	HARDENING SQL SERVER	
		V.1

CONTROL DE VERSIONES

Elaborado por:	David A. Hernández D., Ricardo Mario Andrade Muñiz, Roberto Vladimir Bernal Hernández	**No. de Versión:**	1
Revisado por:		**Fecha de revisión:**	
Aprobado por:		**Fecha de aprobación:**	

Historia de Modificaciones

No. de Versión	Fecha de Versión	Autor	Revisado por	Aprobado por	Descripción
1	Febrero 24, 2023	David Alejandro Hernández Díaz			
1	Febrero 24, 2023	Ricardo Mario Andrade Muñiz			
1	Febrero 24, 2023	Roberto Vladimir Bernal Hernández			

Este documento es propiedad de SETI. Prohibida su reproducción total o parcial sin previa autorización del autor.

1. DESCRIPCIÓN GENERAL

Objetivo

Presentar los puntos primordiales para ejecutar un proceso de gestión de seguridad y auditoria de los servicios del motor de Base de Datos SQL Server, establecidos por SETI, basados en documentos generados por el Fabricante y en la experiencia de SETI S.A.S.

Definiciones

Observaciones

Las características evaluadas en este documento aplican para las versiones de SQL Server 2012 en adelante en un servidor bajo sistema operativo Windows.

Tener en cuenta:

- Los Hardening por demanda realizados por parte del cliente pueden generar oportunidades de mejora puesto que los servicios están en constante cambio o evolución.
- Cualquier punto tratado en este documento podría representar un Riesgo el cual demandará un análisis más profundo del tema específico.
- La ejecución de esta actividad de Hardening puede llevar a la planeación y ejecución de actividades de PAYM.

Las tareas presentadas en este documento también aplican cuando se desea ajustar y mejorar los lineamientos vigentes conforme lo demanden las buenas prácticas del fabricante o la experiencia de SETI.

La Plantillas Default mencionada en este documento, se relacionan a continuación:
CIS_Microsoft_SQL_Server_2019_Benchmark_v1.2.0.pdf
CIS_Microsoft_SQL_Server_2016_Benchmark_v1.3.0.pdf

Este documento es propiedad de SETI. Prohibida su reproducción total o parcial sin previa autorización del autor.

CALIFICACION GENERAL DEL HARDENING

Teniendo en cuenta los puntos tratados en este documento , su nivel de criticidad y las evidencias obtenidas , resultado del proceso ejecutado, se presenta a continuación una tabla con los hallazgos que permitirá calificar el nivel de cumplimiento de este Hardening.

		NO CUMPLIMIENTO DE VULNERABILIDADES		
Motor	**Elemento (Cantidad Items)**	**ALTA**	**MEDIA**	**BAJA**
SQL Server	**VULNERABILIDADES**	**0%**	**0%**	**0%**
	Sistema Operativo (7)			
	Almacenamiento (3)			
	Instancia SQL Server (7)			
	Bases de Datos (4)			
	Seguridad (6)			
	Backups (2)			
Total ítems Evaluados:	**29**			

Este documento es propiedad de SETI. Prohibida su reproducción total o parcial sin previa autorización del autor.

1. VULNERABILIDADES A NIVEL DE SISTEMA OPERATIVO

TAREA	CRITICIDAD
1. Versión de SQL Server Para el momento del desarrollo de este documento, la versión mas reciente del motor SQL Server es SQL Server 2022, para efectos de hardening, se considerará la versión valida aquella que todavía cuente con soporte de tipo "Mainstream Support" por parte de Microsoft, por lo que versiones anteriores a 2019 se consideran obsoletas o sin soporte. **AUDITORIA:** A. Validar que la versión de SQL Server instalada sea igual o superior a 2019 `select @@version` **EVIDENCIAS:**	Alta
2. Validar el Service Pack aplicado para la versión instalada: Microsoft cada trimestre libera parches de seguridad o de mejora del motor de base de datos SQL Server los cuales se recomienda tener instalados, estos parches permiten mejorar la estabilidad operativa del motor y cerrar brechas de seguridad reportadas por entes externos o bien por Microsoft. Se evaluará si el motor de base de datos tiene el Service Pack (SP) mas reciente para su versión. **AUDITORIA:** A. Validar que el motor de base de datos esté parchado con el último SP disponible `select @@version` **EVIDENCIAS:**	Media

Este documento es propiedad de SETI. Prohibida su reproducción total o parcial sin previa autorización del autor.

TAREA	CRITICIDAD
3. <u>Versión del Sistema operativo, su nivel de parchado o hotfix aplicado</u> Los servicios de SQL Server deben correr en un servidor que tenga una versión de Microsoft Windows con soporte y debe tener los últimos parches o hotfix aplicados. Al igual que para el motor de base de datos, mantener la capa del sistema operativo actualizada permitirá tener un mejor rendimiento y resguardo a nivel de seguridad **NOTA:** Si bien no es obligatorio tener una versión de sistema operativo igual que la versión de SQL Server, se recomienda no instalar motores de bases de datos superiores en sistemas operativos anteriores ya que esto puede generar inestabilidad en el sistema. **AUDITORIA:** A. Verificar que los mínimos procesos de PostgreSQL estén en pleno funcionamiento: `SELECT windows_release, windows_service_pack_level, windows_sku, os_language_version FROM sys.dm_os_windows_info;` **EVIDENCIAS:**	Alta
4. <u>Ruta de instalación del SQL Server</u> Por default, SQL Server se instala en la misma unidad donde se instala el sistema operativo Windows, esto ocurre si en el momento de la instalación no se personaliza la configuración. Es recomendable tener una unidad separada para el motor de base de datos y más si la unidad C es limitada para consumo exclusivo del sistema operativo. **NOTA:** En una instalación, puede ocurrir que el motor de base de datos a nivel de binarios, quede en el mismo disco que el sistema operativo, sin embargo, es de sumo cuidado garantizar que las BD's del sistema (master, model, msdb y tempdb) no queden en el mismo disco C.	Medio

Este documento es propiedad de SETI. Prohibida su reproducción total o parcial sin previa autorización del autor.

TAREA	CRITICIDAD
AUDITORIA: B. Desde el Registry de Windows, ir a la siguiente ruta: `HKEY_LOCAL_MACHINE >> SOFTWARE >> Microsoft >> Microsoft SQL Server >> [INSTANCE NAME] >> Setup` **Nota:** Modificar INSTANCE_NAME por el nombre de la instancia a evaluar, si es una instancia por default el valor debe ser MSSQLSERVER. **EVIDENCIAS:**	
5. Usuarios y grupos a nivel de sistema operativo para SQL Server: Luego de que se instala el producto, a nivel de grupos y usuarios del sistema operativo, SQL Server genera unos personalizados para poder manejar de forma eficiente la seguridad que se le da a la instancia de base de datos, de forma que los usuarios que necesiten acceso administrativo a ella no necesariamente tengan que ser administradores a nivel del sistema operativo. **NOTA:** No es esencial que este grupo exista, de existir, solo usuarios que realmente deban actuar como administradores del servicio en general de la instancia de PostgreSQL, requieren estar asociados al mismo, de lo contrario, se activa un alto riesgo de seguridad. **AUDITORIA:** A. Validar a nivel del sistema operativo que los usuarios que acceden a la instancia para labore no administrativas no estén dentro del grupo de administradores locales. `getent group pg_wheel` B. Validar que los servicios de SQL Server no estén iniciados con usuarios locales o usuarios por default **EVIDENCIAS:**	Medio

Este documento es propiedad de SETI. Prohibida su reproducción total o parcial sin previa autorización del autor.

TAREA	CRITICIDAD
6. <u>Validar servicios instalados vs servicios utilizados:</u> Es común que en un despliegue del motor de base de datos de SQL Server se dejen varios servicios por default los cuales posiblemente no se estén utilizando tales como el Integration Services, Reporting Services, Analysis Services, entre otros. A pesar de que estos servicios no se estén usando, al tenerlos instalados y configurados y el servicio esté activo recursos del servidor están siendo asignados lo que limita las capacidades del sistema operativo y del propio motor de base de datos. **AUDITORIA:** A. Desde el configuration manager de SQL Server validar que servicios se encuentran activos. B. Se debe evaluar con el cliente si los servicios activos distintos al motor de base de datos están siendo usados. **EVIDENCIAS:**	Media
7. <u>Servicios de MSSQLSERVER y MSSQLSERVERAGENT inician de forma automática</u> Es imporatnte que los servicios de base de datos críticos como el motor de base de datos y el agente inicien de forma automática en caso de un reinicio del servidor. Esto permite que el propio sistema operativo gestione los servicios garantizando así la disponibilidad del mismo cuando el sistema operativo lo permita.	Bajo

Este documento es propiedad de SETI. Prohibida su reproducción total o parcial sin previa autorización del autor.

TAREA	CRITICIDAD
AUDITORIA: a) Desde el Configuration Manager de SQL Server garantizar que los servicios SQL Server y SQL Server Agent inicien de forma automática b) Tambien se pueden validar desde los servicios de Windows en la sección de SQL Server (esto en caso que el WMI no esté funcinando o limite el uso del Configuration Manager. **EVIDENCIAS:**	

2. VULNERABILIDADES A NIVEL DEL ALMACENAMIENTO

TAREA	CRITICIDAD
1. <u>Capacidad donde se encuentren almacenadas las bases de datos:</u> Se considera una buena practica tener disponible al menos el 15% del espacio en los almacenamientos donde se encuentren los archivos de datos de las bases de datos bien sean del sistema o del usuario, esto para poder gestionar de forma oportuna la asignación de espacio o depuración de data histórica (en caso de que aplique). **AUDITORIA:** a) Validar que el almacenamiento donde residen las bases de datos del sistema no superen el 85% de uso. b) Validar que el almacenamiento donde residen las bases de datos de usuarios no superen el 85% de uso.	Media

Este documento es propiedad de SETI. Prohibida su reproducción total o parcial sin previa autorización del autor.

TAREA	CRITICIDAD
EVIDENCIAS:	
2. <u>La base de datos TEMPDB debe tener un almacenamiento propio:</u> Se debe garantizar que durante el proceso de instalación del motor de base de datos, se seleccionó un almacenamiento personalizado para la base de datos TEMPDB, se recomienda siempre separar esta base de datos de las bases de datos del sistema y de usuario ya que su alta carga no afectará como tal el I/O sobre las demás. **AUDITORIA:** A. Validar que el mdf, ndf y ldf de la base de datos TEMPDB no estén ubicados en las mismas rutas que las otras bases de datos del sistema o bases de datos de usuario **EVIDENCIAS:**	Alta
3. <u>Separar los logs de transacciones de las bases de datos en discos independientes:</u> Por default, SQL Server crea tanto los archivos de datos (mdf, ndf) como los archivos del log de transacciones (ldf) en la misma ruta, a menos que durante la instalación se haya personalizado las rutas o bien se configuró en la instancia posteriormente en la sección de propiedades. Se recomienda ubicar los logs de transacciones en discos separados de los mdf y ndf de las bases de datos. **AUDITORIA:** A. Validar que los archivos de log (ldg) no estén en las mismas rutas donde residen los archivos de datos (mdf y/o ndf) **EVIDENCIAS:**	Alta

Este documento es propiedad de SETI. Prohibida su reproducción total o parcial sin previa autorización del autor.

3. VULNERABILIDADES A NIVEL DE LA INSTANCIA DE SQL SERVER

TAREA	CRITICIDAD
1. Configuración del máximo grado de paralelismo (MAXDOP): El máximo grado de paralelismo (MAXDOP) en una instancia SQL Server permite la ejecución de consultas en paralelo. Esto se genera cuando el valor del costo asociado de la consulta supera el valor configurado en la instancia y activa el paralelismo en función al valor del MAXDOP, es decir, si el valor del MAXDOP está configurado en 4, la consulta que se ejecute en paralelo iniciará 4 hilos en la instancia. **NOTA:** Si bien el paralelismo permitirá que las consultas se ejecuten más rápido, una configuración inadecuada puede generar problemas de rendimiento en la instancia de SQL Server. **AUDITORIA:** A. Validar en las propiedades de la instancia el valor del MAXDOP configurado B. Este valor debe estar asociado a la cantidad de CPU visibles por la instancia y su valor nunca debe exceder el máximo de CPU. C. Si el cliente no requiere el uso del MAXDOP su valor se debe establecer en cero (0) (Valor por Default) **EVIDENCIAS:**	**Media**
2. Configuración del Cost Threshold para el paralelismo en la instancia: Si bien el MAXDOP determina la cantidad de hilos que se activarán cuando se ejecuten sentencias con paralelismo, el cost threshold permite establecer el valor mínimo del costo de la consulta (a nivel de plan de ejecución) para que las funciones de paralelismo se activen. Este valor por default es 5 y se mide como segundos asociado al tiempo de ejecución aproximado. Adicional a estos parámetros, es importante verificar los evaluados en este documento, los cuales, estan	**Media**

Este documento es propiedad de SETI. Prohibida su reproducción total o parcial sin previa autorización del autor.

TAREA	CRITICIDAD
registrados en su mayoría en el archivo postgresql.conf. **NOTA:** Si el MAXDOP está apagado, esta función debe estar en su valor default (5), en caso de que el MAXDOP esté configurado, no se recomienda dejar en 5 ya que la gran mayoría de los procesos se ejecutarán con paralelismo. **AUDITORIA:** A. Validar si el MAXDOP está activado, en caso contrario omitir este paso. B. En caso afirmativo, validar los costos asociados a los queries ejecutados para determinar el valor mínimo requerido del Threshold. **EVIDENCIAS:**	
3. Rutas default para archivos de datos y archivos de logs: Para garantizar la buena administración a nivel de archivos de todas las bases de datos, se debe garantizar que durante el proceso de instalación se realizó la separación de archivos de datos (mdf y/o ndf) de los archivos de logs (ldf). Si se está realizando la evaluación posterior a la instalación, se debe validar esta condición en las propiedades de la instancia. **AUDITORIA:** A. Validar que la ruta default para los datos y logs no sean la misma ni estén donde reside el binario de base de datos. **EVIDENCIAS:**	**Alta**
4. Servicio de correo utilizado por la instancia de base de datos	**Media**

Este documento es propiedad de SETI. Prohibida su reproducción total o parcial sin previa autorización del autor.

TAREA	CRITICIDAD
En versiones superiores de SQL Server se recomienda utilizar la configuración del Database Mail de la propia instancia de base de datos para establecer los mecanismos de notificaciones. El uso de herramientas como el XP_SENDMAIL está prohibido ya que no es un procedimiento documentado por Microsoft y no tiene los controles de seguridad correspondientes. **AUDITORIA:** A. Se debe evaluar si el Database Mail está configurado en la instancia. B. Se debe evaluar que para los envios de correo no se esté utilizando el XP_SENDMAIL. **EVIDENCIAS:**	
5. Configuración del uso de los CPU y su afinidad Por default, cuando se instala el motor de SQL Server las propiedades de los CPU y la afinidad de los mismos toma todos los disponibles a nivel del sistema operativo. Dependiendo del alcance y al configuración requerida sobre la instancia, este valor se debe dejar en automático, para que la instancia pueda hacer uso de todos los procesadores presentes en el servidor.	**Media**

Este documento es propiedad de SETI. Prohibida su reproducción total o parcial sin previa autorización del autor.

TAREA	CRITICIDAD
AUDITORIA: A. Se debe evaluar que la afinidad y la cantidad de CPUs utilizados por la instancia esté en Automático (al menos que el cliente justifique la razón de la configuración custom) **EVIDENCIAS:**	
6. <u>Configuración de la memoria de la instancia (valor mínimo y máximo)</u> Por default, la instancia de SQL Server tiene configurado que tome toda la memoria del sistema operativo y que la reserve de forma dinámica, si bien en muchos casos esto puede traer beneficios, se recomienda establecer un valor mínimo y máximo para la memoria de la instancia de forma que el sistema operativo pueda manejar una parte para los demás servicios. **AUDITORIA:** A. Se debe validar que la instancia tenga configurado un valor mínimo y máximo para el uso de la memoria. Esto se valida a través de las propiedades de la instancia de SQL Server. B. El valor máximo debe estar acorde a la cantidad de memoria y no debe superar el 80% de la memoria total del servidor.	**Alta**

Este documento es propiedad de SETI. Prohibida su reproducción total o parcial sin previa autorización del autor.

TAREA	CRITICIDAD
EVIDENCIAS:	

B. VULNERABILIDADES A NIVEL DE BASE DE DATOS

TAREA	CRITICIDAD
1. Modo de recuperación de la base de dato debe estar en Full: Para poder garantizar un buen proceso de respaldo y recuperación de una base de datos, esta debe estar configurada en el modo de recuperación FULL, de esta forma, mediante la restauración de los respaldos de la BD como del log de transacciones, se puede miminizar la pérdida de datos en caso de fallos en la base de datos. **NOTA:** Si la base de datos está en modo de reuperación FULL, se deben establecer políticas de respaldo de archivelogs y actividades de mantenimiento de shrink en orden de mantener el tamaño del log de manera apropiada. **AUDITORIA:** A. En las propiedades de la base de datos, ir al apartado de opciones y validar que el modo de recuperación esté en el valor FULL.	**Alta**

Este documento es propiedad de SETI. Prohibida su reproducción total o parcial sin previa autorización del autor.

EVIDENCIAS:	
2. Las bases de datos deben tener asociado un Owner: Cuando se crea la base de datos, se define el usuario dueño de la misma el cual se conoce como Owner, el Owner es el dueño del esquema dbo y todo lo que se cree por default en la base de datos se creará bajo este esquema. Se recomienda configurar un owner apropiado para cada base de datos de forma de segmentar la seguridad y garantizar que solo ese owner pueda tener acceso a ella a ese nivel administrativo. **AUDITORIA:** A. Mediante el siguiente T-SQL observaremos la propiedad Owner de cada base de datos, cada una debe tener un usuario correspondiente y no debe estar en blanco o usando usuarios del sistema como el SA. `SELECT suser_sname( owner_sid ), * FROM sys.databases` **EVIDENCIAS:**	
3. EL collation de la base de datos debe ser igual al de la instancia Si bien cada base de datos puede tener un idioma especifico, sobre todo cuando varias Bases de datos de diferentes productos o aplicaciones conviven en la misma instancia, es recomendable utilizar el mismo collation de la instancia para la base de datos, esto permite que procesos como la creación de tablas temporales o el cruce (joins) entre tablas de una BD y otra no generen inconvenientes porque los collation sean distintivos. Nota: Esto solo aplica si los casos de negocio corresponden al hecho de que la instancia tenga 2 o más Bds de diferntes proveedores en diferentes collation.	**Baja**

Este documento es propiedad de SETI. Prohibida su reproducción total o parcial sin previa autorización del autor.

AUDITORIA:
 	
A. Validar el Collation a nivel de la instancia
`SELECT CONVERT (varchar(256), SERVERPROPERTY('collation'));`
B. Validar el Collation a nivel de las Bds
`SELECT name, collation_name FROM sys.databases;`
 	
EVIDENCIAS:	

4. Separación de esquemas dentro de las bases de datos: Es usual que durante la creación de la base de datos no se definan usuarios distintos al owner o al usuario de servicio quien tendrá acceso a la base de datos, esto se transforma a que todos los objetos desplegados sobre la base de datos van a pertenecer al esquema default dbo el cual está asociado al db_owner de la base de datos. Si varios usuarios acceden a la base de datos y existen funciones o elementos distintos donde cada objeto tenga un propósito independiente se debe establecer la separación de objetos por esquema. **Ejemplo:** Si en la base de datos conviven objetos asociados a un proceso de nómina pero también existen objetos asociados a procesos de talento humano, logística, u otro personal distinto, si todas las tablas pertenecen a DBO no es viable segmentar los permisos por lo que todos tendrán acceso a la información de los demás departamentos. **AUDITORIA:** A. Validar que los objetos de las bases de datos no tengan como único esquema a dbo `SELECT s.name AS schema_name,u.name AS schema_owner FROM sys.schemas s INNER JOIN sys.sysusers u ON u.uid = s.principal_id ORDER BY s.name;`	**Medio**

Este documento es propiedad de SETI. Prohibida su reproducción total o parcial sin previa autorización del autor.

EVIDENCIAS:	

C. VULNERABILIDADES A NIVEL DE SEGURIDAD

TAREA	CRITICIDAD
1. Determinar los usuarios con roles administrativos en la instancia de base de datos Usualmente los administradores por Default cuando se instala una instancia de SQL Server son el usuario que hizo la instalación y el usuario SA, durante el proceso se pueden personalizar quien o quienes serán los administradores de la instancia (rol sysadmin). Esto puede generar brechas de seguridad si se agregan uno o más usuarios que realmente no deban tener el rol administrativo. **NOTA:** Es importante validar que no existan grupos de directorio activo con roles administrativos ya que si en el grupo existen 100 usuarios, todos estos usuarios heredarán el rol a nivel de la instancia de SQL Server **AUDITORIA:** A. Validar los usuarios o grupos que tengan el rol administrativo sysadmin en la instancia de base de datos	**Alta**

Este documento es propiedad de SETI. Prohibida su reproducción total o parcial sin previa autorización del autor.

TAREA	CRITICIDAD
`SELECT 'Name' = sp.NAME,sp.is_disabled AS [Is_disabled] FROM sys.server_role_members rm ,sys.server_principals sp WHERE rm.role_principal_id = SUSER_ID('Sysadmin') AND rm.member_principal_id = sp.principal_id` **EVIDENCIAS:**	
2. Determinar si existen roles personalizados creados en la instancia de base de datos Para mejorar la seguridad de la instancia, es viable configurar roles personalizados los cuales puedan tener ciertas características distintas a los roles por Default, sin embargo, sin la correcta administración de estos roles se podrían generar brechas de seguridad ya que estos roles pueden tener a su vez permisos administrativos y pueden estar asignados a usuarios que no deben tenerlos. **AUDITORIA:** A. Validar los roles personalizados creados en la instancia y los usuarios que los tienen asociados. `SELECT r.name role_principal_name, m.name AS member_principal_name FROM sys.database_role_members rm JOIN sys.database_principals r ON rm.role_principal_id = r.principal_id JOIN sys.database_principals m ON rm.member_principal_id = m.principal_id` **EVIDENCIAS:**	**Media**
3. Las auditorías básicas deben estar activadas:	**Media**

Este documento es propiedad de SETI. Prohibida su reproducción total o parcial sin previa autorización del autor.

TAREA	CRITICIDAD
Los sistemas de auditoría de SQL Server nos permiten determinar las acciones realizadas por los usuarios en la instancia o a nivel de base de datos, esto se realiza a través d ella configuración de especificaciones de auditoria personalizadas. Se recomienda tener activas las auditorías mínimas de acceso a la instancia (Login, Logoff, Use Database, create object, drop object, entre otras) de forma de tener información a la hora de evaluar un posible atentado o riesgo operativo a nivel de seguridad. **NOTA:** Se debe evaluar con el DBA las capacidades de la instancia tanto a nivel de CPU, Memoria como de almacenamiento para poder tener las auditorias básicas configuradas. **AUDITORIA:** A. En la sección de seguridad de la instancia, ir al menú auditoria, validar si existe una auditoría configurada y la misma se encuentra en ENABLED. B. En la sección de seguridad de la base de datos, ir al menú de especificaciones de auditoría, validar si existe una auditoría configurada y la misma se encuentra en ENABLED. **EVIDENCIAS:**	
4. El puerto de comunicaciones de SQL Server no debe ser el Default Cuando se realiza una instalación del motor de base de datos SQL Server, el puerto por Default es el 1433, tomando en cuenta que esto es un valor conocido se recomienda	**Alta**

Este documento es propiedad de SETI. Prohibida su reproducción total o parcial sin previa autorización del autor.

TAREA	CRITICIDAD
establecer un puerto distinto o bien realizar la configuración mediante el uso de puertos dinámicos. **AUDITORIA:** A. Validar que el puerto de escucha del motor de base de datos no sea el 1433. `SELECT DISTINCT local_tcp_port FROM sys.dm_exec_connections WHERE local_tcp_port IS NOT NULL;` **EVIDENCIAS:**	

5. No deben existir usuarios huérfanos a nivel de las bases de datos
 	
La condición de usuarios huérfanos en una instancia SQL Server se da cuando no existe relación entre un usuario de base de datos y un login de la instancia, esto se puede generar debido a que se realizó una migración de las bases de datos a través de un backup/restore a una nueva instancia y los logins fueron creados via scripts o bien cuando se elimina un login y no se especifica la opción de tambien eliminar el usuario y su esquema a nivel de base de datos.
 	
AUDITORIA:
 	
A. Validar que no existen usuarios huérfanos a nivel de las bases de datos.
`exec sp_change_users_login @Action='Report';`
`select p.name,p.sid from sys.database_principals p where p.type in ('G','S','U') and p.sid not in (select sid from sys.server_principals) and p.name not in ( 'dbo', 'guest', 'INFORMATION_SCHEMA', 'sys', 'MS_DataCollectorInternalUser');`
	Baja

Este documento es propiedad de SETI. Prohibida su reproducción total o parcial sin previa autorización del autor.

TAREA	CRITICIDAD
EVIDENCIAS:	
6. El usuario SA debe estar deshabilitado Una vez se finalice la instalación y configuración de la instancia de SQL Server y sean definidos los usuarios administradores de la misma, se debe deshabilitar el usuario SA de la instancia, esto con el objetivo de evitar ataques tomando en cuenta que es de conocimiento general la existencia de este usuario en una instancia SQL Server. Nota: esta actividad solo se debe realizar siempre y cuando existan otros usuarios administradores definidos. **AUDITORIA:** A. Validar que el usuario SA esté deshabilitado en la instancia `SELECT name, type_desc, is_disabled FROM sys.server_principals where name = 'SA';` **EVIDENCIAS:**	**Alta**

D. VULNERABILIDADES A NIVEL DE LOS BACKUPS

Este documento es propiedad de SETI. Prohibida su reproducción total o parcial sin previa autorización del autor.

TAREA	CRITICIDAD

1. Las bases de datos deben tener respaldo full y del log de transacciones
 	
Para garantizar la recuperación de la base de datos en caso de perdida de información o daño total, se requiere tener configurado los respaldos tanto de la data como de log de transacciones acorde al política establecida por el negocio para el RTO y el RPO. Del mismo modo, a nivel técnico, se recomienda tener el respaldo del log de transacciones configurado si el modo de recuperación de la base de datos está en modo full de forma de controlar el crecimiento del mismo.
 	
AUDITORIA:
 	
A. Validar que las bases de datos tengan respaldos de tipo Full y Transaccion Log
`SELECT`
`  CONVERT(CHAR(100), SERVERPROPERTY('Servername')) AS Server,`
`  msdb.dbo.backupset.database_name,`
`  msdb.dbo.backupset.backup_start_date,`
`  msdb.dbo.backupset.backup_finish_date,`
`  msdb.dbo.backupset.expiration_date,`
`  CASE msdb..backupset.type`
`  WHEN 'D' THEN 'Database'`
	Alta

Este documento es propiedad de SETI. Prohibida su reproducción total o parcial sin previa autorización del autor.

TAREA	CRITICIDAD
WHEN 'L' THEN 'Log' END AS backup_type, msdb.dbo.backupset.backup_size, msdb.dbo.backupmediafamily.logical_device_name, msdb.dbo.backupmediafamily.physical_device_name, msdb.dbo.backupset.name AS backupset_name, msdb.dbo.backupset.description FROM msdb.dbo.backupmediafamily INNER JOIN msdb.dbo.backupset ON msdb.dbo.backupmediafamily.media_set_id = msdb.dbo.backupset.media_set_id WHERE (CONVERT(datetime, msdb.dbo.backupset.backup_start_date, 102) >= GETDATE() - 7) ORDER BY msdb.dbo.backupset.database_name, msdb.dbo.backupset.backup_finish_date **EVIDENCIAS:**	
2. Los respaldos full y de logs de las base de datos no deben quedar locales Como estrategia de respaldos de bases de datos (Full y Log de Transacciones), el DBA configura jobos que ejecuten de forma periódica respaldos bien sea a discos locales o hacia unidades de red. En el caso de que sea el primer escenario (discos locales), se debe garantizar que las copias sea movidas a una unidad de red en otro servidor o bien se tengan herramientas externas como Data Protector, TSM, Avamar, entre otras que garanticen la retención y la protección de los mismos.	**Alta**

Este documento es propiedad de SETI. Prohibida su reproducción total o parcial sin previa autorización del autor.

TAREA	CRITICIDAD
AUDITORIA: A. Validar los jobs de respaldo que escriban los mismos en unidades externas o bien sean tomados por herramientas como Data Protector, TSM, Avamar, entre otras. **EVIDENCIAS:**	

Este documento es propiedad de SETI. Prohibida su reproducción total o parcial sin previa autorización del autor.