УДК 004-658

МАТЕМАТИЧНА МОДЕЛЬ МОНІТОРИНГУ ПОДІЙ В СИСТЕМІ УПРАВЛІНЯ РОЗПОДІЛЕНИМИ БАЗАМИ ДАНИХ НА ОСНОВІ СЕНСОРІВ ПОДІЙ

DOI 10.36994 / 2788-5518-2023-01-05-17

Корнага Я.І., *д.т.н., проф., Національний технічний університет України «Київський політехнічний університет імені Ігоря Сікорського»,* Київ, Україна, *slovyan_k@ukr.net*
Барабаш А.О. *Національний технічний університет України «Київський політехнічний університет імені Ігоря Сікорського»,* Київ, Україна. *andrew.barbsh@gmail.com*

Анотація: *У роботі розглянуто методи підвищення ефективності виявлення загроз в розподілених базах даних за допомогою системи моніторингу подій. Зазначено, що система працює на основі моделі моніторингу подій різнорідних розподілених баз даних. Дана модель передбачає три етапи обробки подій, які базуються на відповідних методах. Виокремлено, що механізми, які покладені в основу функціонування згаданих методів, повинні бути приведені до єдиного формату даних для усунення можливої появи некоректної роботи в майбутніх обчисленнях. Метод обробки подій на сервері моніторингу дозволяє обробляти матриці подій та передавати на засоби контролю функцію на основі якої приймаються відповідні рішення щодо підвищення надійності.*

В статті розроблено модифікований метод аналізу та моніторингу подій в нереляційних розподілених базах даних. Запропоновано параметри моніторингу подій для операцій пошуку в розподілених базах даних. Для підтвердження теоретичних результатів проведено математичне моделювання та експериментальне дослідження параметрів моніторингу та швидкості обробки даних на сервері моніторингу подій. Досліджено кількості виявлених інцидентів та кількості невірно ідентифікованих подій, що в свою чергу дозволяє забезпечувати захист інформації від несанкціонованих дій чи змін. Також, слід підкреслити, що запропонована в роботі модель описує детальний моніторинг запитів, транзакцій та збережених процедур в базі даних з інформацією щодо можливих інцидентів в режимі реального часу та спонукає до попереджень вторгнень. Крім того, особливості запропонованої моделі дозволяють досліджувати та відстежувати появу нових вразливих сегментів бази даних та оперативно реагувати на них, усуваючи загрози каналам доступу. В роботі проведено порівняння швидкості обробки даних на основі файлів та таблиць баз даних, що дозволяє в повній мірі виконати експерименти та підтвердити аналітичні дані експериментальними.

Ключові слова: *аналіз подій, бази даних, моніторинг подій, розподілені бази даних, швидкість обробки.*

MATHEMATICAL MODEL OF EVENTS MONITORING IN THE MANAGEMENT SYSTEM OF DISTRIBUTED DATABASES BASED ON EVENT SENSORS

Yaroslav Kornaga, *Dr.habil., Prof. National Technical University of Ukraine «Igor Sikorsky Kyiv Polytechnic Institute», Kyiv, Ukraine. slovyan_k@ukr .net*
Andrii Barabash. *National Technical University of Ukraine «Igor Sikorsky Kyiv Polytechnic Institute», Kyiv, Ukraine., andrew.barbsh@gmail.com.*

Abstract: *Methods of increasing the effectiveness of threat detection in distributed databases using the event monitoring system are considered in the work. It is noted that the system works on the basis of the event monitoring model of heterogeneous distributed databases. This model involves three stages of event processing, which are based on appropriate methods. It is highlighted that the mechanisms underlying the functioning of the mentioned methods should be reduced to a single data format to eliminate the possible appearance of incorrect work in future calculations. The event processing method on the monitoring server allows for processing event matrices and transferring the function to control tools, on the basis of which appropriate decisions are made to improve reliability.*

The article develops a modified method of analyzing and monitoring events in non-relational distributed databases. Event monitoring options for lookup operations in distributed databases are offered. To confirm the theoretical results, mathematical modeling and experimental research on monitoring parameters and data processing speed on the event monitoring server were carried out. The number of detected incidents and the number of incorrectly identified events were studied, which in turn allowed for the protection of information from unauthorized actions or changes. Also, it should be emphasized that the model proposed in the work describes

detailed monitoring of requests, transactions, and stored procedures in a database with information about possible incidents in real time and prompts intrusion warnings. In addition, the features of the proposed model make it possible to investigate and monitor the emergence of new vulnerable segments of the database and promptly respond to them, eliminating threats to access channels. The paper compares the speed of data processing based on files and database tables, which allows for fully performing experiments and confirming analytical data experimentally.

Keywords: *event analysis, databases, event monitoring, distributed databases, processing speed.*

Вступ

В сучасних різнорідних розподілених системах управління базами даних (СУБД) та існуючих пошукових системах використовуються різноманітні способи, методи та засоби моніторингу подій, які дозволяють здійснювати керування базами даних у відповідності із вимогами що висуваються до систем. Механізми, що дозволяють одночасно налаштовувати параметри функціонування бази даних та управляти нею потребують приведення до єдиного формату обробки даних. При цьому, методи обробки подій в СУБД, що використовуються для операцій пошуку, не в повній мірі висвітлюють вплив даних подій на конкретні умови функціонування бази даних [1,2,3].

Виконання досліджень

З метою збільшення швидкості операцій в базах даних необхідно створити новий метод моніторингу подій, який дозволить удосконалити та уніфікувати обробку подій в різнорідних розподілених баз даних та за допомоги додаткового сервера пришвидшить їх обробку, а також, надасть можливість нейтралізувати можливі загрози шляхом автоматичного втручання системи без обслуговуючого персоналу або за допомоги системного адміністратора.

Класична модель моніторингу подій базується на основі алгоритму найбільших статистичних аномалій (AHCA). Основу даного алгоритму складає формування вектору Бернулі, який в подальшому обробляється сервером моніторингу подій. Таким чином, формується вектор $\bar{X} = \{x_1, x_2, \dots, x_n\}$, який порівнюється з пороговим вектором, після чого пересилається на сервер обробки моніторингу подій. В свою чергу, сервер приймає рішення щодо тієї чи іншої події. Складові вектору Бернулі включають в себе тільки параметри подій. Проте, вектор не містить параметри впливу події на той чи інший елемент системи[3]. Тому потрібно створити систему моніторингу подій, яка б могла обробляти не тільки події, а й їх розподіл за відповідними властивостями бази даних та СУБД.

Запропонована математична модель моніторингу подій розподілених баз даних заснована на використанні сенсорів подій та включає в себе такі кроки:

I. Збір подій моніторингу.

1. Для кожної бази даних з розподіленою системою управління будуємо сенсори подій на основі тригерів.

2. Для кожної множини подій $\overline{E_\iota} = \{e_1, e_{2,\dots,} e_n\}$, що вимірюється сенсорами подій, будується порогова матриця за можливими наслідками кожної події із множини $\overline{C_\iota} = \{c_1, c_{2,\dots,} c_m\}$.

3. Сенсори подій проводять збір інформації для матриці подій $S_i = \overline{E_\iota} \times C_i$ з простору S і також пересилають їх до серверу моніторингу подій.

II. Метод обробки подій моніторингу на сервері моніторингу подій.

1. Після передачі матриці подій на сервер моніторингу подій проводяться операції аналізу відмінності між отриманою матрицею S_i та заданною пороговою матрицею S_0, тобто $B_i = S_i - S_0$.

2. Проводиться формування експертних оцінок на основі формування вектору $\overline{W_\iota} = \{w_1, w_{2,\dots,} w_m\}$.

3. Після отримання вектору експертних оцінок $\overline{W_\iota}$, матриця B_i множиться на вектор експертних оцінок $\overline{W_\iota}$: $\overline{Z_\iota} = B_i \times \overline{W_\iota}$.

4. Отримані результати оцінюються за допомогою функції R, яка визначає від'ємні значення у векторах, та формує вектор відповіді $\overline{F_\iota} = \{f_1, f_{2,\dots,} f_n\}$ в якому записується результат обробки події.

$$\bar{F} = R(\overline{Z_\iota}) = R(B_i \times \overline{W_\iota}) = R((S_i - S_0) \times \overline{W_\iota}). \quad (1)$$

III. Обробка відповіді серверу моніторингу подій.

1. Вектор відповіді $\overline{F_i} = \{f_1, f_{2,\dots,} f_m\}$ передається в модуль контролю подій.

2. В модулі контролю подій у випадку наявності подій, що можуть спричинити негативний вплив на бази даних розподіленої системи, генерується рішення про застосування певних заходів щодо нейтралізації негативних наслідків.

3. У випадку виникнення нештатних ситуацій, модуль контролю подій пересилає сигнал на блок формування попередження для адміністратора серверу моніторингу подій.

Архітектура засобів моніторингу подій розподілених баз даних на основі сенсорів подій.

Отримана система моніторингу подій, в автоматичному режимі, ідентифікує всі події, які відбуваються у розподілених базах даних. Під час роботи сенсор подій приєднується до тимчасово-виділеної пам'яті та циклічно проводить опитування з моніторингу подій на основі вибірки даних з пам'яті із деякою частотою[4]. Під час кожного циклу сенсор робить аналіз активних, у цей момент процесів, під час кожної сесії в базі даних та визначає події на основі правил, які встановлені адміністратором. Надалі інформація про події надсилається на сервер моніторингу подій де відбувається їх подальший аналіз та генеруються відповідні повідомлення.

На рис. 1 показано схему, що була отримана на основі розробленої моделі.

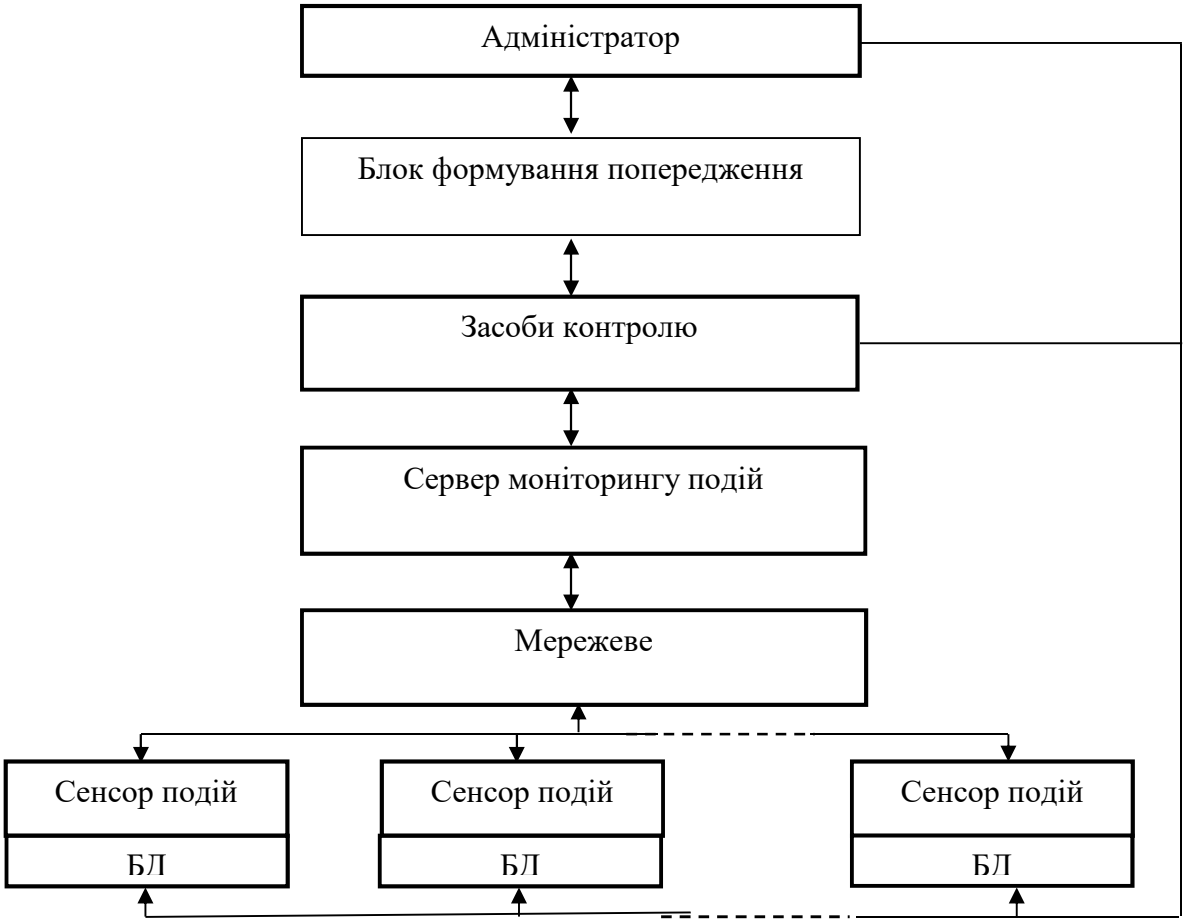

Рис.1. Архітектура моделі моніторингу подій на основі сенсорів подій

Система моніторингу подій у свою чергу також має можливість примусово припиняти сесії роботи у випадку певних несанкціонованих подій, що можуть бути вчинені, з боку користувачів. З іншого сторони, для роботи сенсора подій необхідні невеликі обчислювальні ресурси бази даних та його робота практично не впливає на основні операції щодо обробки запитів. Попереджуючу функція сенсора можна реалізувати із використанням DDL-тригерів, що вибірково призупиняють виконання команд DDL та DCL на невеликі проміжки часу (до декількох мс), а це дозволяє сенсору подій вчасно реагувати щодо небезпечних дій [5].

На рис. 2 представлена детальне архітектури сенсора і серверу подій.

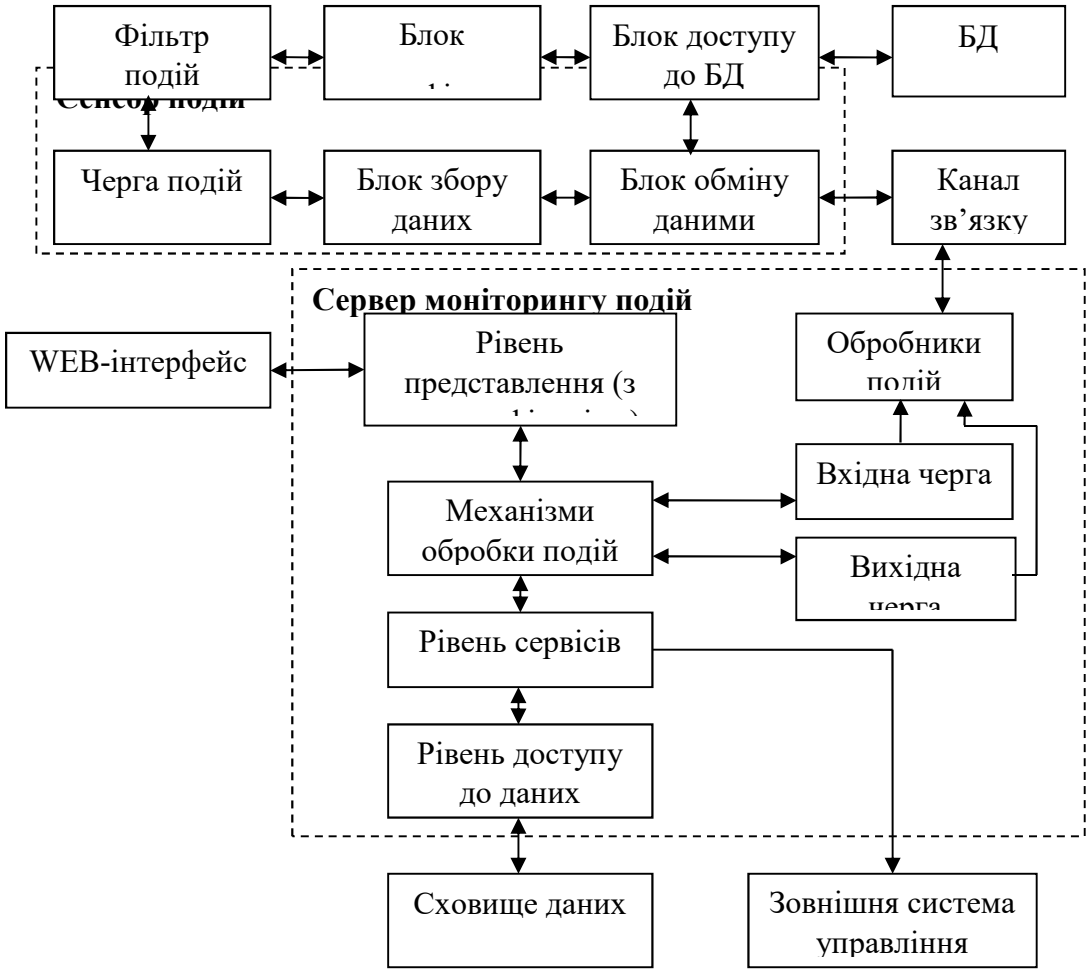

Рис.2.Зображення архітектури сенсора подій і сервера моніторингу подій

Визначений сервер моніторингу подій у представленій системі моніторингу подій здатний керувати декількома сенсорами подій із різних розподілених баз даних. Крім того, сервер підтримує опцію масштабування кількості сенсорів подій. Він достатньо легко інтегрується у структуру системи моніторингу СУБД. В свою чергу це дозволяє підвищити загальну ефективність управління надійністю обробки даних у розподілених базах даних.

Архітектура запропонованої системи забезпечує розподіл прав та повноважень між суб'єктами, що є однією із основних умов, які висуваються до систем найденої обробки різного типу даних[6]. Адміністратор даної системи є персоною, яка визначає основні правила політики надійності і безпеки та отримує повідомлення щодо інцидентів використовуючи блок формування попередження. Розглянута система дозволяє по-перше забезпечити потрібний рівень надійності щодо обробки даних у розподілених базах даних, а по-друге – неперервне виконання обробки запитів.

Методика оцінки ефективності системи моніторингу подій розподілених баз даних.

Щоб провести дослідження параметрів системи моніторингу подій у розподілених базах даних була розроблена методика, що дозволяє оцінювати ефективність роботи сервера моніторингу подій на основі використання сенсорів подій [7].

Операції у базах даних класифікуються за двома ознаками:

- Дозволені.
- Не дозволені.

Нехай G – кількість можливих подій у базі даних, K – множина правильно відпрацьованих подій в базі даних. Позначимо: $i \in G$ – множина подій, що не були знайдені системами

моніторингу подій, $o \in K$ – множина правильно відпрацьованих подій, що були неправильно ідентифіковані системою моніторингу подій як елементи, які порушують надійність.

Ймовірність коректного виявлення подій можна розраховувати наступним чином:

$$P_i = 1 - \frac{i}{G}. \quad (2)$$

.Ймовірність неправильних ідентифікацій правильно відпрацьованих подій розраховується у вигляді:

$$P_k = 1 - \frac{i}{G-i+k}, \quad (3)$$

де $G - i + k -$ це кількість несанкціонованих подій, що були виявлені за допомогою системи моніторингу подій.

На основі цього ймовірність правильного виявлення несанкціонованих подій системою моніторингу подій можна розрахувати на основі формули:

$$P = 1 - max\{P_i, P_k\} = 1 - max\left\{\frac{i}{G}, \frac{k}{G-i+k}\right\}. \quad (4)$$

Визначений показник можна використовувати для комплексної оцінки ефективності щодо функціонування системи моніторингу подій у розподілених базах даних [8].

Крім того, ймовірність коректної роботи серверу моніторингу подій можна визначити наступним чином:

$$P = 1 - max\{P_I, P_{II}\}, \quad (5)$$

де P_I – ймовірність помилки першого роду («пропущена подія» – відсутність реакції на подію), P_{II} – ймовірність помилки другого роду («хибна тривога» – помилкове спрацювання).

Експериментальне підтвердження, як це показано на рисунку 3, відбувалися у двох напрямках, а саме:

- Знаходження кількості спроб інцидентів.
- Загальна кількість помилок першого та другого роду, які знайдені в системі моніторингу подій.

Аналіз експериментальних досліджень показує, що сумарна імовірність помилок першого та другого роду складає 6 – 12 %. Це в свою чергу дозволяє стверджувати, що імовірність коректного функціонування серверу моніторингу складає 88 – 94 %.

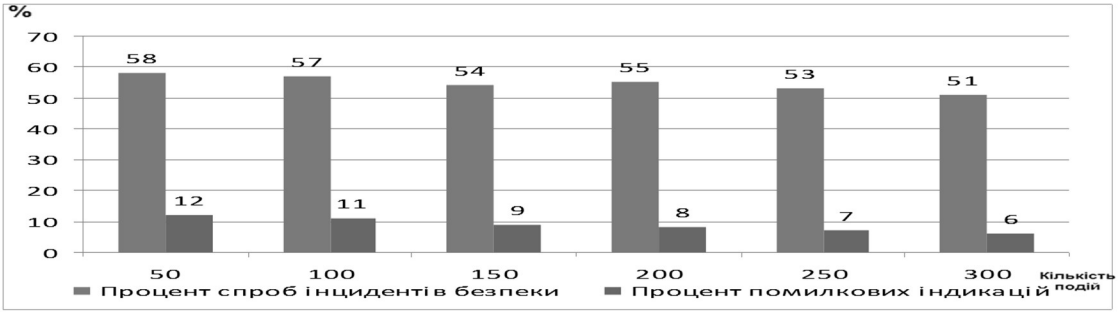

Рис. 3. Результати експериментальних досліджень серверу моніторингу подій

Висновки

Запропоновано математичну модель реалізації моніторингу подій баз даних, на основі якої виконується всебічний та комплексний моніторинг усіх дій з базою даних та забезпечується захист інформації від несанкціонованих дій легальних користувачів. Разом із цим, запропонована модель дозволяє описати детальний моніторинг запитів, транзакцій та збережених процедур в базі даних з повідомленнями щодо інцидентів в реальному режимі часу та попередженням вторгнень. Крім того, запропонована модель дозволяє дослідити та відстежити нові виявлені

вразливі місця бази даних й оперативно усунути ці вразливі канали доступу. Це є досить важливим в практиці застосування розподілених баз даних.

Література

1. Vadym Mukhin, Valerii Zavgorodnii, Yaroslav Kornaga, Ivan Krysak, Maxim Bazaliy, Oleg Mukhin. Program Code Protecting Mechanism Based on Obfuscation Tools. IEEE International Conference on System Analysis&Intelligent Computing. 2022. Pp 407-419.
2. Marcus Hirt, Marcus Lagergren. Oracle jrockit: The Definitive Guide. Packt Publishing, 2010. P.588.
3. Vadym Mukhin, Yaroslav Kornaga, Yurii Bazaka, Ievgen Krylov, Andrii Barabash, Alla Yakovleva, Oleg Mukhin. The Testing Mechanism for Software and Services Based on Mike Cohn's Testing Pyramid Modification. 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS). 2021. Pp.589-595.
4. Teo Lachev. Applied Microsoft SQL Server 2008. Reporting Services. Prologika Press, 2008. P.768.
5. Коннор Мак Дональд. Oracle. Практичні рішення. 2006. С.560.
6. Babich Oksana, Vyshnyvskiy Viktor, Mukhin Vadym, Zamaruyeva Irina, Sheleg Michail, Kornaga Yaroslav. The Technique of Key Text Characteristics Analysis for Mass Media Text Nature Assessment. International Journal of Modern Education&Computer Science. 2022. Vol. 14, Issue 1, pp.1-16.
7. Мухін В.Є., Корнага Я.І. Механізми підвищення ефективності процедури моніторингу безпеки в розподілених базах даних. Вісник Національного технічного університету «Харківський політехнічний інститут». Збірник наукових праць. Серія: Інформатика та моделювання. Харків: НТУ «ХПІ». 2012. № 38., С.128-135.
8. Vadym Mukhin, Valerii Zavgorodnii, Oleg Barabash, Anna Zavgorodnya, Vitalii Statkevych. Method of Restoring Parameters of Information Objects in a Unified Information Space Based on Computer Networks. International Journal of Computer Network & Information Security. 2020.12 (2).

References

1. Vadym Mukhin, Valerii Zavgorodnii, Yaroslav Kornaga, Ivan Krysak, Maxim Bazaliy, Oleg Mukhin. Program Code Protecting Mechanism Based on Obfuscation Tools. *IEEE International Conference on System Analysis&Intelligent Computing*. 2022. Pp. 407-419.
2. Marcus Hirt, Marcus Lagergren. Oracle jrockit: The Definitive Guide. Packt Publishing, 2010. P.588.
3. Vadym Mukhin, Yaroslav Kornaga, Yurii Bazaka, Ievgen Krylov, Andrii Barabash, Alla Yakovleva, Oleg Mukhin. The Testing Mechanism for Software and Services Based on Mike Cohn's Testing Pyramid Modification. *11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*. 2021. Pp.589-595.
4. Teo Lachev. Applied Microsoft SQL Server 2008. Reporting Services. Prologika Press, 2008. P.768.
5. Konnor Mak Donald. Oracle. Praktychni rishennia [Oracle. Practical solutions]. 2006. C.560.
6. Babich Oksana, Vyshnyvskiy Viktor, Mukhin Vadym, Zamaruyeva Irina, Sheleg Michail, Kornaga Yaroslav. The Technique of Key Text Characteristics Analysis for Mass Media Text Nature Assessment. *International Journal of Modern Education&Computer Science.* 2022. Vol. 14, Issue 1, pp.1-16.
7. Mukhin V.I., Kornaha Y.I. Mekhanizmy pidvyshchennia efektyvnosti protsedury monitorynhu bezpeky v rozpodilenykh bazakh danykh [Mechanisms for increasing the efficiency of the security monitoring procedure in distributed databases] Visnyk Natsionalnoho tekhnichnoho universytetu «Kharkivskyi politekhnichnyi instytut». Zbirnyk naukovykh prats. Seriia: Informatyka ta modeliuvannia. Kharkiv: NTU «KhPI». 2012. № 38., S.128-135.
8. Vadym Mukhin, Valerii Zavgorodnii, Oleg Barabash, Anna Zavgorodnya, Vitalii Statkevych. Method of Restoring Parameters of Information Objects in a Unified Information Space Based on Computer Networks. *International Journal of Computer Network & Information Security.* 2020. 12 (2).