آزمایشگاه تخصصی آپا در حوزه امنیت سرویس‌های شبکه و تجهیزات بی‌سیم

c e r t @ u m . a c . i r

امنیت در SQL Server 2005

(قسمت اول)

فتانه زرین کلام

fatane.zarrinkalam@stu-mail.um.ac.ir

عالیه سعیدی

álieh.saeedi@stu-mail.um.ac.ir

ویرایش اول – شهریورماه ۱۳۸۹

شماره سند: APA_FUM_W_SQL_0058

چکیده: در این مقاله معماری امن سرویس‌دهنده‌ی SQL Server 2005 شرح داده شده است. این معماری مبتنی بر اعتبارها، امن‌شدنی‌ها و مجوزها می‌باشد. اعتبارها، حساب‌های امنیتی هستند که به سیستم دسترسی دارند و به دو سطح ویندوز و سرویس‌دهنده تقسیم می‌شوند. امن‌شدنی‌ها منابع سیستم از قبیل جدول‌ها، رویه‌ها، فایل‌ها و غیره هستند که باید محافظت شوند. امن‌شدنی‌ها نیز همانند اعتبارها، به دو سطح ویندوز و سرویس‌دهنده‌ی SQL تقسیم می‌شوند. مجوزها به اعتبارها اجازه‌ی اجرای عمل خاصی را روی امن‌شدنی‌ها می‌دهند و در دو سطح سرویس‌دهنده و پایگاه داده‌ها اعمال می‌شوند.

واژه‌های کلیدی: امنیت، پایگاه داده‌ها، SQL Server 2005، اعتبار، امن‌شدنی، مجوز، ورود

۱- مقدمه

هر دسترسی به داده‌ها و پردازه‌ها در یک پایگاه داده‌ها باید محدود به افرادی باشد که به آن‌ها نیاز دارند و مدیر پایگاه داده‌ها باید بتواند از هر عملی که توسط هر فردی در سیستم انجام شده است، آگاه شود. هر پایگاه داده‌ها دارای سیاست‌های امنیتی تعیین شده و مستند می‌باشد. این سیاست‌ها به تحلیل مقادیر، میزان حساسیت داده‌ها و پردازه‌های برنامه‌های کاربردی آن وابسته است. مدل امنیتی سرویس‌دهنده‌ی SQL به منظور ایجاد انعطاف‌پذیری لازم برای پیاده‌سازی انواع مختلف سیاست‌های امنیتی طراحی شده است و با معماری‌های برنامه‌های کاربردی مختلف فعلی سازگار می‌باشد. افراد و برنامه‌های کاربردی برای اتصال به پایگاه داده‌ها به یک یا چندین ورود[1] و یا عضویت در یک *ورود گروهی*[2] نیاز دارند. یک برنامه‌ی کاربردی ساده ممکن است نیاز به یک ورود برای استخراج داده‌هایش از پایگاه داده‌ها داشته باشد، اما یک برنامه‌ی کاربردی با داده‌های مختلف حساس، شخصی و مالی ممکن است نیازمند یک سلسله مراتب قوی از انواع ارتباطات با پایگاه داده‌ها باشد.

هر شخص که با یک ورود به سرویس‌دهنده دسترسی پیدا می‌کند، برای اتصال به هر پایگاه داده‌های آن سرویس‌دهنده به یک نام کاربری یا نام مستعار نیاز دارد. به عبارتی، به عنوان کاربر آن پایگاه داده‌ها نیز باید ثبت نام شده باشد. به علاوه، آن کاربر برای دسترسی به اشیای مختلف آن پایگاه داده‌ها مانند جداول، رویه‌ها، دیدها و غیره و یا اجرای کدهایی که ساختار آن پایگاه داده‌ها را تغییر می‌دهند، نیاز به مجوز دارد. معمولا به هر کاربر یک نقش[3] نسبت داده می‌شود که به هر نقش مجوزهایی داده شده است. برنامه‌ی کاربردی توسط نقش‌های مختلف کاربران مورد استفاده قرار می‌گیرد. اعضای هر نقش نیازمندی‌های مشابهی دارند. هر نقش، وابسته به عملی که در سازمان انجام می‌دهد نیاز به انواع مختلفی از دسترسی به پایگاه داده‌ها را دارد. بنابراین هر سرویس‌دهنده‌ی پایگاه داده‌ها امنیت را در دو سطح پایگاه داده‌ها و سرویس‌دهنده مدیریت می‌کند. مالک یک پایگاه داده‌های خاص، قادر به کنترل دسترسی‌ها به پایگاه داده‌ها توسط *سیستم مجوز*[4] می‌باشد و تنها مدیر سیستم قادر به لغو این عمل می‌باشد.

امنیت سرویس‌دهنده‌ی SQL Server 2005 شامل بخش‌های مختلفی است که سند حاضر معماری امن سرویس‌دهنده‌ی SQL Server 2005 را در قالب سه اصل اعتبارها[5]، امن‌شدنی‌ها[6] و مجوزها[7] مورد بررسی قرار می‌دهد و دیگر جنبه‌های امنیتی در نظرگرفته شده در آن مانند پیکربندی امن، *رمزگ داده‌ها*[8]، *سیاست‌های کلمه‌ی عبور*[9]، *امنیت فراداده‌ها*[10] و غیره در مقالات بعدی مورد بررسی قرار خواهد گرفت.

---

[1] Login
[2] Group login
[3] Role
[4] Permission system
[5] Principals
[6] Securables
[7] Permissions
[8] Data encryption
[9] Password policies
[10] Metadata security

۲- معماری SQL Server 2005

- امنیت در SQL server 2005 مبتنی بر اعتبار، امن‌شدنی‌ها و مجوزها می‌باشد.
- اعتبارها، حساب‌های امنیتی هستند که به سیستم دسترسی دارند.
- امن‌شدنی‌ها، منابع سیستم هستند که باید محافظت شوند.
- مجوزها به اعتبارها اجازه اجرای عمل خاصی را روی امن‌شدنی‌ها می‌دهند.

۲-۱- اعتبارها

اعتبارها به سه سطح ذیل تقسیم می‌شوند:

سطح ویندوز

اعتبارها در این سطح شامل گروه‌های ویندوز[1]، حساب‌های کاربری دامنه[2] و حساب‌های کاربری محلی[3] می‌باشند.

سطح سرویس‌دهنده‌ی SQL

این سطح شامل ورودها و نقش‌های سرویس‌دهنده می‌باشد. ورودها در ادامه و نقش‌ها به تفصیل در بخش ۲-۱-۱ مورد بررسی قرار خواهند گرفت. ورودها می‌توانند ورود ویندزو یا ورود سرویس‌دهنده‌ی SQL باشند که هر دو می‌توانند به نقش‌های سرویس‌دهنده نسبت داده شوند. به این ترتیب، مدیریت کاربرانی که مجوزهای یکسان دارند، آسان می‌شود. به طور پیش‌فرض، ورود ویندوز فعال است. حساب-های ویندوز به ورودهای ویندوز نگاشت می‌یابند. کلمه‌های عبور برای ورود به ویندوز، توسط ویندوز اعتبارسنجی و توسط سیاست‌های مربوط به حساب‌های ویندوز محدود می‌شوند. این سیاست‌ها توسط ویندوز مدیریت می‌شوند و محدودیت‌هایی روی پیچیدگی، تاریخ انقضا و غیره بر روی کلمات عبور اعمال می‌کنند. در سرویس‌دهنده‌ی SQL Server 2005، کلمه‌های عبور برای ورود سرویس‌دهنده‌ی SQL، توسط سرویس‌دهنده‌ی SQL اعتبارسنجی و توسط سیاست‌های کلمه‌ی عبور، محدود می‌شوند. سیاست‌های کلمه عبور با دستور CREATE LOGIN تنظیم می‌شوند.

سطح پایگاه داده‌ها

این سطح شامل کاربران، نقش‌های پایگاه داده‌ها و نقش‌های برنامه‌ی کاربردی می‌باشد. ورودها به کاربران پایگاه داده‌ها نگاشت می‌یابند و کاربران می‌توانند به یک یا چند نقش پایگاه داده‌ها اضافه شوند. نقش‌های برنامه‌های کاربردی برای تنظیم یک متن امنیتی جایگزین مبتنی بر برنامه‌ی کاربردی مشتری استفاده می‌شوند. جدا بودن سطح پایگاه داده‌ها از سطح سرویس‌دهنده‌ی SQL شامل مزایای زیر می‌باشد:

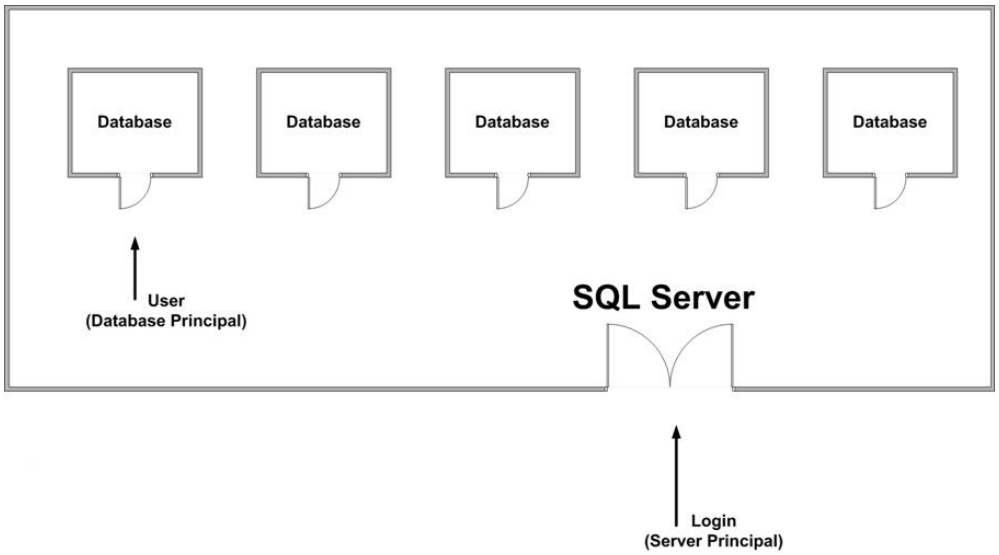
[1] Widows groups
[2] Domain user accounts
[3] Local user accounts

الف) پایگاه داده‌ها می‌تواند از یک سرویس‌دهنده‌ی SQL جدا شود و به یک سرویس‌دهنده‌ی SQL دیگر اضافه شود. در این صورت، کاربرها باید دوباره به ورودها نگاشت داده شوند که این کار به صورت دستی انجام می‌گیرد. برای این کار در SQL Server 2005 از دستور ALTER USER استفاده می‌شود.

ب) وقتی پایگاه‌های داده‌ها به صورت موجودیت‌های مستقل در نظر گرفته شوند، برنامه‌های کاربردی در پایگاه داده‌های مربوط به خود قرار می‌گیرند. در نتیجه دسترسی ناخواسته و مخرب به وسیله‌ی برنامه‌های کاربردی دیگر کمتر صورت می‌گیرد.

ج) مجوزها در دو حوزه‌ی سرویس‌دهنده و پایگاه داده‌ها، اعطا می‌شوند. مجوزهای داده شده به سرویس‌دهنده‌ی مستقل از پایگاه داده‌های فعلی اثر می‌کند ولی مجوزهای مربوط به یک پایگاه داده‌های خاص فقط در همان پایگاه داده‌ها موثر واقع می‌شود.

مفاهیم کاربر (User) و ورود (Login) با هم متفاوتند و نباید به جای یکدیگر استفاده شوند. ورود، اعتبار سرویس‌دهنده و کاربر، اعتبار ویندوز است. برای ایجاد ورود از دستور CREATE LOGIN و برای ایجاد کاربر از دستور CREATE USER استفاده می‌شود. ورود، اجازه‌ی اتصال به سرویس‌دهنده را می‌دهد ولی کاربر، به یک ورود وابسته است و مجوزهای خاصی را روی پایگاه داده‌ها می‌دهد. ورود سرویس‌دهنده‌ی SQL که به کاربر پایگاه داده‌ها نگاشت داده نشده است، تنها می‌تواند وارد سرویس‌دهنده‌ی SQL شود و به هر پایگاه داده‌هایی که به *کاربر مهمان*[1] اجازه‌ی ورود داده است، دسترسی پیدا کند. هر کاربر پایگاه داده‌ها باید به یک ورود سرویس‌دهنده‌ی SQL وابسته باشد. به عبارتی دیگر، کاربر پایگاه داده‌ها بدون ورود سرویس‌دهنده‌ی SQL غیرممکن است. شکل ۱ این تفاوت را نشان می‌دهد:



شکل۱- تفاوت کاربر و ورود

[1] Guest

۲-۱-۱- نقش‌های موجود در SQL Server 2005

نقش‌ها، گروه‌هایی هستند که برای گروه‌بندی کاربرانی که نیازهای دسترسی مشابه دارند، استفاده می‌شوند. گروه‌ها به سه دسته‌ی ذیل تقسیم می‌شوند:

- **نقش‌های سرویس‌دهنده**

این نقش‌ها در سطح سرویس‌دهنده تعریف شده‌اند و امکان حذف، افزودن و اصلاح آن‌ها وجود ندارد. ۸ نقش ثابت این دسته عبارتند از:

bulkadmin: اعضای این نقش، مجوز اجرای دستور BULK INSERT را دارند.

sysadmin: اعضای این نقش، مجوز انجام هر عملی در سرویس‌دهنده و کنترل کامل روی پایگاه‌های داده‌های آن را دارا می‌باشد.

serveradmin: اعضای این نقش، مجوز پیکربندی مشخصات سرویس‌دهنده و خاموش کردن آن را دارا می‌باشند.

setupadmin: اعضای این نقش، مجوز مدیریت *سرویس‌دهنده‌های متصل*[1] را دارند.

securityadmin: اعضای این نقش، مجوز ایجاد ورودهای سرویس‌دهنده و مدیریت مشخصات آن‌ها را دارند.

processadmin: اعضای این نقش، مجوز خاتمه‌ی فرآیندهای در حال اجرا در سرویس‌دهنده‌ی SQL را دارند.

dbcreator: این گروه مجوز ایجاد، تغییر و بازیابی پایگاه‌های داده‌ها در سرویس‌دهنده را دارد.

diskadmin: این گروه مجوز ایجاد و مدیریت فایل‌ها در دیسک را دارد.

برای بازیابی این نقش‌ها می‌توان از روال ذخیره شده‌ی sp_helpsrvrole استفاده کرد. نحو آن به صورت کامل در ذیل آمده است که در آن @srvrolename نام نقش ثابت سرویس‌دهنده است.

```
sp_helpsrvrole [[@srvrolename =] 'role']
```

- **نقش‌های پایگاه داده‌ها**

این نقش‌ها در سطح پایگاه داده‌ها تعریف می‌شوند و به سه دسته‌ی کلی ذیل تقسیم می‌شوند:

الف) نقش‌های ثابت: در سطح پایگاه داده‌ها تعریف می‌شوند و امکان حذف، افزودن و اصلاح آن‌ها وجود ندارد. ۹ نقش ذیل جزء نقش‌های ثابت می‌باشند:

db-owner: اعضای این نقش، مجوز انجام هر عملی در پایگاه داده‌ها را دارند.

db-accessadmin : اعضای این نقش، مجوز اضافه و حذف کاربران ویندوز NT و کاربران سرویس‌دهنده‌ی SQL را در پایگاه داده‌ها دارا می‌باشند.

[1] Linked Servers

db-datareader: اعضای این نقش، مجوز مشاهده‌ی تمام جداول کاربری را دارند. (اجرای دستور SELECT روی تمام جداول و دیدها)

db-datawriter: اعضای این نقش، مجوز اضافه کردن، تغییر و حذف داده‌های تمام جداول کاربری پایگاه داده‌ها را دارند.

db-ddladmin: اعضای این نقش، مجوز ایجاد *فرمان‌های زبان تعریف داده*[1] را دارند.

db-securityadmin: اعضای این نقش، مجوز مدیریت دستورات و مجوزهای اشیای پایگاه داده‌ها را دارند.

db-backupoperator: اعضای این نقش، مجوز تهیه‌ی نسخه‌ی پشتیبان از پایگاه داده‌ها را دارند.

db-denydatareader: اعضای این نقش، مجوز سلب مجوز انتخاب (SELECT) از جداول پایگاه داده‌ها را دارند.

db-denydatawriter: اعضای این نقش، اجازه‌ی سلب مجوز تغییر داده‌های جداول پایگاه داده‌ها را دارند.

برای اضافه کردن یک نقش پایگاه داده‌ها به پایگاه داده‌های جاری، می‌توان از روال ذخیره شده‌ی سیستمی sp_addrolemember استفاده کرد. نحو آن به‌صورت ذیل است که در آن @rolename نام نقش پایگاه داده‌ها و @membername نام حساب امنیتی است.

sp_addrolemember [@rolename =] 'role', [@membername =] 'security_account'

ب) نقش‌های عمومی: نقش عمومی، یک نوع نقش ویژه‌ی پایگاه داده‌ای است که تمام کاربران پایگاه داده‌ها عضوی از آن هستند. این نقش به هنگام ایجاد پایگاه داده‌ها ایجاد می‌شود. فایده‌ی این نقش زمانی مشخص می‌شود که قرار است مجموعه‌ای از مجوزهای پیش‌فرض به تمام کاربران داده شود. امکان حذف کردن این نقش وجود ندارد.

ج) نقش‌های پایگاه داده‌های تعریف شده توسط کاربر: این نقش‌ها امکان گروه‌بندی کاربرانی که به توابع خاصی از پایگاه داده‌ها دسترسی دارند را فراهم می‌کند.

برای ایجاد این نقش از روال ذخیره شده‌ی سیستمی ذیل استفاده می‌شود که در آن @rolename نام نقش پایگاه داده‌ها و @ownername نام مالک نقش جدید است.

sp_addrole [@rolename =] 'role' [,[@ownername =] 'owner']

برای حذف یک نقش می‌توان از sp_droprole [@rolename =] 'role' استفاده کرد.

- **نقش‌های برنامه‌ی کاربردی**

این نقش‌ها برای تامین امنیت لازم یک برنامه‌ی کاربردی ایجاد می‌شوند وکاربران را مجاب می‌کند تا از طریق برنامه‌ی کاربردی به داده‌های پایگاه داده‌ها دستیابی پیدا کنند. برای ایجاد یک نقش از این نوع، از نحو ذیل استفاده می‌شود که در آن

[1] Data Definition Language Command

@rolename نام نقش برنامه‌ی کاربردی و @password کلمه‌ی عبور برای نقش کاربردی جدید است.

sp_addapprole [@rolename =] 'role', [@password =] 'password'

برای حذف این نوع نقش از sp_dropapprole [@rolename =] 'role' استفاده می‌شود.

۲-۱-۲- اعتبارهای خاص

اعتبارهای خاص عبارتند از:

- **sa**: یک حساب مدیریتی برای ورود به سرویس‌دهنده‌ی SQL است که در صورت فعال بودن mixed authentication، قابل استفاده است. معمولا افراد برای این حساب از کلمات عبور ضعیف (اغلب بدون کلمه‌ی عبور) استفاده می‌کنند که این سبب آسیب‌پذیر شدن سیستم در برابر حمله‌کنندگان می‌شود. این حساب مدیریتی، یکی از اعضای از پیش تعریف شده برای نقش sysadmin است. بنابراین باید از یک کلمه‌ی عبور قوی برای آن استفاده کرد. چون sa از پیش تعریف شده است، اغلب حمله‌کنندگان در ابتدا سعی در ورود به سیستم از طریق این حساب می‌کنند. لازم به ذکر است که مجوزهای sa قابل حذف نمی‌باشند. در سرویس‌دهنده‌ی SQL Server 2005 ویژگی‌های امنیتی ذیل برای ایمن کردن این حساب در برابر حمله-کنندگان در نظر گرفته شده است:

۱- اگر از sa استفاده نشود اما قصد استفاده از SQL Authentication وجود داشته باشد، می‌توان حساب sa را با استفاده از فرمان ALTER LOGIN … DISABLE غیرفعال کرد. تا زمانی که یک ورود غیرفعال است، دسترسی به سرویس‌دهنده‌ی SQL از طریق آن امکان‌پذیر نیست.

۲- برای کلمه‌ی عبور حساب sa به طور پیش‌فرض باید از بررسی سیاست‌های کلمه‌ی عبور پیروی کرد. این نکته قابل توجه است که بررسی سیاست‌های کلمه‌ی عبور درصورتی قابل اعمال است که از سیستم عامل ویندوز 2003 یا ویندوز Vista استفاده شود. این سیاست‌ها کلمه‌ی عبور را در برابر حملات به روش Brute Force ایمن می‌کنند.

۳- چنانچه تلاش‌های مکرر ناموفق برای ورود به حساب sa صورت گرفته باشد، می‌توان آن را تغییر نام داد. با این کار حمله‌کننده متوقف می‌شود، زیرا ابتدا باید نام حساب را برای ورود پیدا کند. برای این منظور می‌توان از دستور ALTER LOGIN … WITH NAME استفاده کرد.

- **dbo**: معمولا ابتدا کاربر ایجاد می‌شود، سپس به یک ورود نگاشت پیدا می‌کند اما dbo از ابتدای ایجاد پایگاه داده‌ها وجود دارد. مالک پایگاه داده‌ها به این ورود نگاشت پیدا می‌کند. پرس‌وجوی ذیل برای پیداکردن مالک پایگاه داده‌های جاری استفاده می‌شود:

```
select suser_sname(sid) from sys.database_principals where principal_id = user_id('dbo')
```

برای تغییر نگاشت dbo به یک ورود دیگر، باید مالک پایگاه داده‌ها تغییر پیدا کند. برای این منظور می‌توان از روال ذخیره شده‌ی sp_changedbowner یا نحو ALTER AUTHORIZATION استفاده کرد. نام مالک پایگاه داده‌ها در sys.databases نیز ثبت شده است. چنانچه یک پایگاه داده‌ها از یک سیستم به سیستمی دیگر منتقل شود و نام مالک آن در سیستم قبلی در سیستم جدید وجود نداشته باشد، نتیجه‌ی پرس‌وجوی بالا NULL خواهد شد. برای تعیین مالک پایگاه داده‌ها در سیستم جدید می‌توان از فرمان تغییر مالکیت پایگاه داده‌ها استفاده کرد. این حساب یکی از اعضای نقش db_owner می‌باشد و مانند حساب sa در این حساب نیز حذف مجوزها امکان‌پذیر نیست. نکته‌ی قابل توجه این است که sa و اعضای sysadmin بدون توجه به این که مالک پایگاه داده‌ها باشند، همیشه به dbo نگاشت می‌یابند. بنابراین می‌توان گفت dbo یک مفهوم مبهم است که تنها یک اعتبار واحد برای نگاشت به آن مشخص نمی‌شود.

- **Guest**: امکان دسترسی بدون نام به هر پایگاه داده‌ای از طریق این حساب امکان‌پذیر است. این نوع دسترسی معمولا پیشنهاد نمی‌شود و اغلب در تمام پایگاه‌های داده‌ها غیرفعال است. اگر این حساب فعال نباشد، فقط مالک پایگاه داده‌ها و اعضای نقش sysadmin قادر به اتصال به پایگاه داده‌ها می‌باشند. ولی در صورت فعال بودن این نوع حساب، به هر ورودی که به طور صریح به هیچ کاربری نگاشت نیافته است اجازه اتصال به پایگاه داده‌ها به عنوان مهمان داده می‌شود. این نکته قابل توجه است که امکان حذف موثر این نوع حساب از پایگاه داده وجود ندارد. مجوزهای نسبت داده شده به حساب Guest را می‌توان از طریق پرس‌وجوی ذیل بررسی کرد:

```
select permission_name, state_desc, object_name(major_id) as securable,
user_name(grantor_principal_id) as grantor from sys.database_permissions where
grantee_principal_id = user_id('guest')
```

۲-۲- امن‌شدنی‌ها

امن‌شدنی‌ها به دو سطح ذیل تقسیم می‌شوند:

- **سطح ویندوز**

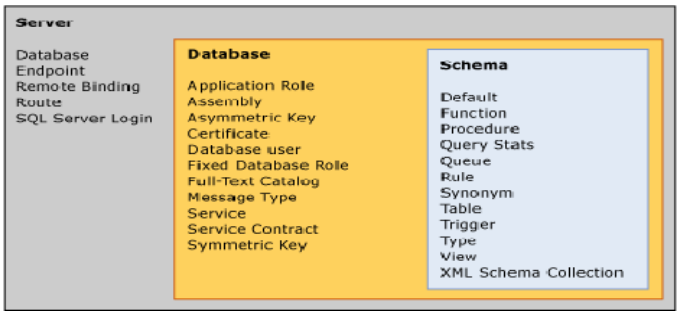
این سطح شامل فایل‌ها و کلیدهای رجیستری که سرویس‌دهنده‌ی SQL از آن‌ها استفاده می‌کند، می‌شود.

- **سطح سرویس‌دهنده‌ی SQL**

همان‌طور که در شکل ۲ نمایش داده شده است، در این سطح امن‌شدنی‌ها به حوزه‌ها[1]ی ذیل تقسیم می‌شوند:

[1] Scopes

- حوزه‌ی سرویس‌دهنده به عنوان بالاترین سطح به اعتبارهای سرویس‌دهنده‌ی SQL مرتبط است. این حوزه شامل امن‌شدنی‌هایی از جمله ورودها، HTTP endpoint ها، گواهی‌نامه‌ها و Event notification ها می‌باشد. این حوزه شامل یک یا چند پایگاه داده‌ها است که سطح بعدی حوزه را مشخص می‌کنند.
- حوزه‌ی پایگاه داده‌ها شامل امن‌شدنی‌هایی از قبیل سرویس‌ها، اسمبلی‌ها و XML schema می‌باشد. یک پایگاه داده‌ها شامل یک یا چند طرح[1] است که هر یک به عنوان *فضای نامی*[2] برای اشیا و پایین‌ترین حوزه‌ی امن‌شدنی‌ها عمل می‌کند.
- حوزه‌ی طرح شامل امن‌شدنی‌هایی از جمله جداول، دیدها[3] و رویه‌ها می‌باشد.



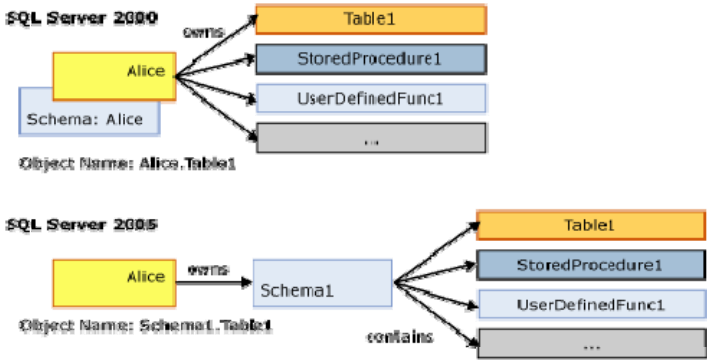
شکل ۲- سلسله مراتب حوزه‌های سطح سرویس‌دهنده

۲-۲-۱- *جدایی طرح و کاربر*[4] در SQL Server 2005

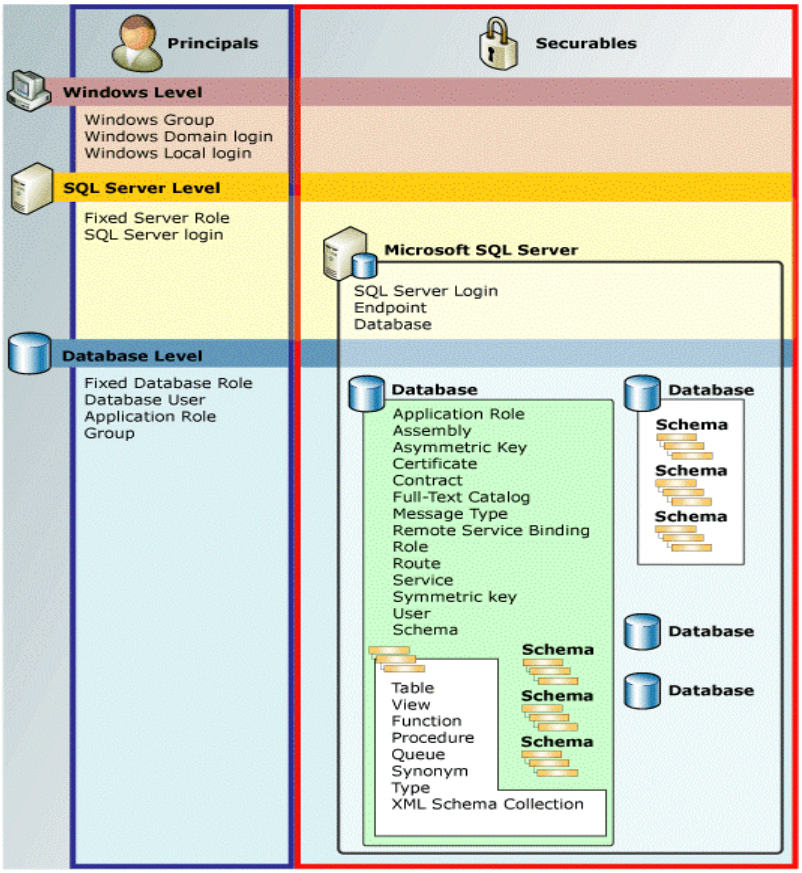
یکی از تغییرات مهم معرفی شده در سرویس‌دهنده‌ی SQL Server 2005 جدایی طرح و کاربر می‌باشد. در نسخه‌های قبلی، کاربر پایگاه داده‌ها به عنوان نام طرح نیز استفاده می‌شد و هر کاربر دارای یک طرح بود. طرح مجموعه‌ای از اشیای پایگاه داده‌ها است که یک فضای نام را شکل می‌دهد.

در سرویس‌دهنده‌ی SQL Server 2000 یک شیء به صورت Server.Database.Owner.Object آدرس‌دهی می‌شد، ولی در سرویس‌دهنده‌ی SQL Server 2005 یک شیء به صورت Server.Database.Schema.Object آدرس‌دهی می‌شود. شکل ۳ این تفاوت را نمایش می‌دهد:

[1] Schema
[2] Name Space
[3] View
[4] User schema separation



شکل ۳- کاربر/طرح/اشیاء در SQL Server 2000, 2005



شکل ۴- اعتبارها/ امن‌شدنی‌ها

مزایای این ویژگی عبارتند از:

۱- چندین کاربر می‌توانند از طریق نقش‌ها یا گروه‌های ویندوزی دارای یک طرح مشترک باشند.

۲- حذف کاربران پایگاه داده‌ها ساده می‌شود.

۳- چندین کاربر می‌توانند دارای یک طرح پیش‌فرض مشترک باشند (وقتی یک طرح پیش‌فرض مشخص نشده است، از طرح پیش‌فرض dbo استفاده می‌شود).

۴- تولیدکنندگان و برنامه‌های کاربردی به جای استفاده از dbo امکان به اشتراک گذاشتن اشیاء را در یک طرح خاص دارند.

۵- مدیریت مجوزها در سطح طرح به جای سطح شیء امکان‌پذیر است.

شکل ۴ یک نمای کلی از اعتبارها و امن‌شدنی‌ها را در SQL Server 2005 نمایش می‌دهد.

۲-۳- مجوزها

مجوزها به اعتبارها اجازه‌ی دسترسی به امن‌شدنی‌ها را می‌دهند. در سطح ویندوز، *فهرست کنترل دسترسی ویندوز*[1] *برای مجوز دادن*[2] و یا *سلب مجوز*[3] استفاده می‌شود. در سرویس‌دهنده‌ی SQL، دستورات GRANT، DENY و REVOKE (GDR) برای کنترل اعمال اعتبارها بر روی امن‌شدنی‌ها استفاده می‌شود. GRANT، عمل خاصی که یک اعتبار (*صاحب امتیاز*[4]) می‌تواند اجرا کند را مشخص می‌کند. DENY برخلاف GRANT، امتیاز را از صاحب امتیاز سلب می‌کند. REVOKE مکانیزمی ساده برای حذف مجوزهای داده شده یا سلب شده است.

فردی که یک عمل GDR را انجام می‌دهد، امتیازدهنده[5] نام دارد. اعمال GDR دارای سه بخش اصلی است. نام مجوز، نام صاحب امتیاز (فاعل عمل GDR) و نام نهادی که مجوز روی آن داده می‌شود (امن‌شدنی- مفعول عمل GDR). امن‌شدنی بخش اختیاری عمل GDR است، زیرا اغلب مجوز به آن دلالت می‌کند. به عنوان مثال آلیس مجوز انتخاب (SELECT) را روی جدول t به باب می‌دهد. این عمل با دستور ذیل انجام می‌شود:

```
grant SELECT on t to Bob
```

در این عمل آلیس امتیاز دهنده، باب صاحب امتیاز، SELECT نام مجوز و t امن‌شدنی است. برای هر امن‌شدنی مجوزهای خاصی وجود دارد. در SQL Server 2005، مجوزهای جدیدی وجود دارد که به امن‌شدنی‌ها و حوزه‌های مختلف اعمال می‌شود. مجوزهای هر حوزه توسط امن‌شدنی‌ها در حوزه‌ی پایین‌تر به ارث برده می‌شوند. برای مثال یک کاربر پایگاه داده‌ها که دارای مجوز SELECT روی یک طرح است، به صورت خودکار این مجوز را روی تمام امن‌شدنی‌هایی که در آن طرح وجود دارد، دارا می‌باشد. با توجه به حوزه‌ی امن-شدنی، مجوز به دو نوع تقسیم می‌شود:

- **مجوز سرویس‌دهنده**

در صورتی که مجوز از این نوع باشد، امن‌شدنی به طور ضمنی سرویس‌دهنده است و در دستور GRD مشخص نمی‌شود. مجوزهای سرویس‌دهنده در کاتالوگ sys.server_permissions ذخیره می‌شود. یک مجوز سرویس‌دهنده برای یک اعتبار سرویس‌دهنده قابل اعمال است.

[1] Windows Access Control List (ACL)
[2] Grant
[3] Deny
[4] Grantee
[5] Grantor

- **مجوز پایگاه داده‌ها**

مجوزهای پایگاه داده‌ها در کاتالوگ sys.database_permissions ذخیره می‌شود. مجوز پایگاه داده‌ها برای اعتبارهای پایگاه داده‌ها قابل اعمال است.

Revoke در جایی ذخیره نمی‌شود زیرا مجوزها را حذف می‌کند و به عبارتی مدخل‌های کاتالوگ را حذف می‌کند. اگر فردی دارای دو نقش باشد که در یکی مجوز انجام عملی را دارد و در دیگری همان مجوز از او سلب شده است، نتیجه نهایی این است که فرد دارای آن مجوز نمی‌باشد. به عبارتی دیگر، همیشه DENY نسبت به GRANT الویت دارد.

۳- نتیجه‌گیری

این سند با هدف آشنایی با امنیت سرویس‌دهنده‌ی SQL Server 2005 به بررسی معماری سرویس‌دهنده می‌پردازد. معماری یکی از جنبه‌های مهم در امنیت سرویس‌دهنده‌ی SQL Server 2005 می‌باشد. سایر جنبه‌های آن در مقالات بعدی مورد بررسی خواهند گرفت.

مراجع

[1] David Litchfield, Chris Anley, John Heasman, Bill Grindlay, *The Database Hacker's Handbook: Defending Database Servers*, July 2005.

[2] Mark Horninger, *How to Cheat at Securing SQL Server 2005*, Sep 2007

[3] http://blogs.msdn.com