УДК 681.325

О. В. Нечипоренко, *к.т.н., доцент,.*
С. А. Міценко, *аспірант*
Черкаський державний технологічний університет
б-р Шевченка, 460, м. Черкаси, 18006, Україна
kafedra_ckc@mail.ru

МЕХАНІЗМИ ЗАБЕЗПЕЧЕННЯ ЗАХИСТУ БАЗ ДАНИХ В СУЧАСНИХ СУБД

В статті наведено результати аналізу основних сервісів безпеки і якісного оцінювання механізмів захисту та моделей безпеки для визначення ефективних технологій захисту баз даних в сучасних СУБД.

Ключові слова: *захист даних, управління доступом, модель безпеки, аутентифікація, шифрування, протоколювання і аудит, оператори SQL.*

Вступ. Основною формою організації інформаційних масивів в інформаційних системах є бази даних (БД). На відміну від файлової системи організації і використовування інформації, БД існує незалежно від конкретної програми і призначена для сумісного використовування багатьма користувачами. Сьогодні найбільш поширеною моделлю даних БД є реляційна модель [1].

Системи управління базами даних (СУБД), особливо реляційні, стали домінуючим інструментом зберігання великих масивів інформації. В зв'язку з цим забезпечення інформаційної безпеки СУБД має вирішальне значення для безпеки організації в цілому. Основні вимоги по безпеці даних, що висуваються до БД і СУБД, багато в чому збігаються з вимогами до безпеки даних в комп'ютерних системах – контроль доступу, криптозахист, перевірка цілісності, протоколювання і т. д. Щоб забезпечити ефективний захист даних в БД, необхідно визначити і проаналізувати набір сервісів та їх механізмів, що забезпечують захист.

Метою роботи є проведення аналізу основних сервісів безпеки, якісної оцінки характеристик механізмів і методів захисту БД в сучасних СУБД для визначення ефективних технологій захисту БД.

Результати досліджень. Аналіз засобів захисту СУБД показує, що система захисту спрямована на реалізацію таких сервісів безпеки: управління доступом; ідентифікація і аутентифікація; криптографія; захист сервісів, протоколювання і аудит; засоби мови SQL. Розглянемо їх детальніше.

Управління доступом в БД включає такі питання, як доступ до таблиць і її полів. Для організації цього доступу використовуються моделі безпеки, які включають дискреційну, мандатну і рольову моделі [2].

Способом формалізованого представлення дискреційного доступу є матриця доступу або списки управління доступом, що встановлюють перелік користувачів і перелік дозволених операцій відносно кожного об'єкта БД. Можливі декілька підходів до побудови дискреційного управління доступом: децентралізована, централізована та змішана моделі безпеки. Саме змішаний варіант реалізований у більшості СУБД.

Мандатна модель поєднує захист і обмеження прав, що використовуються відносно комп'ютерних процесів, даних і системних пристроїв, та призначена для запобігання їх небажаному використовуванню. Для СУБД мандатна модель безпеки може розширювати або замінювати дискреційний контроль доступу і концепцію користувачів і груп.

Права доступу кожного суб'єкта і характеристики конфіденційності кожного об'єкта відображаються у вигляді сукупності рівня конфіденційності і набору категорій конфіденційності. Для реалізації безпеки за допомогою мандатної моделі рядкам і стовпцям таблиці БД приписуються мітки, які потім надаються користувачам. Ефективно застосовуватися мандатна модель може тільки разом з дискреційною.

Рольова модель – розвиток політики виборного управління доступом, при цьому права доступу суб'єктів системи на об'єкти групуються з урахуванням специфіки їх використання, утворюючи ролі. Управління правами доступу здійснюється як на основі матриці доступу, так і на основі правил, що регламен-

тують поведінку (ролі) користувача та їх активацію під час сеансів.

Рольове розмежування доступу дозволяє реалізувати гнучкі, динамічні правила розмежування доступу. Безпека в цій моделі забезпечується чіткими визначеннями ролей адміністратора БД і користувача БД на права доступу до об'єктів БД і прав на читання, модифікацію, запис і видалення об'єктів. Технологія управління доступом на основі ролей є достатньо гнучкрю і потужною, щоб змоделювати як виборне, так і мандатне управління доступом.

Процедури ідентифікації, аутентифікації і авторизації є обов'язковими для будь-якої захищеної БД. Сутність процедури ідентифікації полягає в призначенні користувачу, тобто об'єкту – споживачу ресурсів сервера БД – імені. Ім'я користувача – це деяка унікальна мітка, що відповідає прийнятим угодам і забезпечує однозначну ідентифікацію об'єкта реального світу в просторі об'єктів, що відображаються. Сутність процедури аутентифікації полягає в підтвердженні автентичності користувача, що представив ідентифікатор. Сутність процедури авторизації полягає у визначенні переліку конкретних інформаційних ресурсів, з якими аутентифікованому користувачу дозволена робота. В процесі авторизації встановлюється і набір можливих операцій з даними, які може здійснювати користувач.

З погляду БД процедури аутентифікації можуть бути як внутрішніми (засобами самої БД), так і зовнішніми (засобами операційної системи (ОС) або мережі). У ряді сучасних СУБД використовується зовнішня аутентифікація, що ґрунтується на біометричній аутентифікації, аутентифікації із застосуванням токенів (ґрунтується на володінні) або парольній аутентифікації (ґрунтується на знанні).

Біометрична аутентифікація – процес доведення і перевірки автентичності заявленого користувачем імені через пред'явлення користувачем свого біометричного образу. Біометричними характеристиками людини є відбитки пальців і долоні, звуки її голосу, обличчя, відбиток сітківки ока, особливості рухів і хода, особливості роботи на клавіатурі, підпис. У статті [3] наведені результати системного аналізу із визначенням якісної оцінки сучасних біометричних датчиків відбитків пальця.

Серед процедур аутентифікації виділяється аутентифікація, що ґрунтується на знанні (на паролях). Парольні фрази забезпечують більшу безпеку, ніж короткі паролі, але вимагають більшого часу для введення.

Заходи, що дають змогу значно підвищити надійність парольного захисту – накладення технічних обмежень; управління терміном дії паролів, їх періодична зміна; обмеження доступу до файлу паролів; обмеження кількості невдалих спроб входу в систему; навчання і виховання користувачів; використовування програмних генераторів паролів [4]. Перелічені заходи доцільно застосовувати завжди, навіть якщо разом з паролями використовуються інші методи аутентифікації, наприклад, які ґрунтуються на володінні токенами (їх використанні).

Токен – це предмет або пристрій, володіння яким підтверджує автентичність користувача. Розрізняють токени з пам'яттю (пасивні, які тільки зберігають, але не обробляють інформацію) та інтелектуальні токени (активні).

Найпоширенішим різновидом токенів з пам'яттю є картки з магнітною смугою. Для використовування подібних токенів необхідний пристрій читання, забезпечений також клавіатурою і процесором. Інтелектуальні токени характеризуються наявністю власної обчислювальної потужності. Вони підрозділяються на інтелектуальні карти, стандартизовані ISO й інші. Карти потребують інтерфейсного пристрою, інші токени зазвичай володіють ручним інтерфейсом.

Ефективність процедур ідентифікації та аутентифікації істотним чином впливає на ефективність системи безпеки в цілому. Нині вимоги до безпеки з боку споживачів достатньо високі.

Одна з основних загроз для БД – це несанкціоноване копіювання даних або фізична крадіжка носія інформації. Найефективнішим методом боротьби з такими загрозами є шифрування даних, тобто зберігання і передача важливих даних у зашифрованому вигляді.

За способом функціонування системи шифрування СУБД ділять на два класи [5]:

- системи прозорого шифрування (включаються адміністратором);
- системи непрозорого шифрування (викликаються користувачем).

У системах прозорого шифрування криптографічні перетворення здійснюються непомітно для користувача, а його програми не змінюються. Системи прозорого шифрування можуть бути як вбудованими в СУБД, так і зовнішніми відносно СУБД.

При прозорому шифруванні використовується ключ шифрування БД, який зберігається в завантажувальному записі БД для доступності при відновленні БД. Функція прозорого шифрування даних захищає «неактивні» дані, тобто файли даних і журналів.

Системи шифрування другого класу викликаються користувачем (непрозоре шифрування) і можуть використовувати як засоби шифрування самої СУБД, так і зовнішні відносно СУБД утиліти.

Розрізняють два основні методи шифрування: симетричне й асиметричне. В першому з них один і той же ключ використовується і для шифрування, і для розшифрування повідомлень. У свою чергу, симетричне шифрування поділяється на потокове і блокове шифрування.

Потоковий шифр можна перетворити на блоковий, розбиваючи вхідні дані на окремі блоки і шифруючи їх по окремості. Проте блокові шифри є більш криптостійкими порівняно з потоковими. Потокові шифри часто реалізуються в апаратному вигляді, оскільки представлення даних та їх обробка в потокових шифрах дуже близькі до обробки даних та їх передачі в апаратурі.

В асиметричних методах застосовуються два ключі. Один із них, несекретний, використовується для шифрування і може публікуватися разом з адресою користувача, другий, секретний, застосовується для розшифрування і відомий тільки одержувачу. Асиметричні методи шифрування дозволяють реалізувати так званий електронний підпис, або електронне завірення повідомлення, ідея якого розкрита в [4].

Послуги асиметричного шифрування можна реалізувати і за допомогою симетричних методів, якщо є надійна третя сторона, що знає секретні ключі своїх клієнтів. Для компенсації недоліків симетричного шифрування широко застосовується комбінована криптографічна схема, де за допомогою асиметричного шифрування передається сеансовий ключ, що використовується сторонами для обміну даними за допомогою симетричного шифрування.

Криптографічні методи дозволяють надійно контролювати цілісність інформації. На відміну від традиційних методів контрольного підсумовування, здатних протистояти тільки випадковим помилкам, криптографічна контрольна сума практично виключає всі можливості непомітної зміни даних.

Сучасні СУБД включають резервне копіювання і аудит як неодмінні складові системи безпеки. Суть резервного копіювання полягає в зберіганні копії БД. При необхідності (несанкціоноване видалення або модифікація БД) з цієї копії відновлюється правильна версія БД.

Аудит полягає у відстежуванні всіх значущих з погляду безпеки подій, які зберігаються в текстовому файлі (Log-файл). Цей файл шифрується при використовуванні прозорого шифрування для підвищення захищеності БД від атак зловмисників.

Реалізація протоколювання і аудиту має на меті: забезпечення підзвітності користувачів і адміністраторів; забезпечення можливості реконструкції послідовності подій; надання інформації для виявлення і аналізу проблем [4].

Дуже обширне або детальне протоколювання не тільки знижує продуктивність сервісів, що негативно позначається на доступності, але й утруднює аудит, зменшуючи, а не збільшуючи інформаційну безпеку. Складною проблемою є організація злагодженого протоколювання і аудиту в розподіленій різнорідній системі.

Для захисту БД можна використовувати такі основні засоби мови SQL: оператори надання і відміни прав доступу; збережені процедури і тригери; оператори для шифрування даних; резервне копіювання і відновлення даних.

Оператор GRANT використовується для надання користувачу прав доступу на вибірку, вставку, видалення і модифікацію стовпців у таблиці. Окрім цього, оператор GRANT може використовуватися власником БД для передачі привілеїв іншим користувачам. Оператор REVOKE використовується для відміни прав доступу на вибірку, вставку, видалення і модифікацію стовпців у таблиці.

Збережена процедура – це модуль (іменований набір команд) мови SQL, що зберігається на сервері і є самостійним об'єктом БД. Збережена процедура існує незалежно від таблиць або яких-небудь інших об'єктів БД і може використовуватися для шифрування даних в БД.

Тригер – це програмний блок, асоційований з таблицею БД, що автоматично виконує вказані в ньому дії, коли над зв'язаною з ним таблицею здійснено окремо визначену подію.

Оператори SQL для шифрування даних можуть використовуватися як для прозорого, так і для непрозорого шифрування. Для шифрування застосовуються функції T-SQL, що являють собою спеціальне доповнення мови SQL. T-SQL підтримує управляючі оператори, локальні змінні і різні додаткові функції [6].

Оператор CREATE KEY створює новий ключ, а оператор DROP KEY видаляє існуючий ключ. Для відкриття ключа використовується оператор OPEN KEY. Оператор ALTER KEY змінює властивості ключа. Додатково вказується тип ключа: SYMMETRIC (симетричний) або ASYMMETRIC (асиметричний).

Для того щоб запобігти пошкодженням БД внаслідок помилок у роботі апаратних засобів або мережних пристроїв, проводиться процедура паралельного резервування. Оператор CREATE SHADOW починає паралельне копіювання та створення тіні, а DROP SHADOW – припиняє ці дії. Отже, мова SQL має необхідні інструменти для захисту БД. Результати аналізу та оцінки сервісів безпеки (СБ), їх механізмів і моделей захисту БД наведені в табл. 1.

Таблиця 1

Аналіз та оцінка сервісів безпеки, їх механізмів і моделей захисту БД

СБ	Механізми і моделі захисту БД	Переваги	Недоліки
1	2	3	4
УПРАВЛІННЯ ДОСТУПОМ	**Дискреційна модель** Будується на основі виборчого принципу розмежування, при якому доступ до об'єктів здійснюється на основі безлічі дозволених відношень доступу	▪ універсальність; ▪ наочність; ▪ гнучкість; ▪ зручність для користувача при децентралізованому управлінні	▪ низькорівневість і складність; ▪ при децентралізованому управлінні утруднюється контроль; ▪ складність адміністрування, незручність і негнучкість при централізованому управлінні
	Мандатна модель Будується на принципі приписування об'єктам і суб'єктам доступу міток. Якщо мітки об'єкта і суб'єкта в якомусь значенні відповідають, то суб'єкт одержує право на доступ /дію, інакше ні	▪ обмеження прав користувача в управлінні своїми ресурсами; ▪ більш проста і зрозуміла, ніж дискреційна модель; ▪ може бути повністю формалізована; ▪ простота настроювання на конкретну ситуацію	▪ низькорівневість, громіздкість, жорсткість однорівневої моделі; ▪ описує тільки властивість конфіденційності; ▪ неможливість передачі даних від високого рівня до низького; ▪ складність реалізації
	Рольова модель Права доступу групуються з урахуванням специфіки їх застосування, утворюючи ролі. Управління правами доступу здійснюється на основі матриці та правил, що регламентують поведінку користувача	▪ більш високорівнева та інтуїтивно зрозуміла, ніж дискреційна і мандатна моделі; ▪ у користувача може бути декілька несуперечливих ролей; ▪ можливе успадкування привілеїв; ▪ допускає ієрархію	▪ складність перенесення ролей з іншої рольової моделі; ▪ складність реалізації; ▪ складність адміністрування (несуперечливість ролей); ▪ складніша для адміністрування, ніж з дискреційна і мандатна моделі
ІДЕНТИФІКАЦІЯ І АУТЕНТИФІКАЦІЯ	**Внутрішня аутентифікація** Підтвердження справжності користувача, що реалізується засобами самої БД	▪ гнучке управління атрибутами безпеки і правами користувачів; ▪ під час аутентифікації можливе використання SSL-протоколу	▪ вбудовані засоби захисту недостатні і вразливі; ▪ ключовий матеріал зберігається в контейнерах, як звичайні файли
	Зовнішня аутентифікація Підтвердження справжності користувача, що реалізується засобами ОС або мережі.	▪ зручність користувача СУБД (не треба запам'ятовувати додатковий пароль для зв'язку з СУБД)	▪ зменшується рівень захисту СУБД (залежно від захисту ОС); ▪ використовується для аутентифікації адміністратора
	Біометрична аутентифікація: ***- статична; - динамічна*** Процес доведення і перевірки достовірності заявленого користувачем імені через пред'явлення свого біометричного образу	▪ біометричні характеристики важко підробити; ▪ зручно для користувача; ▪ біометрика залишається перспективним механізмом захисту, що розвивається	▪ висока ймовірність помилок; ▪ компрометація біометричних даних веде до суттєвих змін; ▪ складний супровід системи; ▪ малоефективна на неконтрольованій території; ▪ висока вартість

Продовження табл. 1

	2	3	4
	Парольна аутентифікація (ґрунтується на знанні) Як загальний секрет користувача і системи виступає послідовність знаків абетки, яку користувач в змозі запам'ятати. Для отримання доступу до ресурсу БД користувач представляє свій ідентифікатор і пароль	▪ простота і дешевизна; ▪ простота зміни пароля; ▪ при правильному використовуванні (накладення технічних обмежень, управління термінами дії паролів, обмеження кількості невдалих спроб входу в систему) забезпечується достатній рівень захисту	▪ один з найслабкіших механізмів аутентифікації; ▪ простота дублювання; ▪ безпека використання пароля залежить від його якості; ▪ вразливість відносно електронного перехоплення; ▪ не відповідає вимогам, що висуваються до безпеки ІС
КРИПТОГРАФІЯ	**Прозоре шифрування** Процес включається адміністратором і виконується в реальному часі для шифрування і дешифрування даних і журналів в операціях введення-виведення. Використовується ключ шифрування БД, який зберігається в завантажувальному записі БД	▪ дані на носії зашифровані, а в незашифрованому вигляді знаходяться мінімальний час; ▪ шифрування і розшифрування виконуються абсолютно прозоро для додатків; ▪ забезпечується відповідність вимогам багатьох законів і рекомендацій в галузі безпеки ІТ	▪ велике навантаження на ЦП; ▪ необхідність збереження резервних копій сертифікатів; ▪ складна і дорога система; ▪ складність перенесення системи з одного програмного середовища в інше; ▪ забезпечення постійного контролю інфопотоків
	Непрозоре шифрування Процес ініціюється користувачем і використовує зовнішні утиліти і засоби СУБД. Шифрування створює об'єкт, що містить зашифрований образ початкового	▪ мінімальне навантаження на центральний процесор для шифрування і дешифрування; ▪ немає необхідності адміністрування	▪ великий час знаходження даних у незашифрованому вигляді; ▪ для забезпечення достатньої стійкості криптографічного захисту необхідне одночасне виконання багатьох умов
	Симетричне шифрування: ***- потокове; - блокове*** Один і той же ключ (секретний) використовується для шифрування і дешифрування. При потоковому шифруванні одиницею кодування є біт, а при блочному – дані шифруються в блоках	▪ висока швидкість; ▪ простота реалізації; ▪ менша необхідна довжина ключа для порівняної стійкості; ▪ вивченість (за рахунок більшого віку); ▪ можливість використання складових ключів (принцип розподілу обов'язків)	▪ складність управління ключами у великій системі; ▪ секретний ключ повинен бути відомий обом сторонам і постійно оновлюватися; ▪ неможливість використання для підтвердження авторства; ▪ вразливість ключа до крадіжок
	Асиметричне шифрування Використовуються два ключі. Один з них (несекретний) використовується для шифрування, другий (секретний) – для дешифрування	▪ не потрібно попередньо передавати особистий ключ; ▪ секретний ключ відомий тільки одній стороні; ▪ ключі можна міняти рідко; ▪ кількість ключів є невеликою	▪ довгі ключі і мала швидкодія; ▪ складність внесення змін; ▪ необхідність спільного використання з симетричними методами для підвищення ефективності системи
ЗАХИСТ СЕРВІСІВ, ПРОТОКОЛЮВАННЯ І АУДИТ	**Резервне копіювання:** ***- повне; - диференційне; - інкрементальне*** Збереження копії БД. Копіюватися може вся БД (повне) або тільки зміни з попередньої копії (диференційне) з можливістю зберігання на кількох носіях (інкрементальне)	▪ повне резервне копіювання є основою стратегії відновлення; ▪ вибрані дані будуть скопійовані без втручання в схему ротації носіїв; ▪ дозволяє звільнити пам'ять і забезпечити більш ефективну організацію даних	▪ надлишковий захист даних при повному копіюванні; ▪ повне копіювання може займати досить великий час; ▪ для інкрементального копіювання ефективне відновлення вимагає великої кількості часу і правильного порядку обробки носіїв
	Протоколювання і аудит Протоколювання – збір і накопичення інформації про події, що відбуваються в системі. Аудит – відстежування і аналіз всіх значущих подій, який проводиться оперативно або періодично	▪ важливий стримуючий чинник для користувачів сервісів; ▪ виявлення слабкості в захисті сервісів; ▪ при правильній організації може істотно посилити захист	▪ детальне протоколювання знижує продуктивність сервісів і утруднює аудит; ▪ складність організації злагодженого протоколювання і аудиту в розподіленій різнорідній системі; ▪ залежність від інших засобів
УПРАВЛІННЯ БЕЗПЕКОЮ	**Засоби мови SQL:** - оператори надання і відміни прав доступу; - збережені процедури і тригери; - оператори шифрування даних; - оператори резервного копіювання і відновлення даних	▪ незалежність від конкретної СУБД; ▪ наявність стандартів; ▪ декларативність; ▪ виконання більшості операцій із захисту даних в СУБД	▪ складність (потрібен певний рівень знань); ▪ невідповідність реляційній моделі даних; ▪ складність роботи з ієрархічними структурами; ▪ порушення стандартів

Висновки:

1. В сучасних СУБД використовуються гібридні моделі захисту, які включають дискреційну, мандатну і рольову моделі безпеки. З усіх моделей безпеки найзручнішою для користувачів є рольова модель, проте вона найскладніша для адміністрування. За допомогою мандатної моделі можна створювати багаторівневі системи захисту. Найпростішою моделлю є дискреційна, але вона може виявитися надмірно детальною.

2. З усіх схем аутентифікації найчастіше використовується парольний захист, зважаючи на дешевизну і простоту. Часто використовується зовнішня аутентифікація за допомогою парольного захисту ОС, оскільки це зручно для користувачів. Досить поширеною є аутентифікація за допомогою токенів. Перспективною є біометрична аутентифікація.

3. У сучасних СУБД широко використовується прозоре шифрування, оскільки при цьому дані завжди зашифровані, хоча це створює додаткове навантаження на центральний процесор. Окрім цього, при прозорому шифруванні користувачу не треба змінювати свої програми. Спільне використання симетричних і асиметричних методів шифрування підвищує ефективність СУБД та зменшує їх завантаженість.

4. Обов'язковою частиною системи безпеки СУБД є системи резервного копіювання (відновлення) і аудиту. Резервне копіювання і відновлення в сучасних СУБД може здійснюватися через графічний інтерфейс, а також за допомогою команд SQL.

5. Мова SQL відіграє важливу роль у захисті СУБД. За допомогою команд SQL можна виконувати практично всі аспекти захисту СУБД.

6. Для ефективного захисту БД в СУБД потрібен комплексний, систематичний підхід, необхідне поєднання різних сервісів безпеки та їх механізмів.

Список літератури

1. Нечипоренко О. В. Классификационная схема моделей баз данных для лазерных технологических комплексов / О. В. Нечипоренко, С. А. Миценко // Вісник Черкаського державного технологічного університету. – 2013. – № 2. – С. 48–54. – (Серія : технічні науки).
2. Анализ концептуальных подходов к обеспечению защиты баз данных [Электронный ресурс] // Мир компьютеров. – Режим доступа : http://compsmir.ru/?p=112
3. Лукашенко В. М. Системний аналіз біометричних датчиків відбитків пальця для системи управління доступом лазерного технологічного комплексу / В. М. Лукашенко, Т. Ю. Уткіна, О. С. Вербицький та ін. // Вісник Черкаського державного технологічного університету. – 2012. – № 4. – С. 29–34. – (Серія : технічні науки).
4. Галатенко В. Информационная безопасность [Электронный ресурс] / В. Галатенко // Открытые системы. СУБД. – 1996. – № 04. – Режим доступа : http://www.osp.ru/os/1996/04/178931/
5. Шифрование данных в СУБД [Электронный ресурс] // Мир компьютеров. – Режим доступа : http://compsmir.ru/?p=118
6. Комаров А. Базу данных не стащить! Правильные способы защитить данные в таблицах БД [Электронный ресурс] / А. Комаров // Хакер. – № 04/09 (124). – Режим доступа : http://www.xakep.ru/magazine/xa/124/032/1.asp

References

1. Nechyporenko, O. V. and Mitsenko, S. A. (2013) Classification scheme of database models for laser technological complexes. *Visnyk Cherkaskogo derzhavnogo technologichnogo universitetu. Seriya: technichni nauky,* (2), pp. 48–54 [in Russian].
2. The analysis of conceptual approaches to providing of data-base defence [Internet]. *World of computers*. Available from: http://compsmir.ru/?p=112
3. Lukashenko. V. M., Utkina, T. Yu., Verbytskiy, O. S. et al. (2012) System analysis of biometric fingerprint sensors for access control systems of laser technological complex. *Visnyk Cherkaskogo derzhavnogo technologichnogo universitetu. Seriya: technichni nauky,* (4), pp. 29–34 [in Ukrainian].
4. Galatenko V. (1996) Information security [Internet]. *Open systems. SUBD,* (04). Available from: http://www.osp.ru/os/1996/04/ 178931/
5. Data coding in SUBD [Internet]. *World of computers*. Available from: http://compsmir.ru/?p=118

6. Komarov, A. Not to steal the database! Correct methods to protect data in DB tables [Internet]. *Hacker*, 04/09 (124). Available from: http://www.xakep.ru/magazine/xa/124/032/1.asp

Стаття надійшла до редакції 00.08.2014.

O. V. Nechyporenko, *Ph.D. (Engineering), associate professor*
S. A. Mitsenko, *post-graduate student*
Cherkasy State Technological University
Shevchenko blvd., 460, Cherkasy, 18006, Ukraine
kafedra_ckc@mail.ru

MECHANISMS FOR DATABASE DEFENCE IN MODERN DBMS

The article presents an analysis of basic security services and qualitative assessment of defence mechanisms and security models for determining of effective data security technologies in modern DBMS using a hybrid security model, which includes discretionary, mandatory and role security models. The role model is the most comfortable security model for users, but it is most difficult to administer. With the help of mandatory model it is possible to generate multi-protection systems. Discretionary model is the simplest one, but it may be too detailed.

Authentication schemes often use password security given the low cost and simplicity. External authentication via password security of operating system is often used, because it is convenient for users. The authentication with the help of tokens is quite common. Biometric authentication is perspective.

In modern DBMS transparent encryption is widely used, as this data is always encrypted, even though it creates an additional burden on CPU. Besides at transparent encrypting the user does not need to change his programs. Sharing of symmetric and asymmetric encryption methods improves DBMS efficiency and reduces their database load.

File backup and audit systems are compulsory part of database security. File backup and recovery in modern DBMS can be carried out through GUI and using SQL commands. SQL plays an important role in database security. Using SQL commands, you can perform almost all aspects of DBMS security. To effectively protect the database in DBMS a comprehensive, systematic approach, a combination of different services and their security mechanisms are needed.

Keywords: *data security, access control, security model, authentication, encryption, logging and audit, SQL operators.*