فصل دهم

امنیت در بانک اطلاعات

آشنایی

امروزه سازمان‌ها بیش‌ازپیش وابسته به اطلاعاتی هستند که افراد، منابع و امور سازمان را مدیریت می‌کنند. بنابراین تخلف در امنیت اطلاعات ممکن است کل سیستم را به خطر بیاندازد. فقط کافی است تصور کنید که چه اتفاقی ممکن است بیفتد اگر اطلاعات مربوط به بیماران یک بیمارستان به صورت نادرست تغییر داده شود.

تعریف: امنیت (Security) یعنی محافظت از تلاش‌های تبهکارانه برای سرقت (Disclosure)، تغییر (Alternation) و یا تخریب (Destruction) داده‌های بانک اطلاعات.

تهدیدات امنیت به دو صورت عمدی و غیر عمدی هستند. دسته‌ی اول شامل تمام تهدیداتی است که نتیجه‌ی تخلف عمدی کاربران و یا برنامه‌ها است. این کاربران می‌توانند کاربران مجاز سیستم و یا کاربران خارجی (که می‌توانند به صورت غیر مجاز به سیستم و منابع آن دسترسی داشته باشند) باشند. برنامه‌ها نیز می‌توانند هر برنامه محلی و یا هر برنامه از راه دور (شامل ویروس‌ها یا اسب‌های تراوا و...) باشند. دسته‌ی دوم شامل تهدیداتی است که بر اثر ناآگاهی یا عدم دقت و کوتاهی افراد و نارسایی و کیفیت پایین برنامه‌ها رخ می‌دهد.

هنگامی که صحبت از یک بانک اطلاعات امن به میان می‌آید معمولاً سه **هدف** زیر در رابطه با آن مطرح می‌شود:

۱- محرمانگی (Secrecy) : محرمانگی به صورت عدم دسترسی کابران غیر مجاز به اطلاعات تعریف می‌شود. به عنوان مثال یک دانشجو نباید اجازه مشاهده‌ی نمره‌ی سایر دانشجویان را داشته باشد.

۲- جامعیت (Integrity): فقط کاربران مجاز می‌توانند داده‌های مربوطه را تغییر دهند. به عنوان مثال دانشجویان می‌توانند نمرات خود را ببینند اما اجازه‌ی تغییر آن را ندارند. جامعیت به صورت جلوگیری از تغییر، حذف و یا دخالت ناخواسته و نادرست در اطلاعات نیز تعریف می‌شود.

۳- دسترس پذیری (availability)**:** مجوز کاربران مجاز نباید به طور ناخواسته قطع شود. به عنوان مثال استاد درسی که اجازه تغییر نمرات را دارد، همواره باید بتواند این عمل را انجام دهد.

برای رسیدن به سه هدف فوق باید **سیاست‌های امنیتی** واضح و مشخصی تدوین گردد. به عبارت دیگر باید به طور کامل و صریح روشن گردد که چه بخش یا بخش‌هایی از داده‌ها باید محافظت شوند و چه کاربرانی اجازه دسترسی به چه قسمت‌هایی از داده‌ها را دارند و نیز کاربران چه اعمالی مجازند انجام دهند.

این نکته بسیاز مهم است که اغلب کاربران ما فقط به بخش کوچکی از داده‌های بانک اطلاعات دسترسی دارند. بنابراین اینکه اجازه بدهیم همه به کل بانک اطلاعات دسترسی داشته باشند کار بسیار اشتباهی است.

مفاهیم امنیت در بانک اطلاعات

کنترل دسترسی (access control)

کنترل دسترسی باعث می‌شود که وقتی درخواست‌هایی برای دستیابی به بانک اطلاعات مطرح می‌شود، سیستم آنها را ارزیابی کند که قبول و یا رد کند. در این رابطه باید دو مورد را از هم تفکیک کنیم؛ اول سیاست‌ها و دوم مکانیزم‌ها.

سیاست به راهبردهای سطح بالایی گفته می‌شود که چگونگی انجام کارها و قواعد را معین می‌نماید.

مکانیزم به توابع نرم‌افزاری و راهکارهای سخت افزاری سطح پایینی گفته می‌شود که چگونگی پیاده‌سازی سیاست‌ها را بیان می‌کنند.

جداسازی سیاست از مکانیزم چندین فایده دارد:

۱. امکان مقایسه سیاست‌های مختلف و ارزیابی خصوصیات آنها را می‌دهد، بدون آنکه نیازی به چگونگی پیاده‌سازی آنها باشد.
۲. امکان ایجاد مکانیزم‌های جدید که سیاست‌های جدید را اجرا می‌کنند، به وجود می‌آید به طوری که در تغییر یک سیاست احتیاج به تغییر کل پیاده‌سازی نیست.
۳. وجود مکانیزم‌های مختلف که بخش‌هایی از چندین سیاست را در یک زمان اجرا می‌کنند، این امکان را به کاربران می‌دهد که بهترین سیاست را انتخاب کنند.

سیاست **کنترل دسترسی** به روش‌های زیر تقسیم می‌شود:

- کنترل دسترسی محطاتانه (discretionary)
- کنترل دسترسی الزامی (mandatory)

البته کلمه‌ی دیگری نیز برای اینها به کار رفته با نام مدل که عبارت است از مجموعه‌ای از سیاست‌ها که کار دسترسی به یک بانک اطلاعات را مشخص می‌کند.

کنترل دسترسی محطاتانه (discretionary)

این سیاست‌ها، دسترسی کاربران به بانک اطلاعات را بر اساس هویت کاربران (Id) و قوانینی به نام اجازه ورود (مجوز) کنترل می‌کنند. در اینجا برای هر کاربر (یا گروهی از کاربران) نوع دسترسی‌های آنها به هر شیء موجود در بانک اطلاعات مشخص می‌شود. نوع این اشیاء بستگی به بانک اطلاعات دارد. در مدل رابطه‌ای، اشیاء می‌توانند رابطه‌ها (جداول)، دیدها، صفت‌ها و سطرهای جدول باشند. در مدل شیءگرایی می‌توانند شامل کلاس‌ها، اشیاء و متدها باشند.

کنترل‌های قابل اجرا بر روی اشیاء در مدل **رابطه‌ای** معمولاً در دو سطح صورت می‌گیرد:

۱- **سطح account**: در اینجا امتیازهای مختلف توسط مدیر بانک اطلاعات، تشخیص و به کاربران مختلف اعطا می‌شود و در همه رابطه‌ها قابل اعمال می‌باشد.

۲- **سطح جدول**: امتیازهای اعطا شده فقط برای بخش خاصی از جدول می‌باشد.

بر این اساس معمولاً ***انواع مجوزهایی*** که به یک کاربر داده می‌شود به صورت زیر است:

۱- در سطح داده‌ها

در سطح داده‌ها یعنی اینکه کاربر به چه داده‌هایی می‌تواند دسترسی داشته باشد و یا تغییر بدهد.

- **مجوز read**: اجازه خواندن داده‌ها را می‌دهد ولی تغییر داده‌ها را نمی دهد.
- **مجوز insert**: اجازه درج داده جدید را می‌دهد.
- **مجوز update**: اجازه تغییر داده‌ها را می‌دهد.
- **مجوز delete**: اجازه حذف داده‌ها را می‌دهد.

۲- در سطح شِما (schema)

در سطح شما یعنی اینکه آیا کاربر می‌تواند شکل و شمایل جدول را تغییر دهد یا حذف یا اضافه کند.

- **مجوز index**: اجازه ایجاد یا حذف شاخص‌ها را می‌دهد.
- **مجوز resource**: اجازه ایجاد رابطه‌های جدید را می‌دهد.
- **مجوز alteration**: اجازه اضافه یا حذف صفات رابطه را می‌دهد.
- **مجوز drop**: اجازه حذف رابطه‌ها را می‌دهد.

سیاست‌های دسترسی نیز به دو گونه زیر تعریف می‌شوند:

۱. **سیاست‌های بسته**: در این نوع سیاست‌ها دسترسی‌هایی اجازه داده می‌شود که مجوز صریح آنها موجود باشد و تصمیم پیش‌فرض این است که دسترسی رد شود. اکثر سیستم‌ها از سیاست‌های بسته پشتیبانی می‌کنند.

۲. **سیاست‌های باز:** در این سیاست‌ها، مجوز منفی داده می‌شود و در صورت وجود آن، اجازه دسترسی داده نمی‌شود و تصمیم پیش‌فرض (در صورت عدم وجود مجوز منفی) این است که دسترسی قبول شود. سیاست باز فقط در سیستم‌هایی استفاده می‌شود که به حفاظت محدود نیاز دارند و اکثر دسترسی‌ها اجازه داده می‌شود.

تعیین مجوز برای هر کاربر، هر مد دسترسی، و هر شیء، بار اجرایی زیادی را بر سیستم مدیریت بانک اطلاعات می‌گذارد. با **گروه‌بندی** کاربران، مدها و اشیاء این مجوزها برای گروهی از کاربران، دسته‌ای از مدها و یا مجموعه‌ای از اشیاء نگهداری می‌شود.

ممکن است یک گروه از کاربران دارای مجوز مثبت و چند استثناء با مجوز منفی برای چند کاربر باشد. در این حالت برخوردهایی پیش می‌آید. مثلاً فرض کنید یک کاربر به دو گروه تعلق داشته باشد و یکی از گروه‌ها برای یک دسترسی، دارای مجوز مثبت و دیگری برای همین دسترسی دارای مجوز منفی باشد. راه حل‌های مختلفی برای برخورد دسترسی‌ها وجود دارد:

- مجوزهای منفی حفظ می‌شوند (اولویت با ردی‌ها).
- برخورد ممکن است از طریق رابطه‌های ممکن بین گروه‌ها رفع شود. مثلاً اگر یکی از گروه‌ها عضو دیگری باشد ممکن است مجوز تعیین شده برای گروه اول نگهداری شود.
- تعیین اولویت‌های صریح.

نمایش و اعمال مجوز

ما چگونه می‌توانیم در سیستم خود، این روش‌ها و سیاست‌های مختلف را پیاده‌سازی کنیم؟ یعنی چگونه می‌توانیم این مجوزها را نمایش دهیم و آنها را اعمال کنیم؟

یکی از راه‌های نمایش این مجوزها استفاده از ماتریس دسترسی (access matrix) است. در این ماتریس:

- سطرها نشان‌دهنده کاربران و فرایندها (subject) می‌باشند.
- ستون‌ها نشان‌دهنده اشیای موجود (object) می‌باشند.
- درایه‌ها مد دسترسی (mode) کاربر به آن شیء مربوطه می‌باشد.

متأسفانه ماتریس دسترسی ممکن است بسیار بزرگ و پراکنده باشد. همچنین ذخیره مجوزها در این ماتریس ممکن است باعث ناکارایی گردد.

اما راه حل معمولی که در اکثر بانک‌های اطلاعات قرار می‌گیرد، گراف مجوز کاربر است.

گراف مجوز کاربر (authorization graph)

گراف مجوز کاربر گرافی است که در آن:

- مدیر بانک اطلاعات (DBA)، ریشه گراف است.
- هر یک از کاربران گره‌ی دیگری از این گراف هستند.
- اصلاع جهت‌داری از چه ریشه و چه گره‌ی دیگر به گره‌های دیگری که به آن مربوط می‌شود رسم می‌شوند و هر ضلعی از گره‌ی U به W، با برچسب P مشخص می‌شود به این معنی که U امتیاز P را به W واگذار کرده است.

مثال (در SQL):

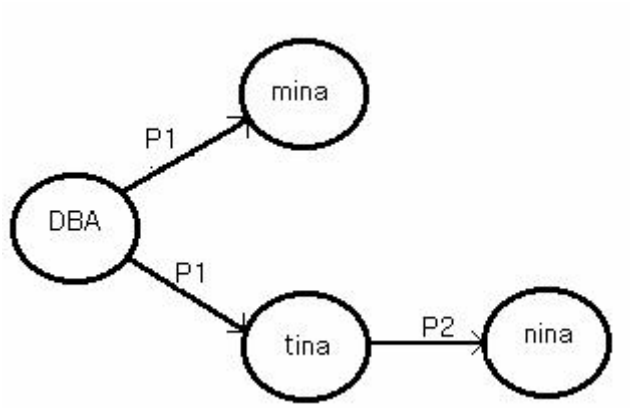
DBA دستور زیر را اجرا می‌کند:

```
GRANT  SELECT ON Student TO tina, mina WITH GRANT OPTION;
```

کاربر tina دستور زیر را اجرا می‌کند:

```
GRANT SELECT ON Student TO nina;
```

گراف مجوز این مثال به صورت زیر خواهد بود:



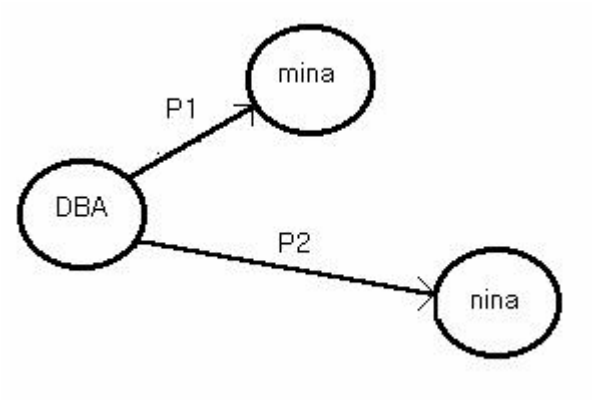
P1: SELECT ON Student WITH GRANT OPTION;
P2: SELECT ON Student

با استفاده از گراف مجوز کاربر، اگر مسیری از DBA به U وجود داشته باشد به طوری که تمام اضلاع این مسیر بر چسب P را داشته باشند، آنگاه U امتیاز P را دارد. درمثال بالاtina, nina, mina می‌توانند روی جدول Student، SELECT کنند اما فقط tina و mina حق واگذاری به غیر را دارند؛ یعنی حق واگذاری به غیر به nina واگذار نشده است.

اگر برای **پس گرفتن** امتیاز، DBA دستور زیر را اجرا کند:

```
REVOKE SELECT ON Student FROM tina;
```

گراف مجوز به صورت زیر در می‌آید:



اگر به دستور فوق قید CASCADE اضافه شود آنگاه فقط mina می‌تواند از جدول Student، انتخاب (SELECT) کند.

مسئله اسب‌های تراوا

سیاست‌های محتاطانه، از کاستی‌هایی برخوردارند که سبب دسترسی های غیرمجاز کاربران به اطلاعات محرمانه سیستم می‌شود. یکی از مهمترین این کاستی‌ها، آسیب پذیری در مقابل اسب‌های تراوا (trojan horses) است. از طریق روش اسب‌های تراوا، یک کاربر غیر مجاز می‌تواند از طریق کاربران مجاز به اطلاعات محرمانه دسترسی داشته باشد بدون اینکه سوء استفاده‌ی او کاملاً واضح و مشخص باشد.

به عنوان مثال فرض کنید دانشجوی x می‌خواهد به لیست نمرات استاد y که در جدول grades ذخیره شده است دسترسی داشته باشد. برای این منظور، دانشجوی x اعمال زیر را انجام می‌دهد:

- جدول جدیدی به نام sample ایجاد کرده و اجازه عمل insert در این جدول را به استاد y می‌دهد (استاد y از این موضوع بی‌اطلاع است).
- سپس برخی توابع DBMS را که استاد y همواره استفاده می‌کند، چنان تغییر می‌دهد که در هنگام کار او با سیستم، اطلاعات از جدول grades خوانده شده و در جدول sample کپی شوند (اسب تراوا).

دانشجوی x سپس منتنظر کار استاد با سیستم شده و پس از آنکه نمرات از جدول grades به جدول sample کپی شدند، توابع تغییر داده شده را به حالت اول بر می‌گرداند تا مدیر بانک اطلاعات، از آن مطلع نشود. در اینجا می‌بینیم که بر خلاف سیاست‌های سیستم که دسترسی به جدول grades را تنها به استاد y داده بود، کاربر دیگری توانست اطلاعات سیستم را کپی نموده و از آنها مطلع شود. بنابراین نیاز به سیاست‌های دسترسی امن‌تری داریم که امنیت کامل دسترسی‌ها را برقرار کرده و در مقابل حملات امنیتی (از جمله روش اسب‌های تراوا) در امان باشد.

روش‌هایی که تا به حال گفتیم و روش‌های معمول امنیت در بانک اطلاعات هستند، نقاط ضعفی دارند. برای رفع این نقاط ضعف، روش‌های جدیدی مطرح شده است که به اختصار به آنها می‌پردازیم.

کنترل دسترسی الزامی (mandatory access control)

معروف‌ترین مدل کنترل دسترسی الزامی، مدل Bell-LaPadula است. در این مدل پدیده‌های سیستم به چهار دسته‌ی زیر می‌شوند:

۱- شیء (object)

۲- فرایند (subject)

۳- کلاس‌های امنیتی (security classes)

۴- **حساسیت** (clearance)

سطوح امنیتی، به صورت زیر تعریف می‌شوند: خیلی محرمانه (TS)، محرمانه (S)، سری (C)، طبقه‌بندی نشده (U) و ترتیب آنها به صورت TS > S > C > U است.

مدل Bell-LaPadula دو **محدودیت** زیر را در تمام دسترسی‌های به اشیاء بانک اطلاعات قائل است:

۱. فرایند S اجازه دسترسی خواندن به شیء O را دارد اگر

Class(S) >= Class(O)

به عنوان مثال کاربر X با حساسیت TS، می‌تواند جدولی با حساسیت C را بخواند. اما کاربر Y با حساسیت C، اجازه خواندن از جدولی با حساسیت TS را ندارد.

۲. فرایند S اجازه نوشتن بر روی شیء O را دارد اگر

Class(O) >= Class(S)

به عنوان مثال کاربر X با حساسیت S، اجازه نوشتن در جدولی با حساسیت S یا TS را دارد.

دانشجویان عزیز کتاب را به دقت مطالعه فرمایند.

رمزنگاری

یکی دیگر از روش‌های برقراری امنیت در سیستم‌های بانک اطلاعات، رمزنگاری است. این روش امنیت کانال‌های ارتباطی را برقرار می‌کند. ایده اصلی رمز نگاری داده‌ها استفاده از الگوریتم‌های رمزنگاری و نیز یک کلید رمزنگاری مخصوص مدیر بانک اطلاعات است که به صورت امن نگاه داشته می‌شود.

روش‌های رمز نگاری به دو دسته کلی زیر تقسیم می‌شوند:

- **متقارن:** در روش رمزنگاری متقارن، فرستنده و گیرنده از یک کلید سری مشترک برای رمزنگاری و رمزگشایی استفاده می‌کنند. روش Data Encryption Standard (DES) مثالی از رمز نگاری متقارن است. بدیهی است که برای دو طرف ناشناس توافق بر روی یک کلید سری مشترک دشوار و ناامن است. بنابراین این روش مورد استفاده کمتری دارد.
- **نامتقارن:** روش دیگر، روش رمزنگاری نامتقارن است. در این روش هر فرد دو کلید در اختیار دارد: کلید عمومی که آزادانه منتشر می‌شود و کلید خصوصی که به صورت خصوصی و کاملاً محرمانه نگهداری می‌شود. در این روش، فرستنده، داده‌ها را با کلید عمومی رمز کرده و به گیرنده ارسال می‌کند. داده‌های رمز شده تنها با کلید خصوصی گیرنده قابل رمز گشایی هستند و از آنجایی که کلید خصوصی هر شخص منحصر به فرد است و به طور کاملاً امن نگاه داشته می‌شود، شخص دیگری نمی‌تواند این اطلاعات را رمزگشایی کرده و یا از آنها استفاده نماید. این روش از امنیت بالایی برخوردار است و مورد استفاده فراوانی دارد.

بحث‌های تکمیلی امنیت

نقش مدیر بانک اطلاعات (DBA)

مدیر بانک اطلاعات (DBA) نقش بسیار مهمی در تعیین و تبیین سیاست‌های امنیتی سیستم دارد. معمولاً مدیر بانک اطلاعات دارای حساب ویژه‌ای در DBMS است که با این حساب می‌تواند سیستم را کنترل کرده و امنیت آن را برقرار نماید.

به طور کلی وظایف مدیر بانک اطلاعات در قبال امنیت سیستم، به شرح زیر است:

۱- ایجاد حساب (account) برای کاربران: همه کاربران و گروه‌ها برای استفاده از بانک اطلاعات نیازمند حساب کاربری و کلمه عبور می‌باشند. بدیهی است برنامه‌هایی که با بانک اطلاعات در تعامل هستند نیز باید دارای حساب ویژه‌ای در سیستم DBMS باشند.

۲- کنترل سیاست‌های الزامی(mandatory): در صورتی که سیستم مدیریت بانک اطلاعات، از روش کنترل الزامی برخوردار باشد مدیر بانک اطلاعات باید کلاس‌های امنیتی مربوط به اشیاء بانک اطلاعات، کاربران، گروه‌ها و همچنین روابط بین آنها را مشخص کند.

۳- پیگیری اجازه‌ها و مدیریت کارنامه (log): مسئولیت دیگر مدیربانک اطلاعات، مدیریت و کنترل کارنامه سیستم است. اینکه هر کاربر چه دستوراتی را اجرا کرده و یا به چه داده‌هایی دسترسی دارد و یا چه دستوراتی به سیستم ارسال کرده است، می‌تواند در بررسی مشکلات سیستم و یافتن مشکلات امنیتی آن، مؤثر و مفید باشد.

امنیت در بانک‌های اطلاعات آماری (statistical databases)

بانک اطلاعات آماری، بانکی است که **فقط** به پرسش و پاسخ‌های آماری پاسخ می‌دهد. به عنوان مثال اگر یک بانک اطلاعات آماری از اطلاعات دانشجویان نگهداری کنیم، تنها پرسش و پاسخ‌های آماری از قبیل میانگین نمرات، بالاترین سن، کمترین تعداد واحدها و... برای این بانک مجاز خواهد بود و پاسخگویی به پرسش‌هایی در مورد تک تک دانشجوها مجاز نخواهد بود. امنیت در چنین بانک اطلاعاتی نیازمند در نظر گرفتن مسایل جدیدی است.

مشکل اساسی بانک‌های اطلاعات آماری عمدتاً به این صورت است که کاربر با ارسال چند پرسش آماری بتواند اطلاعات محرمانه‌ای در مورد تک تک اعضا و داده‌های آن به دست آورد. مثلاً با سناریوی زیر، یک کاربر می‌تواند حقوق مسن‌ترین و جوان‌ترین استاد را تشخیص دهد:

- پرسش اول: چند استاد وجود دارد که سن آنها از X بزرگتر باشد؟ کاربر، مقدار X را مرتباً عوض می‌کند تا پاسخ دریافتی از سیستم عدد یک باشد.
- پرسش دوم: بیشترین حقوق افرادی که سنی بالاتر از X دارند چند است؟ جواب این پرسش حقوق پیرترین فرد دانشگاه را مشخص می‌کند. به همین ترتیب می‌توان حقوق جوان‌ترین استاد دانشگاه را پیدا کرد.

عمده‌ترین روش‌های حل مشکل فوق این است که:

۱- تنها به پرسش هایی پاسخ داده شود که برای پاسخگویی آنها نیاز به حداقل N سطر از جداول باشد. با در نظر گرفتن عدد مناسبی برای N (بسته به سیستم) می‌توان جلوی مشکلات فوق را تا حدی گرفت. مثلاً با در نظر گرفتن $N > 1$ در مثال قبل می‌توان جلوی پاسخ‌دهی به سؤال دوم مبنی بر اینکه "بیشترین حقوق افراد با سن بالاتر از X چند است؟ " را گرفت.

۲- محدودیتی برای اشتراک ردیف‌هایی که در پرسش‌های متوالی که توسط یک شخص داده می‌شود در نظر بگیریم.