مقدمه

در عصر حاضر، داده‌ها به عنوان یکی از ارزشمندترین دارایی‌های سازمان‌ها و کسب‌وکارها شناخته می‌شوند. حفاظت از این داده‌ها و تضمین امنیت آن‌ها، به یکی از اصلی‌ترین چالش‌ها در حوزه فناوری اطلاعات تبدیل شده است. پایگاه داده‌های SQL Server به عنوان یکی از پرکاربردترین سامانه‌های مدیریت پایگاه داده، نقش بسزایی در ذخیره، مدیریت و بازیابی داده‌ها دارد. با این حال، به دلیل ماهیت حساس اطلاعاتی که در این سیستم‌ها نگهداری می‌شود، همواره در معرض تهدیدات امنیتی قرار دارند.

با گسترش روزافزون حملات سایبری و تهدیدات داخلی، امنیت اطلاعات به یکی از دغدغه‌های اصلی مدیران فناوری اطلاعات تبدیل شده است. این تهدیدات ممکن است از منابع مختلفی مانند هکرهای خارجی، کاربران داخلی بدون مجوز، و یا بدافزارها ناشی شوند. عواقب این تهدیدات شامل سرقت اطلاعات حساس، خرابکاری در داده‌ها، ایجاد اختلال در خدمات و حتی از دست دادن اعتبار سازمانی است. این چالش‌ها نیازمند استفاده از راهکارهای نوآورانه و استانداردهای پیشرفته امنیتی هستند تا سطح اطمینان و اعتماد به سیستم‌های اطلاعاتی ارتقاء یابد.

پایگاه داده‌های SQL Server به دلیل ویژگی‌های منحصر به فرد و توانمندی‌های پیشرفته‌ای که ارائه می‌دهد، در بسیاری از سازمان‌ها و صنایع مختلف مورد استفاده قرار می‌گیرد. با این وجود، گستردگی قابلیت‌ها و پیچیدگی این سامانه‌ها می‌تواند منجر به وجود حفره‌های امنیتی و آسیب‌پذیری‌هایی شود که توسط مهاجمان مورد سوءاستفاده قرار گیرند. برای کاهش این مخاطرات، استانداردهای امنیتی نقش کلیدی ایفا می‌کنند. این استاندارد مجموعه‌ای از بهترین شیوه‌ها و دستورالعمل‌های کاربردی هستند که به سازمان‌ها در ایمن‌سازی سیستم‌های خود کمک می‌کنند.

استاندارد CIS یا Center for Internet Security Benchmark، یک چارچوب جامع امنیتی است که برای ارزیابی و مقاوم‌سازی سامانه‌های اطلاعاتی در برابر تهدیدات طراحی شده است. این استاندارد با ارائه فهرستی از تنظیمات پیشنهادی، سازمان‌ها را در بهبود امنیت و کاهش خطرات امنیتی یاری می‌دهد. این استاندارد در زمینه‌های مختلف مانند پیکربندی‌های امن، دسترسی‌ها، رمزنگاری، و نظارت امنیتی برای پایگاه‌های داده مختلف از جمله SQL Server، توصیه‌های مفیدی دارد.

هدف این تحقیق، طراحی و پیاده‌سازی ابزاری کاربردی برای مقاوم‌سازی پایگاه داده SQL Server با استفاده از استاندارد CIS است. این ابزار نه تنها وضعیت امنیتی فعلی پایگاه داده را تحلیل می‌کند، بلکه تنظیمات امنیتی مورد نیاز را نیز پیشنهاد داده و امکان اعمال این تنظیمات را فراهم می‌سازد.

در این پژوهش، ابتدا مفاهیم بنیادی امنیت پایگاه داده و استانداردهای مطرح در این حوزه بررسی خواهد شد. در ادامه، استاندارد CIS و ویژگی‌های آن در حوزه پایگاه داده‌های SQL Server به تفصیل مورد مطالعه قرار می‌گیرد. سپس با مرور کارهای پیشین، روش‌شناسی پیاده‌سازی ابزار پیشنهادی تشریح می‌شود. در نهایت، نتایج حاصل از پیاده‌سازی ابزار و تأثیرات آن بر امنیت پایگاه داده تحلیل خواهد شد.

اهمیت این پژوهش در ارائه یک ابزار عملیاتی و قابل استفاده برای سازمان‌ها نهفته است که می‌تواند به طور مؤثری مخاطرات امنیتی را کاهش داده و اعتماد سازمان‌ها به امنیت داده‌های خود را افزایش دهد.

فصل دو:  
دسته بندی امن سازی ها:

1. Installation, Updates, and Patches (نصب، به‌روزرسانی‌ها و پچ‌ها)

این بخش به نصب صحیح SQL Server و اعمال به‌روزرسانی‌های امنیتی مربوط می‌شود:

۱.۱: اطمینان از نصب آخرین به‌روزرسانی‌ها و پچ‌های امنیتی SQL Server

توصیه می‌شود که همیشه از آخرین نسخه و به‌روزرسانی‌های SQL Server استفاده شود تا از آسیب‌پذیری‌های امنیتی جلوگیری شود.

۱.۲: استفاده از سرورهای تک‌منظوره برای SQL Server

SQL Server باید روی سرورهایی که تنها برای این منظور اختصاص داده شده‌اند نصب شود تا سطح حمله کاهش یابد و مدیریت راحت‌تر شود.

۲. Surface Area Reduction (کاهش سطح حمله)

۲.۱: غیرفعال کردن پرس‌وجوهای توزیع شده فوری

این تنظیم برای جلوگیری از اجرای کدهای ناامن و دسترسی‌های غیرمجاز به داده‌های خارجی توصیه می‌شود.

۲.۲: غیرفعال کردن CLR

با این کار، امکان اجرای اسمبلی‌های CLR (که ممکن است خطرناک باشند) حذف می‌شود.

۲.۳: غیرفعال کردن Cross DB Ownership Chaining

این ویژگی امکان دسترسی بین پایگاه‌های داده را محدود می‌کند و خطر افشای اطلاعات را کاهش می‌دهد.

۲.۴: غیرفعال کردن Database Mail XPs

برای جلوگیری از ارسال ایمیل‌های ناخواسته و سوءاستفاده از سرور.

۲.۵: غیرفعال کردن Ole Automation Procedures

این ویژگی برای کاهش سطح حمله و جلوگیری از اجرای کدهای خارجی غیرفعال می‌شود.

۲.۶: غیرفعال کردن دسترسی از راه دور

برای کاهش احتمال حملات DoS از این ویژگی استفاده نمی‌شود.

۲.۷: غیرفعال کردن Remote Admin Connections

دسترسی‌های مدیریتی از راه دور را محدود می‌کند.

۲.۸: غیرفعال کردن Scan for Startup Procedures

جلوگیری از اجرای خودکار رویه‌ها هنگام راه‌اندازی برای افزایش امنیت.

۲.۹: تنظیم ویژگی Trustworthy به حالت خاموش

برای جلوگیری از سوءاستفاده از اسمبلی‌ها و رویه‌های گسترش‌یافته.

۲.۱۰: غیرفعال کردن پروتکل‌های غیرضروری

استفاده از حداقل پروتکل‌ها برای کاهش سطح حمله.

۲.۱۱: استفاده از پورت‌های غیر استاندارد

تغییر پورت پیش‌فرض برای جلوگیری از حملات شناخته‌شده.

۲.۱۲: مخفی کردن نمونه‌های SQL Server

از شناسایی سرورهای SQL در شبکه جلوگیری می‌کند.

۲.۱۳: غیرفعال کردن حساب کاربری sa

برای کاهش خطر حملات brute force.

۲.۱۴: تغییر نام حساب کاربری sa

جلوگیری از شناسایی این حساب شناخته‌شده.

۲.۱۵: تنظیم AUTO\_CLOSE به خاموش

برای بهبود عملکرد و جلوگیری از استفاده بیش از حد منابع سرور، باید این گزینه برای دیتابیس‌های حاوی فعال تنظیم شود

۲.۱۶: اطمینان از عدم وجود حساب با نام sa

حتی اگر حساب sa تغییر نام داده یا غیرفعال شده باشد، هیچ حساب دیگری نباید با این نام وجود داشته باشد.

۲.۱۷: تنظیم clr strict security به فعال

تضمین امنیت بیشتر با محدود کردن اسمبلی‌ها.

۳. Authentication and Authorization (تایید هویت و مجوزها)

این بخش به کنترل فرآیندهای تایید هویت و تخصیص مجوزها برای کاربران می‌پردازد

۳.۱: تنظیم احراز هویت به Windows Authentication

استفاده از حالت Windows Authentication برای احراز هویت کاربران سرور SQL، که امنیت بیشتری نسبت به روش احراز هویت SQL Server ارائه می‌دهد.

۳.۲: لغو مجوز کاربر 'guest'

حذف مجوز اتصال (CONNECT) برای کاربر 'guest' در تمام پایگاه‌های داده به‌جز پایگاه‌های اصلی (مانند master و tempdb) برای جلوگیری از دسترسی غیرمجاز.

۳.۳: حذف کاربران یتیم

کاربران پایگاه داده که دیگر به هیچ حساب کاربری مرتبط نیستند شناسایی و حذف می‌شوند تا از سوءاستفاده احتمالی جلوگیری شود.

۳.۴: عدم استفاده از احراز هویت SQL در پایگاه‌های مستقل

در پایگاه‌های داده مستقل (Contained Databases) فقط از احراز هویت ویندوز استفاده شود و استفاده از احراز هویت SQL ممنوع گردد.

۳.۵: محدود کردن دسترسی حساب MSSQL Service

حساب MSSQL Service نباید دارای دسترسی ادمین باشد و دسترسی‌های آن باید فقط به موارد ضروری محدود شود.

۳.۶: محدود کردن دسترسی SQLAgent Service

حساب SQLAgent Service نیز نباید دارای مجوزهای ادمین باشد تا سطح دسترسی سرور کاهش یابد.

۳.۷: محدود کردن دسترسی Full-Text Service

دسترسی‌های حساب Full-Text Service باید به حداقل ضروری کاهش یابد تا امنیت افزایش یابد.

۳.۸: بررسی و حذف مجوزهای اضافی نقش عمومی (public role)

اطمینان حاصل شود که نقش عمومی فقط مجوزهای پیش‌فرض سیستم را دارد و هیچ مجوز اضافی به آن اختصاص داده نشده است.

۳.۹: حذف دسترسی گروه‌های BUILTIN ویندوز به SQL Server

گروه‌های BUILTIN ویندوز نباید به‌صورت پیش‌فرض به SQL Server دسترسی داشته باشند تا امنیت کلی افزایش یابد.

۳.۱۰: حذف دسترسی گروه‌های محلی ویندوز به SQL Server

اطمینان از اینکه هیچ گروه محلی ویندوز به SQL Server دسترسی ندارد مگر اینکه به‌طور خاص تنظیم شده باشد.

۳.۱۱: محدود کردن نقش عمومی در msdb

نقش عمومی (public role) نباید در پایگاه msdb به پراکسی‌های SQL Agent دسترسی داشته باشد تا از سوءاستفاده جلوگیری شود.

۴. Password Policies (سیاست‌های رمز عبور)

این تنظیمات برای افزایش امنیت حساب های کاربری توصیه میشود.

۴.۱: فعال کردن گزینه 'MUST\_CHANGE' برای تمامی ورودهای SQL Authenticated

این تنظیم تضمین می‌کند که کاربران هنگام ایجاد حساب جدید یا بازنشانی رمز عبور، رمز عبور خود را تغییر دهند.

۴.۲: فعال کردن گزینه 'CHECK\_EXPIRATION' برای ورودهای دارای نقش sysadmin

این گزینه، انقضای رمز عبور را برای ورودهای SQL فعال می‌کند و از استفاده طولانی‌مدت از رمزهای عبور جلوگیری می‌نماید.

۴.۳: فعال کردن گزینه 'CHECK\_POLICY' برای تمامی ورودهای SQL Authenticated

این تنظیم باعث می‌شود که قوانین امنیتی ویندوز (مانند پیچیدگی رمز عبور و حداقل طول) برای ورودهای SQL اعمال شود.

۵. Auditing and Logging (ممیزی و ثبت لاگ‌ها)

این بخش به ثبت و نظارت بر فعالیت‌های سیستم SQL Server می‌پردازد:

۵.۱: تنظیم تعداد فایل‌های گزارش خطا (Error Log) به حداقل ۱۲

تعداد فایل‌های گزارش خطا باید به ۱۲ یا بیشتر تنظیم شود تا بتوان تاریخچه بیشتری از رویدادها را برای بررسی مشکلات نگهداری کرد.

۵.۲: فعال کردن گزینه 'Default Trace'

این گزینه گزارش‌های پیش‌فرض SQL Server را فعال می‌کند و امکان بررسی تغییرات مهم در سرور، مانند ایجاد یا حذف اشیا، را فراهم می‌سازد.

۵.۳: تنظیم 'Login Auditing' به 'failed logins'

این تنظیم فقط ورودهای ناموفق را ثبت می‌کند تا تلاش‌های غیرمجاز برای دسترسی شناسایی شوند.

۵.۴: استفاده از 'SQL Server Audit' برای ثبت ورودهای موفق و ناموفق

این ویژگی تمامی ورودهای موفق و ناموفق را ثبت می‌کند و اطلاعات جامعی از دسترسی‌ها ارائه می‌دهد.

۶. Application Development (توسعه نرم‌افزار)

این بخش به توسعه و پیاده‌سازی نرم‌افزارهای کاربردی امن که با SQL Server ارتباط دارند، می‌پردازد:

۶.۱: اطمینان از سانیتیزه کردن ورودی‌های کاربر در پایگاه داده و برنامه‌های کاربردی

ورودی‌های کاربر باید به‌طور کامل بررسی و تصحیح شوند تا از حملات احتمالی مانند SQL Injection جلوگیری شود.

۶.۲: تنظیم سطح مجوز CLR Assembly Permission Set به SAFE\_ACCESS برای تمامی اسمبلی‌های CLR

برای افزایش امنیت، باید تمامی اسمبلی‌های CLR با سطح مجوز SAFE\_ACCESS اجرا شوند و از اجرای اسمبلی‌های خطرناک (UNSAFE یا EXTERNAL\_ACCESS) جلوگیری شود.

۷. Encryption (رمزنگاری)

این بخش به استفاده از رمزنگاری برای حفاظت از داده‌ها در پایگاه داده و در طول انتقال می‌پردازد:

۷.۱: استفاده از الگوریتم رمزنگاری کلید متقارن AES\_128 یا بالاتر در پایگاه‌های داده غیرسیستمی

این تنظیم برای محافظت از داده‌ها با استفاده از الگوریتم‌های رمزنگاری قوی‌تر مانند AES\_128 توصیه می‌شود.

۷.۲: استفاده از اندازه کلید نامتقارن حداقل ۲۰۴۸ بیتی در پایگاه‌های داده غیرسیستمی

برای امنیت بیشتر، کلیدهای رمزنگاری نامتقارن باید حداقل ۲۰۴۸ بیت باشند.

۷.۳: رمزنگاری پشتیبان‌های پایگاه داده

پشتیبان‌ها باید رمزنگاری شوند تا در صورت دسترسی غیرمجاز به پشتیبان‌ها، داده‌ها محافظت شوند.

۷.۴: پیکربندی و فعال‌سازی رمزنگاری شبکه

شبکه باید برای انتقال امن داده‌ها رمزنگاری شود تا اطلاعات در حال انتقال ایمن باقی بمانند.

۷.۵: استفاده از رمزنگاری Transparent Data Encryption (TDE) برای پایگاه‌های داده

استفاده از TDE برای رمزنگاری داده‌های پایگاه‌داده در حین ذخیره‌سازی، برای محافظت در برابر دسترسی غیرمجاز به داده‌های ذخیره‌شده.

8- سایر ملاحظات:

۸.۱: اطمینان از پیکربندی صحیح سرویس SQL Server Browser

سرویس SQL Server Browser باید به‌درستی پیکربندی شود تا اطمینان حاصل شود که نمونه‌های SQL Server به درستی شناسایی می‌شوند و از دسترسی‌های غیرمجاز جلوگیری می‌شود.

موارد کاربرد:

1. مشاهده هر یک از آیتم های ارزیابی و نتیجه آن
2. آگاهی از توضیحات هر آیتم ، شامل :

* خودکار یا دستی بودن نوع امن سازی

1. اجرای امن سازی

ساختار طبقه بندی هر ارزیابی بر اساس نحوه پیاده سازی:

|  |  |
| --- | --- |
| کد ارزیابی | نحوه ارزیابی |
| 0 | این نوع باید بصورت دستی کنترل شود و پرسمان ندارد |
| 1 | یک ردیف با چند ستون که مقادیر آن باید 0 یا 1 باشند |
| 2 | هیچ مقداری نباید برگردانده شود |
| 3 | مقدار برگشتی باید از مقدار ثابتی بزرگتر باشد |
| 4 | مقدار برگشتی شامل چند ردیف است که مقادیر خاصی باید در آن وجود داشته باشد. |
| 5 | مقدار برگشتی نباید یک مقدار ثابت باشد |
| 6 | یک یا چند ردیف که نباید مقدار یکی از فیلدهایشان یک مقدار ثابت یاشد. |
| 7 | یک یا چند ردیف که باید مقدار یکی از فیلدهایشان یک مقدار ثابت یاشد. |

ساختار طبقه بندی هر امن سازی بر اساس نحوه پیاده سازی:

|  |  |
| --- | --- |
| کد امن سازی | نحوه امن سازی |
| 0 | این نوع باید بصورت دستی کنترل شود و پرسمان ندارد. |
| 1 | پرسمان از نوع ساده. |
| 2 | کاربر باید با توجه به مقادیری که در پرسمان ارزیابی برگردانده میشود، مقداری را در پرسمان مقاوم سازی واردکند. (مانند username ، login\_name) |

پیکربندی و ورودی اطلاعات  
فایل ورودی اطلاعات در این برنامه .csv نام دارد که ساختار آن مطابق با جدول 1 است.

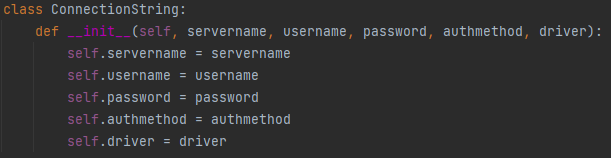
ساختار فایل csv:

|  |  |  |
| --- | --- | --- |
| ردیف | نام ستون | توضیحات |
| 0 | No. | شماره ردیف |
| 1 | Index | شماره امن سازی بر اساس دسته بندی CIS |
| 2 | Dsc | توضیحات |
| 3 | AuditQuery | پرسمان ارزیابی |
| 4 | AuditType | نوع پرسمان ارزیابی |
| 5 | ReturnNumbers | تعداد مقادیر برگشتی |
| 6 | Val1 | مقدار برگشتی 1 |
| 7 | Val2 | مقدار برگشتی 2 |
| 8 | Val3 | مقدار برگشتی 3 |
| 9 | RemType | نوع پرسمان امن سازی |
| 10 | numberOfTransitions | تعداد ترنزاکسن ها |
| 11 | ReplacementName | نام متغییر جایگزین |
| 12 | Tsql1 | ترنز اکشن 1 |
| 13 | Tsql2 | ترنزاکشن 2 |
| 14 | Tsql3 | ترنز اکشن 3 |
| 15 | title | عنوان |

جدول 1: ساهتار فایل

ساختار کلاس ها:

کلاس ConnectionString :  
در شکل 1-4مشخصات این کلاس مشاهده میشود، هر نمونه از این کلاس در واقعا اطلاعات مربوط به نام دیتابیس و نام کاربری و روش ورود را نگه داری میکند.



شکل 1-4

کلاس Login :

مشخصات این کلاس را در شکل 2-4 مشاهده میشود. این کلاس برای ساخت فرم ورود استفاده می‌شود.



شکل 2-4

توابع:   
loginControlFrame(): این تابع فرم ورود را میسازد.

detect\_driver() درایور های sql را شناسایی میکند

UserPassEnable() : غیرفعال کردن ورود username و password در حالتی که روش ورود Windows Authetication باشد.

Connect() ساخت connectionstring

کلاس AdminControls:

مشخصات این کلاس در شکل 3-4 مشاهده میشود، این کلاس برای نگهداری پارامترهای بدنه اصلی فرم، شامل دکمه ها، ورودی ها و جدول نتایج استفاده میشود.



شکل 3-4

توابع:  
adminControlsFrame() :   
tableOutputFrame() :

logout(): برای قطع ارتباط با دیتابیس، خروج از صفحه اصلی و ورود به صفحه لاگین استفاده میشود. این تابع با کلیک کردن بر روی دکمه *Logout* فراخوانی میشود.

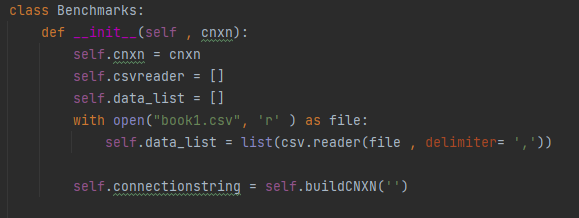
viewResults(): لیست پیشنهادات و نتایج آنها را نمایش/ بروز رسانی میکند. این تابع با کلیک کردن بر روی دکمه *View List* فراخوانی میشود.

getData(): اطلاعات مربوط به هر یک از موارد امن سازی را در قسمت *Description* نمایش میدهد ، این تابع با کلیک کردن بر روی هر ردیف از موارد امن سازی (بخش *List of Remediations*) فراخوانی میشود.

runRemediation() : اجرای امن سازی و نمایش نتایج آن در *remediation result .* این تابع با کلیک کردن بر روی دکمه *run remediation* فراخوانی میشود.

کلاس Bemchmarks:

مشخصات این کلاس در مشاهده میشود، این کلاس و توابع آن به طور کلی برای ارتباط با پایگاه داده ، خواندن فایل و اجرای پرسمان است.



شکل 4-4

توابع:

buildCNXN() : اتصال به پایگاه داده با تنظیمات داده شده و از طریق pyodbc .

get\_databases() : دریافت لیست دیتابیس ها جهت نمایش به کاربر.

handle\_sql\_variant\_as\_string(): خروجی را به فرمت مناسب تبدیل میکند.

runAudit(): برای اجرای پرسمان مربوط به ارزیابی ها

runRemediation() : برای اجرای پرسمان مربوط به مقاوم سازی ها.