بسم الله الرحمن الرحیم

آموزشگاه تحلیل داده

تخصصی ترین مرکز برنامه نویسی و دیتابیس در ایران

مدیریت مجوزها با استفاده از stored procedure ها یا توابع کاربر

مدرس : مهندس افشین رفوآ

دوره آموزش SQL Server Administration

مدیریت مجوزها با استفاده از stored procedure ها یا توابع کاربر

یکی از روش های ایجاد چندین خط دفاعی برای پایگاه داده ی خود پیاده سازی تمامی data access ها (دسترسی به اطلاعات) با استفاده از stored procedure ها یا توابع user-defined می باشد. شما می توانید تمامی مجوزهای دسترسی به اشیا زیرین همچون جداول را لغو (revoke/deny) نموده یا مجوز EXECUTE بر روی stored procedure ها اعطا کنید. این کار در واقع یک خط یا محوطه ی امنیتی در اطراف اطلاعات و اشیا پایگاه داده ی شما ایجاد کرده و ایمنی آن ها را تضمین می کند.

مزایای استفاده از stored procedure

با بکارگیری stored procedure ها می توان منطق اطلاعات و قوانین business را کپسوله سازی نمود تا بدین وسیله کاربران تنها بتوانند به آن شیوه ای که مدنظر مدیران پایگاه داده یا توسعه دهندگان است به داده ها و اشیا دسترسی داشته باشند.

Stored procedure های پارامتری (Parameterized) که تمامی ورودی های کاربر را اعتبارسنجی می کند را می توان جهت خنثی سازی حملات SQL injection مورد بهره وری قرار داد. در صورت استفاده از SQL پویا، ابتدا باید دستورهای خود را پارامتری نموده و همچنین از گنجاندن مقادیر پارامتر به طور مستقیم در query string اجتناب کنید.

Query های ویژه (ad hoc queries) و اصلاحاتی که به داده ها اعمال می شود را می توان رد کرد. این کار مانع از آن می شود که کاربران به طور عمد و یا تصادفی اطلاعات را نابود ساخته یا query هایی اجرا کنند که کارایی سرویس دهنده یا شبکه را مختل سازد.

می توان خطاها را بدون ارسال مستقیم آن ها به برنامه های سمت کاربر (client applications) در کد پروسیجر مدیریت کرد. این امر مانع از ارسال (بازگردانی) پیام های خطا به کاربر می شود. کاربر با بهره وری از پیام های خطای بازگشتی می تواند شکاف ها و نقاط آسیب پذیر در امنیت سیستم را پیدا کند. خطاها را ثبت گزارش (log) کرده و آن ها را در سرور مدیریت کنید.

می توان stored procedure ها را یکبار نوشته و آن ها را توسط برنامه های متعدد مورد دسترسی قرار داد.

لازم نیست برنامه های سمت کاربر اطلاعاتی درباره ی ساختار زیرین و اصلی داده ها داشته باشد. کد stored procedure ها را می توان بدون نیاز به اعمال تغییرات در برنامه های سمت کاربر اصلاح کرد، به این شرط که تغییرات اعمال شده لیست های پارامتر یا نوع داده های بازگشتی را تحت تاثیر قرار ندهند.

Stored procedure ها قادر هستند با ادغام چندین عملیات در یک procedure call (فراخوانی پروسیجر) از ترافیک شبکه بکاهد.

اجرای stored procedure ها

Stored procedure ها با بهره گیری از زنجیره ی مالکیت دسترسی به داده ها را امکان پذیر می سازد تا بدین وسیله کاربر برای دستیابی به اشیا پایگاه داده دیگر به مجوز صریح نیاز نداشته باشد.
Ownership chain یا زنجیره ی مالکیت به زمانی اشاره دارد که اشیایی که به صورت پی در پی (متوالی) به یکدیگر دسترسی دارند تحت مالکیت کاربری یکسان قرار می گیرد. به عنوان مثال، یک stored procedure می تواند stored procedure های دیگر را صدا بزند و یا یک stored procedure به چندین جدول دسترسی پیدا کند. چنانچه تمامی اشیای موجود در زنجیره ی اجرا دارای مالک یکسان باشند، در آن صورت SQL Server صرفا مجوزهای EXECUTE بر روی stored procedure را مورد بررسی قرار می دهد. از این رو، شما فقط باید مجوزهای EXECUTE بر روی stored procedureها را

اعطا نمایید؛ شما می توانید کلیه ی مجوزهای دسترسی بر روی جداول زیرین را لغو (revoke یا deny) کنید.

بهترین روش ها

دقت داشته باشید که صرف نوشتن stored procedure ایمنی برنامه ی کاربردی شما تضمین نمی گردد، بلکه در تامین امنیت برنامه ی خود بایستی حفره های احتمالی زیر را در نظر گرفته و برطرف سازی آن ها را در دستور کار خود قرار دهید.

مجوزهای EXECUTE را به stored procedure ها و برای آن دسته از نقش های کاربری پایگاه داده اعطا نمایید که مایلید به اطلاعات دسترسی داشته باشند.

تمامی مجوزهای دسترسی به جداول زیرین برای کلیه نقش ها و کاربران در پایگاه داده از جمله نقش public را لغو (deny یا revoke) کنید. همه ی کاربران از نقش کاربری public مجوزها را به ارث می برند. بنابراین deny کردن مجوزهای دسترسی به public بر این دلالت دارد که تنها مالکان و عضوهای sysadmin اجازه ی دسترسی خواهند داشت؛ سایر کاربران از به ارث بردن مجوز به واسطه ی عضویت در دیگر نقش ها ناتوان خواهند بود (سایر کاربران قادر نخواهند بود با عضویت در دیگر نقش ها مجوزهای مورد نظر را به ارث ببرند).

از افزودن کاربران و نقش های کاربری به sysadmin یا db_owner خودداری کنید. مدیران سیستم و مالکان پایگاه داده اجازه دارند به تمامی اشیا پایگاه داده دسترسی داشته باشند.

حساب کاربری guest را غیرفعال نمایید. این کار مانع از اتصال کاربران ناشناس به بانک اطلاعاتی مد نظر می شود. البته حساب کاربری مزبور در تمامی پایگاه داده های نوین به صورت پیش فرض غیر فعال می باشد.

پیاده سازی مدیریت و ثبت خطاها را در دستور کار خود قرار دهید.

Stored procedure های پارامتری ایجاد نمایید که تمامی ورودی های کاربر را اعتبارسنجی می کند. توصیه می شود کلیه ی ورودی های کاربر را غیرقابل اطمینان در نظر گرفته و با آن ها به عنوان ناامن برخورد کنید.

تا احد امکان از بکار بردن SQL پویا (dynamic) خودداری کنید. با استفاده از تابع Transact-SQL QUOTENAME() مقدار یک رشته را محصور کرده و از وجود هر گونه نمونه ی محصورکننده (delimiter) در رشته ی ورودی (input string) اجتناب کنید.