-53 Revision 4::CM-7 a
- NIST SP 800-53 Revision 5::CM-7 a

1.92 SQL6-D0-017400 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

Hadoop Connectivity feature must be disabled, unless specifically required and approved.

```
GROUP ID: V-214038
RULE ID: SV-214038r960963
```

Rationale:

Information systems are capable of providing a wide variety of functions and services. Some of the functions and services, provided by default, may not be necessary to support essential organizational operations (e.g., key missions, functions).

It is detrimental for applications to provide, or install by default, functionality exceeding requirements or mission objectives.

Applications must adhere to the principles of least functionality by providing only essential capabilities.

SQL Server may spawn additional external processes to execute procedures that are defined in the SQL Server but stored in external host files (external procedures). The spawned process used to execute the external procedure may operate within a different OS security context than SQL Server and provide unauthorized access to the host system.

SQL Server is capable of providing a wide range of features and services. Some of the features and services, provided by default, may not be necessary, and enabling them could adversely affect the security of the system.

The Hadoop Connectivity feature allows multiple types of external data sources to be created and used across all sessions on the server. An exploit to the SQL Server instance could result in a compromise of the host system and external SQL Server resources.

Audit:

To determine if "Hadoop Connectivity" option is enabled, execute the following query:

```
EXEC SP_CONFIGURE 'show advanced options', '1';
RECONFIGURE WITH OVERRIDE;
EXEC SP_CONFIGURE 'hadoop connectivity';
```

If the value of `config_value` is `0`, this is not a finding.

If the value of `config_value` is `1`, review the system documentation to determine whether the use of "Hadoop Connectivity" option is required and authorized. If it is not authorized, this is a finding.

Remediation:

Disable use of or remove any external application executable object definitions that are not authorized. To disable the use of "Hadoop Connectivity" option, from the query prompt:

```
sp_configure 'show advanced options', 1;
GO
RECONFIGURE;
GO
sp_configure 'hadoop connectivity', 0;
GO
RECONFIGURE;
GO
```

Additional Information:

CCI-000381

Configure the system to provide only organization-defined mission essential capabilities.

- NIST SP 800-53::CM-7
- NIST SP 800-53A::CM-7.1 (ii)
- NIST SP 800-53 Revision 4::CM-7 a
- NIST SP 800-53 Revision 5::CM-7 a

1.93 SQL6-D0-017500 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

Allow Polybase Export feature must be disabled, unless specifically required and approved.

```
GROUP ID: V-214039
RULE ID: SV-214039r960963
```

Rationale:

Information systems are capable of providing a wide variety of functions and services. Some of the functions and services, provided by default, may not be necessary to support essential organizational operations (e.g., key missions, functions).

It is detrimental for applications to provide, or install by default, functionality exceeding requirements or mission objectives.

Applications must adhere to the principles of least functionality by providing only essential capabilities.

SQL Server may spawn additional external processes to execute procedures that are defined in the SQL Server but stored in external host files (external procedures). The spawned process used to execute the external procedure may operate within a different OS security context than SQL Server and provide unauthorized access to the host system.

SQL Server is capable of providing a wide range of features and services. Some of the features and services, provided by default, may not be necessary, and enabling them could adversely affect the security of the system.

The Allow Polybase Export feature allows an export of data to an external data source such as Hadoop File System or Azure Data Lake. An exploit to the SQL Server instance could result in a compromise of the host system and external SQL Server resources.

Audit:

To determine if "Allow Polybase Export" option is enabled, execute the following query: `` EXEC SP_CONFIGURE 'show advanced options', '1'; RECONFIGURE WITH OVERRIDE; EXEC SP_CONFIGURE 'allow polybase export';

```
If the value of `config_value` is `0`, this is not a finding.

If the value of `config_value` is `1`, review the system documentation to
determine whether the use of "Allow Polybase Export" is required and
authorized. If it is not authorized, this is a finding.
```

Remediation:

Disable use of or remove any external application executable object definitions that are not authorized. To disable the use of "Allow Polybase Export" option, from the query prompt:

```
sp_configure 'show advanced options', 1;
GO
RECONFIGURE;
GO
sp_configure 'allow polybase export', 0;
GO
RECONFIGURE;
GO
```

Additional Information:

CCI-000381

Configure the system to provide only organization-defined mission essential capabilities.

- NIST SP 800-53::CM-7
- NIST SP 800-53A::CM-7.1 (ii)
- NIST SP 800-53 Revision 4::CM-7 a
- NIST SP 800-53 Revision 5::CM-7 a

1.94 SQL6-D0-017600 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

Remote Data Archive feature must be disabled, unless specifically required and approved.

```
GROUP ID: V-214040
RULE ID: SV-214040r960963
```

Rationale:

Information systems are capable of providing a wide variety of functions and services. Some of the functions and services, provided by default, may not be necessary to support essential organizational operations (e.g., key missions, functions).

It is detrimental for applications to provide, or install by default, functionality exceeding requirements or mission objectives.

Applications must adhere to the principles of least functionality by providing only essential capabilities.

SQL Server may spawn additional external processes to execute procedures that are defined in the SQL Server but stored in external host files (external procedures). The spawned process used to execute the external procedure may operate within a different OS security context than SQL Server and provide unauthorized access to the host system.

SQL Server is capable of providing a wide range of features and services. Some of the features and services, provided by default, may not be necessary, and enabling them could adversely affect the security of the system.

The Remote Data Archive feature allows an export of local SQL Server data to an Azure SQL Database. An exploit to the SQL Server instance could result in a compromise of the host system and external SQL Server resources.

Audit:

To determine if "Remote Data Archive" option is enabled, execute the following query:

```
EXEC SP_CONFIGURE 'show advanced options', '1';
RECONFIGURE WITH OVERRIDE;
EXEC SP_CONFIGURE 'remote data archive';
```

If the value of `config_value` is `0`, this is not a finding.

If the value of `config_value` is `1`, review the system documentation to determine whether the use of "Remote Data Archive" is required and authorized. If it is not authorized, this is a finding.

Remediation:

Disable use of or remove any external application executable object definitions that are not authorized. To disable the use of "Remote Data Archive" option, from the query prompt:

```
sp_configure 'show advanced options', 1;
GO
RECONFIGURE;
GO
sp_configure 'remote data archive', 0;
GO
RECONFIGURE;
GO
```

Additional Information:

CCI-000381

Configure the system to provide only organization-defined mission essential capabilities.

- NIST SP 800-53::CM-7
- NIST SP 800-53A::CM-7.1 (ii)
- NIST SP 800-53 Revision 4::CM-7 a
- NIST SP 800-53 Revision 5::CM-7 a

1.95 SQL6-D0-017700 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server External Scripts Enabled feature must be disabled, unless specifically required and approved.

```
GROUP ID: V-214041
RULE ID: SV-214041r960963
```

Rationale:

SQL Server is capable of providing a wide range of features and services. Some of the features and services, provided by default, may not be necessary, and enabling them could adversely affect the security of the system.

The External Scripts Enabled feature allows scripts external to SQL such as files located in an R library to be executed.

Audit:

To determine if "External Scripts Enabled" option is enabled, execute the following query:

```
EXEC SP_CONFIGURE 'show advanced options', '1';
RECONFIGURE WITH OVERRIDE;
EXEC SP_CONFIGURE 'external scripts enabled';
```

If the value of `config_value` is `0`, this is not a finding.

If the value of `config_value` is `1`, review the system documentation to determine whether the use of "External Scripts Enabled" is required and authorized. If it is not authorized, this is a finding.

Remediation:

Disable use of or remove any external application executable object definitions that are not authorized. To disable the use of "External Scripts Enabled" option, from the query prompt:

```
sp_configure 'show advanced options', 1;
GO
RECONFIGURE;
GO
sp_configure 'external scripts enabled', 0;
GO
RECONFIGURE;
GO
```

Additional Information:

CCI-000381

Configure the system to provide only organization-defined mission essential capabilities.

- NIST SP 800-53::CM-7
- NIST SP 800-53A::CM-7.1 (ii)
- NIST SP 800-53 Revision 4::CM-7 a
- NIST SP 800-53 Revision 5::CM-7 a

1.96 SQL6-D0-017800 (Manual)

Profile Applicability:

- SEVERITY: CAT III

Description:

The SQL Server Browser service must be disabled unless specifically required and approved.

```
GROUP ID: V-214042
RULE ID: SV-214042r961863
```

Rationale:

The SQL Server Browser simplifies the administration of SQL Server, particularly when multiple instances of SQL Server coexist on the same computer. It avoids the need to hard-assign port numbers to the instances and to set and maintain those port numbers in client systems. It enables administrators and authorized users to discover database management system instances, and the databases they support, over the network. SQL Server uses the SQL Server Browser service to enumerate instances of the Database Engine installed on the computer. This enables client applications to browse for a server, and helps clients distinguish between multiple instances of the Database Engine on the same computer.

This convenience also presents the possibility of unauthorized individuals gaining knowledge of the available SQL Server resources. Therefore, it is necessary to consider whether the SQL Server Browser is needed. Typically, if only a single instance is installed, using the default name (MSSQLSERVER) and port assignment (1433), the Browser is not adding any value. The more complex the installation, the more likely SQL Server Browser is to be helpful.

This requirement is not intended to prohibit use of the Browser service in any circumstances. It calls for administrators and management to consider whether the benefits of its use outweigh the potential negative consequences of it being used by an attacker to browse the current infrastructure and retrieve a list of running SQL Server instances.

Audit:

If the need for the SQL Server Browser service is documented and authorized, this is not a finding.

Open the Services tool.

Either navigate, via the Windows Start Menu and/or Control Panel, to "Administrative Tools", and select "Services"; or at a command prompt, type `services.msc` and press the "Enter" key.

Scroll to "SQL Server Browser".

If its Startup Type is not shown as `Disabled`, this is a finding.

Remediation:

If SQL Server Browser is needed, document the justification and obtain the appropriate authorization.

Where SQL Server Browser is judged unnecessary, the Service can be disabled.

To disable, in the Services tool, double-click "SQL Server Browser". Set "Startup Type" to `Disabled`. If "Service Status" is "Running", click on "Stop". Click on "OK".

Additional Information:

CCI-000366

Implement the security configuration settings.

- NIST SP 800-53::CM-6 b
- NIST SP 800-53A::CM-6.1 (iv)
- NIST SP 800-53 Revision 4::CM-6 b
- NIST SP 800-53 Revision 5::CM-6 b

1.97 SQL6-D0-017900 (Manual)

Profile Applicability:

- SEVERITY: CAT II

Description:

SQL Server Replication Xps feature must be disabled, unless specifically required and approved.

```
GROUP ID: V-214043
RULE ID: SV-214043r960963
```

Rationale:

SQL Server is capable of providing a wide range of features and services. Some of the features and services, provided by default, may not be necessary, and enabling them could adversely affect the security of the system.

Enabling the replication XPs opens a significant attack surface area that can be used by an attacker to gather information about the system and potentially abuse the privileges of SQL Server.

Audit:

To determine if the "Replication Xps" option is enabled, execute the following query:

```
EXEC SP_CONFIGURE 'show advanced options', '1';
RECONFIGURE WITH OVERRIDE;
EXEC SP_CONFIGURE 'replication xps';
```

If the value of `config_value` is `0`, this is not a finding.

If the value of `config_value` is `1`, review the system documentation to determine whether the use of "Replication Xps" is required and authorized. If it is not authorized, this is a finding.

Remediation:

Disable use of or remove any external application executable object definitions that are not authorized. To disable the use of "Replication Xps" option, from the query prompt:

```
sp_configure 'show advanced options', 1;
GO
RECONFIGURE;
GO
sp_configure 'replication xps', 0;
GO
RECONFIGURE;
GO
```

Additional Information:

CCI-000381

Configure the system to provide only organization-defined mission essential capabilities.

- NIST SP 800-53::CM-7
- NIST SP 800-53A::CM-7.1 (ii)
- NIST SP 800-53 Revision 4::CM-7 a
- NIST SP 800-53 Revision 5::CM-7 a

1.98 SQL6-D0-018000 (Manual)

Profile Applicability:

- SEVERITY: CAT III

Description:

If the SQL Server Browser Service is specifically required and approved, SQL instances must be hidden.

```
GROUP ID: V-214044
RULE ID: SV-214044r961863
```

Rationale:

The SQL Server Browser simplifies the administration of SQL Server, particularly when multiple instances of SQL Server coexist on the same computer. It avoids the need to hard-assign port numbers to the instances and to set and maintain those port numbers in client systems. It enables administrators and authorized users to discover database management system instances, and the databases they support, over the network. SQL Server uses the SQL Server Browser service to enumerate instances of the Database Engine installed on the computer. This enables client applications to browse for a server, and helps clients distinguish between multiple instances of the Database Engine on the same computer.

This convenience also presents the possibility of unauthorized individuals gaining knowledge of the available SQL Server resources. Therefore, it is necessary to consider whether the SQL Server Browser is needed. Typically, if only a single instance is installed, using the default name (MSSQLSERVER) and port assignment (1433), the Browser is not adding any value. The more complex the installation, the more likely SQL Server Browser is to be helpful.

This requirement is not intended to prohibit use of the Browser service in any circumstances. It calls for administrators and management to consider whether the benefits of its use outweigh the potential negative consequences of it being used by an attacker to browse the current infrastructure and retrieve a list of running SQL Server instances. In order to prevent this, the SQL instance(s) can be hidden.

Audit:

If the need for the SQL Server Browser service is documented and authorized, check to make sure the SQL Instances that do not require use of the SQL Browser Service are hidden with the following query:

```
DECLARE @HiddenInstance INT
EXEC master.dbo.Xp_instance_regread
 N'HKEY_LOCAL_MACHINE',
 N'Software\Microsoft\MSSQLServer\MSSQLServer\SuperSocketNetLib',
 N'HideInstance',
 @HiddenInstance output
```

```
SELECT CASE
        WHEN @HiddenInstance = 0
             AND Serverproperty('IsClustered') = 0 THEN 'No'
        ELSE 'Yes'
      END AS [Hidden]
```

If the value of `Hidden` is `Yes`, this is not a finding.

If the value of `Hidden` is `No` and the startup type of the "SQL Server Browser" service is not `Disabled`, this is a finding.

Remediation:

If SQL Server Browser is needed, document the justification and obtain the appropriate authorization.

To hide the SQL instance, in SQL Server Configuration Manager, expand SQL Server Network Configuration, right-click Protocols for , select "Properties", on the "Flags" tab, select "Yes" in the `HideInstance` box, then click "OK". The change takes effect immediately for new connections.

Additional Information:

CCI-000366

Implement the security configuration settings.

- NIST SP 800-53::CM-6 b
- NIST SP 800-53A::CM-6.1 (iv)
- NIST SP 800-53 Revision 4::CM-6 b
- NIST SP 800-53 Revision 5::CM-6 b

1.99 SQL6-D0-018100 (Manual)

Profile Applicability:

- SEVERITY: CAT I

Description:

When using command-line tools such as SQLCMD in a mixed-mode authentication environment, users must use a logon method that does not expose the password.

```
GROUP ID: V-214045
RULE ID: SV-214045r961047
```

Rationale:

To prevent the compromise of authentication information, such as passwords and PINs, during the authentication process, the feedback from the information system must not provide any information that would allow an unauthorized user to compromise the authentication mechanism.

Obfuscation of user-provided information when typed into the system is a method used in addressing this risk.

For example, displaying asterisks when a user types in a password or PIN, is an example of obscuring feedback of authentication information.

This requirement is applicable when mixed-mode authentication is enabled. When this is the case, password-authenticated accounts can be created in and authenticated by SQL Server. Other STIG requirements prohibit the use of mixed-mode authentication except when justified and approved. This deals with the exceptions.

SQLCMD and other command-line tools are part of any SQL Server installation. These tools can accept a plain-text password, but do offer alternative techniques. Since the typical user of these tools is a database administrator, the consequences of password compromise are particularly serious. Therefore, the use of plain-text passwords must be prohibited, as a matter of practice and procedure.

Audit:

Run this query to determine whether SQL Server authentication is enabled:

```
EXEC master.sys.xp_loginconfig 'login mode';
```

If the `config_value` returned is `Windows NT Authentication`, this is not a finding.

For `SQLCMD`, which cannot be configured not to accept a plain-text password, and any other essential tool with the same limitation, verify that the system documentation explains the need for the tool, who uses it, and any relevant mitigations; and that AO approval has been obtained; if not, this is a finding.

Request evidence that all users of the tool are trained in the importance of not using the plain-text password option and in how to keep the password hidden; and that they adhere to this practice; if not, this is a finding.

Remediation:

Where possible, change the login mode to Windows-only:

```
USE [master]
GO
EXEC xp_instance_regwrite N'HKEY_LOCAL_MACHINE',
N'Software\Microsoft\MSSQLServer\MSSQLServer', N'LoginMode', REG_DWORD, 1;
GO
```

If mixed-mode authentication is necessary, then for SQLCMD, which cannot be configured not to accept a plain-text password when mixed-mode authentication is enabled, and any other essential tool with the same limitation:

1. Document the need for it, who uses it, and any relevant mitigations, and obtain AO approval.
2. Train all users of the tool in the importance of not using the plain-text password option and in how to keep the password hidden.

Additional Information:

CCI-000206

Obscure feedback of authentication information during the authentication process to protect the information from possible exploitation and use by unauthorized individuals.

- NIST SP 800-53::IA-6
- NIST SP 800-53A::IA-6.1
- NIST SP 800-53 Revision 4::IA-6
- NIST SP 800-53 Revision 5::IA-6

1.100 SQL6-D0-018200 (Manual)

Profile Applicability:

- SEVERITY: CAT I

Description:

Applications must obscure feedback of authentication information during the authentication process to protect the information from possible exploitation/use by unauthorized individuals.

```
GROUP ID: V-214046
RULE ID: SV-214046r961047
```

Rationale:

To prevent the compromise of authentication information, such as passwords and PINs, during the authentication process, the feedback from the information system must not provide any information that would allow an unauthorized user to compromise the authentication mechanism.

Obfuscation of user-provided information when typed into the system is a method used in addressing this risk.

For example, displaying asterisks when a user types in a password or PIN, is an example of obscuring feedback of authentication information.

Database applications may allow for entry of the account name and password as a visible parameter of the application execution command. This practice must be prohibited and disabled to prevent shoulder surfing.

Audit:

Determine whether any applications that access the database allow for entry of the account name and password, or PIN.

If any do, determine whether these applications obfuscate authentication data; if they do not, this is a finding.

Remediation:

Configure or modify applications to prohibit display of passwords in clear text.

Additional Information:

CCI-000206

Obscure feedback of authentication information during the authentication process to protect the information from possible exploitation and use by unauthorized individuals.

- NIST SP 800-53::IA-6

- NIST SP 800-53A::IA-6.1
- NIST SP 800-53 Revision 4::IA-6
- NIST SP 800-53 Revision 5::IA-6

1.101 SQL6-D0-018300 (Manual)

Profile Applicability:

- SEVERITY: CAT I

Description:

Microsoft SQL Server products must be a version supported by the vendor.

```
GROUP ID: V-265870
RULE ID: SV-265870r999516
```

Rationale:

Unsupported commercial and database systems should not be used because fixes to newly identified bugs will not be implemented by the vendor. The lack of support can result in potential vulnerabilities.

Systems at unsupported servicing levels or releases will not receive security updates for new vulnerabilities, which leaves them subject to exploitation.

When maintenance updates and patches are no longer available, the database software is no longer considered supported and should be upgraded or decommissioned.

Audit:

Review the system documentation and interview the database administrator.

Identify all database software components.

Review the version and release information.

Verify the SQL Server version via one of the following methods:

Connect to the server by using Object Explorer in SQL Server Management Studio. After Object Explorer is connected, it will show the version information in parentheses, together with the user name that is used to connect to the specific instance of SQL Server.

Or, from SQL Server Management Studio:

```
SELECT @@VERSION;
```

More information for finding the version is available at the following link:

https://learn.microsoft.com/en-us/troubleshoot/sql/releases/find-my-sql-version

Access the vendor website or use other means to verify the version is still supported.

https://learn.microsoft.com/en-us/lifecycle/products/sql-server-2016

If the installed version or any of the software components are not supported by the vendor, this is a finding.

Remediation:

Remove or decommission all unsupported software products.

Upgrade unsupported DBMS or unsupported components to a supported version of the product.

Additional Information:

CCI-003376

Replace system components when support for the components is no longer available from the developer, vendor, or manufacturer.

- NIST SP 800-53 Revision 4::SA-22 a
- NIST SP 800-53 Revision 5::SA-22 a

Appendix: Summary Table

CIS Benchmark Recommendation		Set Correctly	
		Yes	No
1	**STIG RULES**		
1.1	SQL6-D0-003600 (Manual)	☐	☐
1.2	SQL6-D0-003700 (Manual)	☐	☐
1.3	SQL6-D0-003800 (Manual)	☐	☐
1.4	SQL6-D0-004000 (Manual)	☐	☐
1.5	SQL6-D0-003900 (Manual)	☐	☐
1.6	SQL6-D0-004100 (Manual)	☐	☐
1.7	SQL6-D0-004200 (Manual)	☐	☐
1.8	SQL6-D0-004300 (Manual)	☐	☐
1.9	SQL6-D0-004400 (Manual)	☐	☐
1.10	SQL6-D0-004700 (Manual)	☐	☐
1.11	SQL6-D0-004600 (Manual)	☐	☐
1.12	SQL6-D0-005500 (Manual)	☐	☐
1.13	SQL6-D0-005600 (Manual)	☐	☐
1.14	SQL6-D0-005700 (Manual)	☐	☐
1.15	SQL6-D0-005900 (Manual)	☐	☐
1.16	SQL6-D0-006300 (Manual)	☐	☐
1.17	SQL6-D0-006500 (Manual)	☐	☐
1.18	SQL6-D0-006600 (Manual)	☐	☐
1.19	SQL6-D0-006700 (Manual)	☐	☐

CIS Benchmark Recommendation		Set Correctly	
		Yes	No
1.20	SQL6-D0-006800 (Manual)	☐	☐
1.21	SQL6-D0-006900 (Manual)	☐	☐
1.22	SQL6-D0-007000 (Manual)	☐	☐
1.23	SQL6-D0-007100 (Manual)	☐	☐
1.24	SQL6-D0-007200 (Manual)	☐	☐
1.25	SQL6-D0-007300 (Manual)	☐	☐
1.26	SQL6-D0-007400 (Manual)	☐	☐
1.27	SQL6-D0-007500 (Manual)	☐	☐
1.28	SQL6-D0-007600 (Manual)	☐	☐
1.29	SQL6-D0-007700 (Manual)	☐	☐
1.30	SQL6-D0-007800 (Manual)	☐	☐
1.31	SQL6-D0-007900 (Manual)	☐	☐
1.32	SQL6-D0-008000 (Manual)	☐	☐
1.33	SQL6-D0-008200 (Manual)	☐	☐
1.34	SQL6-D0-008300 (Manual)	☐	☐
1.35	SQL6-D0-008400 (Manual)	☐	☐
1.36	SQL6-D0-008700 (Manual)	☐	☐
1.37	SQL6-D0-008800 (Manual)	☐	☐
1.38	SQL6-D0-009200 (Manual)	☐	☐
1.39	SQL6-D0-009500 (Manual)	☐	☐
1.40	SQL6-D0-009600 (Manual)	☐	☐
1.41	SQL6-D0-009700 (Manual)	☐	☐

CIS Benchmark Recommendation		Set Correctly	
		Yes	No
1.42	SQL6-D0-009800 (Manual)	☐	☐
1.43	SQL6-D0-009900 (Manual)	☐	☐
1.44	SQL6-D0-010000 (Manual)	☐	☐
1.45	SQL6-D0-010100 (Manual)	☐	☐
1.46	SQL6-D0-010400 (Manual)	☐	☐
1.47	SQL6-D0-010500 (Manual)	☐	☐
1.48	SQL6-D0-010900 (Manual)	☐	☐
1.49	SQL6-D0-011000 (Manual)	☐	☐
1.50	SQL6-D0-011100 (Manual)	☐	☐
1.51	SQL6-D0-011200 (Manual)	☐	☐
1.52	SQL6-D0-011400 (Manual)	☐	☐
1.53	SQL6-D0-011500 (Manual)	☐	☐
1.54	SQL6-D0-011800 (Manual)	☐	☐
1.55	SQL6-D0-011900 (Manual)	☐	☐
1.56	SQL6-D0-012300 (Manual)	☐	☐
1.57	SQL6-D0-012400 (Manual)	☐	☐
1.58	SQL6-D0-012700 (Manual)	☐	☐
1.59	SQL6-D0-012800 (Manual)	☐	☐
1.60	SQL6-D0-012900 (Manual)	☐	☐
1.61	SQL6-D0-013200 (Manual)	☐	☐
1.62	SQL6-D0-013400 (Manual)	☐	☐
1.63	SQL6-D0-013600 (Manual)	☐	☐

CIS Benchmark Recommendation		Set Correctly	
		Yes	No
1.64	SQL6-D0-013800 (Manual)	☐	☐
1.65	SQL6-D0-014000 (Manual)	☐	☐
1.66	SQL6-D0-014200 (Manual)	☐	☐
1.67	SQL6-D0-014400 (Manual)	☐	☐
1.68	SQL6-D0-014600 (Manual)	☐	☐
1.69	SQL6-D0-014800 (Manual)	☐	☐
1.70	SQL6-D0-014900 (Manual)	☐	☐
1.71	SQL6-D0-015000 (Manual)	☐	☐
1.72	SQL6-D0-015100 (Manual)	☐	☐
1.73	SQL6-D0-015200 (Manual)	☐	☐
1.74	SQL6-D0-015400 (Manual)	☐	☐
1.75	SQL6-D0-015500 (Manual)	☐	☐
1.76	SQL6-D0-015600 (Manual)	☐	☐
1.77	SQL6-D0-015700 (Manual)	☐	☐
1.78	SQL6-D0-015800 (Manual)	☐	☐
1.79	SQL6-D0-015900 (Manual)	☐	☐
1.80	SQL6-D0-016000 (Manual)	☐	☐
1.81	SQL6-D0-016100 (Manual)	☐	☐
1.82	SQL6-D0-016200 (Manual)	☐	☐
1.83	SQL6-D0-016300 (Manual)	☐	☐
1.84	SQL6-D0-016400 (Manual)	☐	☐
1.85	SQL6-D0-016500 (Manual)	☐	☐

CIS Benchmark Recommendation		Set Correctly	
		Yes	No
1.86	SQL6-D0-016600 (Manual)	☐	☐
1.87	SQL6-D0-016700 (Manual)	☐	☐
1.88	SQL6-D0-016800 (Manual)	☐	☐
1.89	SQL6-D0-017000 (Manual)	☐	☐
1.90	SQL6-D0-017100 (Manual)	☐	☐
1.91	SQL6-D0-017200 (Manual)	☐	☐
1.92	SQL6-D0-017400 (Manual)	☐	☐
1.93	SQL6-D0-017500 (Manual)	☐	☐
1.94	SQL6-D0-017600 (Manual)	☐	☐
1.95	SQL6-D0-017700 (Manual)	☐	☐
1.96	SQL6-D0-017800 (Manual)	☐	☐
1.97	SQL6-D0-017900 (Manual)	☐	☐
1.98	SQL6-D0-018000 (Manual)	☐	☐
1.99	SQL6-D0-018100 (Manual)	☐	☐
1.100	SQL6-D0-018200 (Manual)	☐	☐
1.101	SQL6-D0-018300 (Manual)	☐	☐

Appendix: Change History

Date	Version	Changes for this version
Sep 12, 2025	1.0.0	Initial CIS Release