D., Pazzani M. Learning probabilistic user models. In workshop notes of Machine Learning for User Modeling, Sixth International Conference on User Modeling, Chia Laguna, Sardinia, 2-5 June 1997, 6p 5. Konopnicki D., Shmueli O. W3QS: A Query System for the World-Wide Web. VLDB'95, Proceedings of 21th International Conference on VeryLarge Data Bases, September 11-15, 1995, Zurich, Switzerland.54-65p http://www.informatik.uni-rier.de/~ley/db/conf/vldb/KonopnickiS95.html 6. Эйнджел Дж. Proxy-серверы. //LAN/Журнал сетевых решений, №6, 1999. http://www.osp.ru/lan/1999/06/016.htm

УДК 681.3

Д.О. Тарасов, А.М. Пелещишин, П.І. Жежнич
НУ "Львівська політехніка",
кафедра "Інформаційні системи та мережі"

ОБМЕЖЕНИЙ НАБІР ОПЕРАЦІЙ ДЛЯ РОБОТИ З БАЗАМИ ДАНИХ

This paper describes some approaches to defense from unauthorized or mistaken changes of data with INSERT…, DELETE… and UPDATE… operations. We consider data operations semantics according to requirements of information system designer and security disturber. Several variants of these operations using in practice, which can result in data authenticity violation, are adduced. Therefore, we propose to limit data manipulating operations using that can ensure stable functionality of information system without data authenticity violations.

У статті подано деякі підходи для захисту від неавторизованих або помилкових змін даних операціями INSERT…, DELETE… та UPDATE… . Розглянуто семантику операцій з данимі з точки зору розробника інформаційної системи та порушника безпеки. Запропоновано використовувати, наведений у роботі, обмежений набір операцій з даними, що дозволяє зменшити ризики порушення справжності даних у інформаційних системах.

Операції з даними у реляційних базах даних

Традиційно у базах даних (БД) розрізняють два види операцій над даними – операції вибірки та операції зміни (модифікації) даних. Ці операції реалізуються командами мови маніпулювання даними DML (Data Manipulating Language). Права на виконання саме цих операцій має переважна більшість операторів ЕОМ, які працюють з БД.

Операції вибірки – це проекція, вибірка кортежів, з'єднання та об'єднання відношень тощо. У мові SQL вони реалізуються за допомогою конструкції SELECT… .

Несанкціоноване використання операцій вибірки призводить до порушення конфіденційності даних. Безпосереднього впливу на роботу інформаційної системи (ІС) несанкціоноване використання операцій вибірки не має. Єдина проблема для роботи ІС –

несанкціоноване використання обчислювальних ресурсів та зменшення продуктивності роботи ІС.

Операції зміни даних – це додавання кортежів у відношення, вилучення кортежів з відношення, заміна значень атрибутів кортежів. У мові SQL вони реалізуються за допомогою конструкцій INSERT…, DELETE…, UPDATE… .

Несанкціоноване використання операцій зміни даних призводить до порушення цілісності даних, достовірності інформації в ІС, працездатності ІС, конфіденційності даних.

Наслідками несанкціонованого, або помилкового, застосування операцій зміни даних може бути часткове або повне припинення роботи ІС з можливим призупиненням функцій підприємства, які базуються на роботі ІС.

Далі розглядаються шляхи захисту саме від несанкціонованих або помилкових змін даних за допомогою операцій INSERT…, DELETE…, UPDATE… .

Семантика операцій з даними з точки зору проектувальників ІС

Під час проектування схеми БД, проектувальник ІС враховує можливості DML та правила роботи з відношеннями (наприклад, тільки вставка даних, можливість заміни значень, пакетна обробка кортежів тощо).

Таблиця 1

Семантика операцій з даними з точки зору проектувальників ІС

Операція	**Призначення**
INSERT	Внесення у ІС нових даних.
DELETE	Виправлення помилок.
	“Чистка БД” – вилучення непотрібної фактографічної, історичної, нормативно-довідкової та іншої інформації з метою збільшення об’ємів вільної пам’яті та зменшення навантаження на обчислювальні ресурси.
UPDATE	Оновлення фактографічної інформації, “довизначення” значень атрибутів кортежів.
	Виправлення помилок.

Як бачимо, призначенням операцій зміни даних є: внесення нової інформації, виправлення помилок, оновлення фактографічної інформації, “чистка БД”.

Семантика операцій з даними з точки зору порушника безпеки ІС

Потенційний порушник безпеки ІС часто отримує доступ до БД під виглядом легального користувача (наприклад, за допомогою викраденого пароля). Іншими шляхами може бути використання слабких місць політики безпеки, засобів аутентифікації, отримання від адміністратора прав на “додаткові” об’єкти БД. Часто свідомими або несвідомими порушниками безпеки є співробітники підприємства і легальні користувачі ІС.

В усіх цих випадках порушник має можливість виконувати передбачені проектувальниками ІС та діючою політикою безпеки операції над доступними для цього об’єктами, зокрема, створювати та видаляти записи, виправляти помилки.

Але порушник використовує наявні можливості для іншого призначення.

Таблиця 2

Семантика операцій з даними з точки зору порушника безпеки ІС

Операція	**Варіант використання**
INSERT	Внесення у ІС нових даних (наприклад підробка інформації).
	Розширення та закріплення сфери власних повноважень.
	Занесення сміття з метою блокування ресурсів (сторінки файлів БД, обчислювальні ресурси, вільне місце пристроїв збереження даних тощо).
DELETE	Знищення даних ІС.
	Знищення слідів попередніх втручань у роботу ІС.
	Знищення фактографічної інформації з метою підробки інформації для процесів прийняття рішень.
	Блокування ресурсів (окремі записи, сторінки файлів БД, таблиці, обчислювальні ресурси тощо).
UPDATE	Внесення у ІС помилкових даних з метою підробки інформації для процесів прийняття рішень.
	Знищення слідів попередніх втручань у роботу ІС.
	Розширення та закріплення сфери власних повноважень (не все можна зробити за допомогою INSERT, заважають обмеження унікальності значень полів).
	Знищення даних ІС.
	Блокування ресурсів (окремі записи, сторінки файлів БД, таблиці, обчислювальні ресурси тощо).
Комбінація: DELETE + INSERT	Замість команди UPDATE

Легальні користувачі ІС часто стають несвідомими порушниками безпеки внаслідок помилкового використання операцій зміни даних.

Таблиця 3

Варіанти помилкового використання операцій з даними

Операція	**Розповсюджені варіанти помилкового використання**
INSERT	Внесення у ІС помилкових даних.
DELETE	Явне знищення даних ІС за помилковим критерієм.
	Неявне знищення даних ІС (наприклад, при використанні механізму каскадного видалення).
	Блокування ресурсів (окремі записи, сторінки файлів БД, таблиці, обчислювальні ресурси тощо).
UPDATE	Внесення у ІС помилкових даних з можливістю відновлення (за допомогою іншої команди UPDATE, або відкочення транзакції).
	Внесення у ІС помилкових даних з можливістю відновлення з резервних копій, архівних документів тощо.
	Блокування ресурсів (окремі записи, сторінки файлів БД, таблиці, обчислювальні ресурси тощо).

Постає завдання зменшення втрат при використанні порушником безпеки операцій зміни даних, з повним збереженням визначеного вище призначення операцій зміни даних.

У випадку можливості користувачів виправляти помилкові дані у ІС заміною окремих даних (UPDATE) або вилученням помилкової інформації, порушується

відповідність між інформацією в БД системи та дійсною інформацією. А це, у свою чергу, порушує вимоги до захисту інформації [1].

Приклад 1. В момент часу t_1, користувач u_1 ввів у ІС дані d_1. У момент часу $t_2>t_1$, користувач u_2, згідно з бізнес-правилом

$$\begin{cases} d_1 \Rightarrow a_1, \\ \overline{(d_1)} \Rightarrow \overline{(a_1)}. \end{cases} \quad (1)$$

прийняв на основі d_1 рішення a_1. В момент часу $t_3>t_2$, користувач u_1 помітив помилку у введених даних та виправив помилку (наприклад, шляхом DELETE+INSERT) замінивши d_1 на d_2, де $d_2 = \overline{(d_1)}$.

Бізнес-правила регламентують прийняття на основі d_2 рішення a_2

$$a_2 = \overline{(a_1)}. \quad (2)$$

Отже, у випадку перевірки (скарга, рекламація тощо) обґрунтованості рішення a_1, у момент часу $t_4>t_3$, на підставі даних наявних у ІС, отримуємо

$$\overline{(d_1)} \Rightarrow a_1 \quad (3)$$

– користувач u_2 прийняв неправильне (з точки зору наведених бізнес-правил) рішення.

Наведена схема дій користувача u_1 описує виправлення помилки (або дії порушника безпеки у момент часу t_1 і приховування своїх дій у момент часу t_3).

Як бачимо, використання ІС для перевірки обґрунтованості прийняття рішень неможливе. Необхідно використовувати інші джерела інформації – оригінали документів (не завжди містять d_1 або d_2 у явній формі, необхідні різного роду розрахунки), резервні копії та журнали роботи ІС (поряд з технічною складністю використання та аналізу даних з резервних копій, квант часу копіювання не завжди містить зміну d_1 або d_2). Таким чином неможлива повноцінна автоматизація документообігу, управління інформацією та персоналом, зростають затрати на захист ІС.

Шляхи забезпечення функціональності ІС з обмеженим набором операцій з БД

Для уникнення проблем, аналогічних до розглянутих у прикладі 1, для об'єктів захисту О потрібно створити своєрідний "накопичувач" інформації

$$\text{Hist} = \langle A, Tr, hist \rangle, \quad (4)$$

де субнакопичувач A – інформація для аудита, Tr – інформація для короткотермінового збереження цілісності (до підтвердження виконання команди користувачем включно), $hist$ інформація для забезпечення можливості повернення об'єкту у цілісний стан на протязі тривалого терміну.

A – зберігає інформацію, необхідну для аудиту використання об'єкта. Реалізація субнакопичувача A стандартними системними засобами промислових СКБД не задовольняє багато вимог до повноти інформації та ефективності використання інформації [1, 2].

Tr – забезпечує інформацією механізм транзакцій та зберігається до остаточного підтвердження користувачем необхідності виконання операції. Підтвердженням є команда завершення транзакції (у системах з підтримкою транзакцій), багатократний безпомилковий ввід інформації про об'єкт, або інші події згідно з правилами роботи ІС. У більшості промислових СКБД, які працюють в архітектурі клієнт/сервер, субнакопичувач Tr автоматично реалізується за допомогою транзакцій.

$hist$ – інформація для забезпечення можливості повернення об'єкта у цілісний стан на заданий момент часу упродовж тривалого терміну. Стандартними системними засобами промислових СКБД цей субнакопичувач не реалізований.

Типовими об'єктами захисту у реляційних БД є кортежі відношень БД - $O = \langle a_1, ..., a_n \rangle \in R$. Постає задача виправлення помилок без знищення існуючої у кортежах інформації, іншими словами – забезпечення збереження історичної інформації у БД.

Першим з методів збереження історичної інформації є використання часових БД (temporal databases), реалізація версійності [5]. Часові БД природно дозволяють реалізувати операцію оновлення даних зі збереженням попереднього значення за допомогою фіксації "станів" значень атрибутів кортежів.

Другим методом збереження історичної інформації є архівування інформації у альтернативні до О структури даних з подальшою заміною старих значень О на нові. Недоліками методу архівування є:

- практично двократне дублювання структур даних;
- розділення актуальної та історичної інформації ускладнює процедури аналізу;
- необхідність у використанні програмних кодів.

Третім методом збереження історичної інформації є проектування схеми БД для подальшого забезпечення накопичення історичної інформації спільно з "актуальним" останнім значенням О. Як накопичувачі можна використовувати структури вигляду <об'єкт, історія об'єкта> запропоновані у [3, 4]. Отже, значення кортежу О зберігаються у нових структурах даних $R_1, ...R_m$, серед яких виділяються три множини: S^c, S^h, S^a. У відношеннях $R_i \in S^c, 1 \le i \le k \le m$ зберігається незмінна інформація О, у $R_j \in S^h, k+1 \le j \le m$ частина О – атрибути, які змінюються (довизначаються). Дані аудиту виділяються у окремі відношення $R_l \in S^a, 1 \le l \le v$ або зберігаються як частина $R_j \in S^h$. Доцільно проектувати схему S^a з врахуванням функціональних залежностей між даними аудиту О та стандартними атрибутами журналів аудиту СКБД.

Проектування схеми БД з врахуванням потреб захисту інформації та забезпечення збереження історичних даних створює фундамент СЗІ з примусовим управлінням доступом на основі стандартних СКБД.

Таким чином, забезпечення збереження історичних даних у ІС, окрім можливості розширеного аналізу даних у контексті часу, побудови DataWarehouse та використання інших перспективних технологій дає змогу:

- усунути необхідність вилучення даних (DELETE);
- виправляти помилкові дані з використанням лише операції вставки (INSERT);

- не використовувати операції оновлення (як для виправлення помилок, так і для довизначення значень окремих атрибутів існуючих кортежів);
- володіти інформацією про стан ІС на заданий момент часу;
- інтегрувати історичну інформацію з даними аудиту ІС;
- проводити аналіз ефективності роботи персоналу, швидкості документообігу тощо.
- Обмежений набір операцій

Для зменшення втрат при використанні порушником безпеки операцій зміни даних необхідно обмежити застосування операцій зміни даних забороною окремих операцій та частковим обмеженням застосування інших. Зрозуміло, що результуючий набір операцій має бути повним у тому сенсі, що за допомогою операцій з отриманого набору, користувачі можуть виконувати свої функціональні обов'язки.

Тому пропонується:

1) Заборонити операцію заміни значень атрибутів.
2) Обмежити право вилучення кортежів для унеможливлення:
    - заміни UPDATE комбінацією DELETE + INSERT;
    - знищення історичної інформації та даних аудиту;
    - каскадного вилучення;
    - знищення слідів несанкціонованих втручань.

Отже, обмежений набір операцій для користувачів БД містить операції INSERT, SELECT, DELETE (з деякими обмеженнями використання DELETE).

Для зручності, вказані операції можна об'єднувати у процедури та виконувати процедури за допомогою команди EXECUTE.

Комбінація DELETE + INSERT дає змогу здійснити операцію оновлення. Разом з тим, операція DELETE дозволяє порушнику знищити сліди втручання у роботу ІС. Окрім цього, знищення даних порушує відповідність між БД системи та дійсними даними, що унеможливлює правильний аналіз підстав прийняття рішень. Тому пропонується обмежити використання операції вилучення даних (можна навіть повністю усунути операцію вилучення). Зокрема забороняється вилучати кортежі, які містять історичну інформацію об'єктів захисту та дані аудиту; кортежі, на які є посилання зовнішніх ключів.

Особливо небажаною є можливість каскадного вилучення даних. У випаду знищення об'єкта О каскадне вилучення знищує (часто без додаткових попереджень) кортежі кількох відношень, які посилаються на О.

Безумовно, адміністратору БД дозволяється використання всіх операцій, у тому числі операцій UPDATE та DELETE. Використання операцій UPDATE та DELETE часом суттєво спрощує розв'язання задач адміністрування БД.

Вилучення даних для "чистки БД" є задачею адміністрування БД. Відповідно, права на вилучення даних для "чистки" потрібні лише адміністраторам БД. Враховуючи постійне зменшення цін на обчислювальні ресурси та вартість пристроїв збереження інформації, потрібність "чистки БД" постійно зменшується.

Для збереження функціональності ІС без використання операцій UPDATE та DELETE необхідно забезпечити збереження історичної інформації в ІС, для чого слід належно класифікувати дані та проектувати схему БД з врахуванням вимог до захисту інформації та існуючої системи аудиту СКБД.

1. Тарасов Д.О., Основні задачі захисту баз даних // Вісник НУ "Львівська політехніка".— 2000 р., № 406.— с. 216-221. 2. Тарасов Д.О. Аудит баз даних. - Защита информации: Сборник научных трудов.— К.: КМУГА, 2000.— С.136-140. 3. Жежнич П.І., Кравець Р.Б., Пасічник В.В., Пелещишин А.М. Семантично відкриті інформаційні системи // Вісник НУ "Львівська політехніка", 1999р., № 383.— С. 73-84. 4. Жежнич П.І., Кравець Р.Б., Пасічник В.В., Пелещишин А.М. Основні правила побудови семантично відкритих інформаційних систем// Вісник НУ "Львівська політехніка", 1999р., № 383.— С. 84-95 5. Ahmed R., Navathe S.B. Version management of composite objects in CAD databases", Proc. ACM SIGMOD Conf. On the Management of Data, 1991, Pp. 218—227.

УДК 681.3

Н. Б. Шаховська
НУ "Львівська політехніка",
кафедра "Інформаційні системи та мережі"

ЗАСТОСУВАННЯ АПАРАТУ БАГАТОЗНАЧНОЇ ЛОГІКИ У СИСТЕМАХ БАЗ ДАНИХ

Indeterminate and fuzzy data stored in relational database are described in this paper.

Розглянуто способи подання нечіткої та неповної інформації у базах даних за допомогою багатозначної логіки.

Для опису даних та роботи з ними в умовах невизначеності та слабкої структурованості використовують:

- теорію ймовірностей,
- нечітку логіку (нечіткі множини, лінгвістичні змінні, інтервальні оцінки, нечіткі числа, емпіричні оцінки) [18],
- k-значну логіку [18, 29].

Неповнота інформації може зустрічатися на різних рівнях [28]:

- невідомо, чи властивість притаманна ПО – невизначеність на рівні відношення;
- відомо, що властивість притаманна ПО, але невідомо, чи вона притаманна даному об'єктові – невизначеність на рівні кортежів;
- відомо, що властивість притаманна ПО і даному об'єктові, але невідомо, як вона на об'єкті проявляється – невизначеність на рівні значень атрибутів.

Найбільші можливості для подання неповноти даних мають неоднорідні бази даних, коли невизначеність вводиться на рівні значень у відношеннях. Такий підхід дає можливість відображати у базах такі випадки неповноти даних:

- значення знаходяться в інтервалі або є одним із дискретної множини значень, зокрема сюди належить невідоме значення;
- значення не існує;
- є неповна чи часткова інформація про значення, яка подається за допомогою нечіткого поняття.

Класифікацію підходів подання неповноти можна зобразити так: